Advanced CAE Applications for Professionals

Software that works — for you.[SM]

UAI/NASTRAN
User's Reference Manual
for Version 20.1

UNIVERSAL ANALYTICS, INC.

Publication ND-001

©1993-1999 UNIVERSAL ANALYTICS, INC.
Torrance, California USA
All Rights Reserved

First Edition, February 1991
Second Edition December 1992
Third Edition, April 1994
Fourth Edition, September 1995
Fifth Edition, December 1997
Sixth Edition, March 1999

Restricted Rights Legend:

The use, duplication, or disclosure of the information contained in this document is subject to the restrictions set forth in your Software License Agreement with Universal Analytics, Inc. Use, duplication, or disclosure by the Government of the United States is subject to the restrictions set forth in Subdivision (b)(3)(ii) of the Rights in Technical Data and Computer Software clause, 48 CFR 252.227-7013.

The information contained herein is subject to change without notice. Universal Analytics Inc. does not warrant that this document is free of errors or defects and assumes no liability or responsibility to any person or company for direct or indirect damages resulting from the use of any information contained herein.

UNIVERSAL ANALYTICS, INC.

3625 Del Amo Blvd., Suite 370
Torrance, CA 90503
Tel: (310) 214-2922
FAX: (310) 214-3420

FOREWORD

The **UAI/*NASTRAN*** User's Reference Manual has been designed to provide you with all of the detailed information necessary for you to create finite element models and perform analyses which encompass a wide variety of analytical disciplines. The manual includes eight Chapters:

1. **JOB CONTROL**
2. **EXECUTIVE CONTROL COMMANDS**
3. **SUBSTRUCTURE COMMANDS**
4. **CASE CONTROL COMMANDS**
5. **STRUCTURAL PLOTTING COMMANDS**
6. **X-Y PLOTTING COMMANDS**
7. **BULK DATA ENTRIES**
8. **DIRECT MATRIX ABSTRACTION**

Chapter 1 provides you with information needed to execute **UAI/*NASTRAN*** on your host computer. Chapters 2, 3, 4, 5, and 6 describe the different command structures which select analysis methods, control substructuring procedures, define boundary and loading conditions, and request graphical output. All of the detailed Bulk Data entries used to define analysis models is found in Chapter 7. Chapter 8 provides you with descriptions of the DMAP modules that you may use to modify the standard **UAI/*NASTRAN*** analyses and to import and export data to and from **UAI/*NASTRAN***.

The companion to this volume is the **UAI/*NASTRAN User's Guide***. The Guide provides you with detailed information on the modeling and analytical disciplines of the system. It includes descriptions of finite elements, hints to improve your modeling practices, examples of Bulk Data use and many test problems and their solutions. It is strongly recommended that you review the ***User's Guide*** thoroughly before using a **UAI/*NASTRAN*** capability which is new to you.

NASTRAN® is a registered trademark of the National Aeronautics and Space Administration

This page is intentionally blank.

VERSION 20.1 RELEASE NOTES

This section summarizes the specific input file differences between **UAI/*NASTRAN*** Version 20.1 and previous versions of the program. This discussion is limited to a description of input data differences, using the ***User's Reference Manual*** format. A more comprehensive discussion of new features and differences with respect to earlier versions of **UAI/*NASTRAN*** is presented in the Release Notes section of the ***User's Guide***.

ONLINE DOCUMENTATION

As part of UAI's ongoing modernization program, all **UAI/*NASTRAN*** manuals, as well as those for other UAI software products, continue to be delivered in Adobe ***Portable Document File*** (PDF) format. This allows them to be used online with any computer having the Adobe Acrobat Reader (Version 3.0 or higher). This reader is also delivered with our software.

To insure prompt updates to all documentation, any changes and enhancements may be downloaded from UAI's Web site at **`www.uai.com`**.

DOCUMENTATION TRACKING

Beginning with Version 20.1, the documentation has been slightly modified so that each page of specific commands and data entries has a revision date on it. The two forms are: New: V20.1, indicating that the page is new with the current release; and Rev: V20.1, which indicates that the page has been modified for the current release.

NEW FEATURES

Version 20.1 contains several new features as well as many minor enhancements and bug corrections. These features include:

☞ A new feature to perform automatic modal reductions, including *Craig-Bampton*, and automatic static reduction. This feature allows models to be exported in the form of **`DMIG`** Bulk Data entries. The exported models may then be use to couple structural models from different sources. See Chapter 6 of the ***User's Guide*** for complete information.

☞ ***ARCHIVE Database***. Extensions have been added to the ARCHIVE database. These allow DMAP data block entities to be exported to, and imported from, DMAP solution sequences. See DMAP Chapter of this manual.

☞ ***Equivalent Beam Forces.*** New feature for computing equivalent beam forces (moments, shears, axial loads and torques) for sets of solid elements. See Chapter 5 of the ***User's Guide***.

☞ ***Mode Tracking in Design Optimization.*** A new capability in Design Optimization allows the automatic tracking of modes. This is important during the redesign procedure to capture "mode swapping" as the design changes. See Chapter 26 of the ***User's Guide***.

The following sections describe new features and modifications to **UAI*/NASTRAN*** input data.

EXECUTIVE CONTROL PACKET

COMMAND	STAT	DESCRIPTION
APPROACH	**REV**	Clarification of the use of the various options.
ENTITY	**NEW**	New feature for: - defining groups of database entities - assigning groups to selected eBase databases.
SECONVERT	**REV**	Requests execution of a new version of the MSC/NASTRAN Superelement convertor.
SEQUENCE	**REV**	New feature to select or deselect the inclusion of MPC and Rigid Element data in the resequencing.

SUBSTRUCTURE CONTROL PACKET

COMMAND	STAT	DESCRIPTION
COMBINE	**REV**	Enhanced to allow the new **MATCH** option for automatically combining GRID points with identical identification numbers. Especially useful in conjunction with the **SECONVERT** utility.

CASE CONTROL PACKET

COMMAND	STAT	DESCRIPTION
AUTOREDUCE	**NEW**	New Command for automatic Guyan reduction of a model. (See also **NLREDUCE**)
AUTOSPC	**REV**	New feature to select or deselect the application of **AUTOSPC** to both the *g-set* and *n-set* for nonlinear analyses.
B2GG B2PP	**REV**	Extended to allow multiple direct input damping matrices.

COMMAND	STAT	DESCRIPTION
BMFORCE	**NEW**	New feature for computing equivalent beam forces (moments, shears, axial loads and torques) for collections of solid elements.
BOUNDARY	**REV**	Extended for use with **AUTOREDUCE** and **NLREDUCE**
CASE	**REV**	New features to select automatic Guyan reduction or Craig-Bampton modal reduction.
EXPORT	**NEW**	New feature to export a reduced model as direct matrix input at grid points (**DMIG**).
FORCE	**REV**	Extended to support the enhanced feature to compute forces at element integration points.
IC	**REV**	Clarified to describe the use of the **EQUIL**ibrium option and its relationship to using initial conditions.
K2GG K2PP	**REV**	Extended to allow multiple direct input stiffness matrices.
M2GG M2PP	**REV**	Extended to allow multiple direct input mass matrices.
NLREDUCE	**REV**	New name for automatic reduction of nonlinear models. In previous versions, was **AUTOREDUCE**.
NLSTRAIN	**REV**	Modified to describe the enhancements for computing layer strain in composite material for Geometric Nonlinear analyses, and extension to allow strains to be calculated at the extreme fibers of plate elements, or as strains and curvatures at the midsurface of the element.
NLSTRESS	**REV**	Modified to describe the enhancements for computing layer stress in composite material for Geometric Nonlinear analyses.
NLTYPE	**REV**	Typographic correction.
OMODES	**REV**	Typographic correction.
POST	**REV**	Extended to support new **OUPTUT2** interfaces with FEMAP and **UAI** / ***RenderMaster***.
STRAIN	**REV**	Extended to support the enhanced feature for computing strains at element integration points, and typographic correction.
STRESS	**REV**	Extended to support the enhanced feature for computing stresses at element integration points.

BULK DATA PACKET

BULK DATA ENTRY	STAT	DESCRIPTION
ACCEL ACCEL1	**REV**	Expanded description of orientation vector.
BDYS BDYS1	**REV**	Extended for use with **AUTOREDUCE** and **NLREDUCE**.
BMFORCE BMFORC1	**NEW**	New feature for defining collections of solid elements for computing equivalent beam forces (moments, shears, axial loads and torques) .
CGAP	**REV**	Clarification of coordinate system definition.
DCFREQ DCMODR	**REV**	Extended to include mode tracking.
DVPROP	**REV**	Extended to include modal damping as a design variable.
EIGC	**REV**	Equations for damping computations corrected.
EIGR (Lanczos) EIGR (Givens)	**REV**	Modified to allow specification of mass orthogonality test parameter, **E**, in the Configuration File.
FORCE FORCEAX	**REV**	Expanded description of orientation vector.
FSIDATA	**REV**	Enhanced to allow computation of free-free surface modes.
GRAV	**REV**	Expanded description of orientation vector.
MOMAX MOMENT MOMENT1 MOMENT2	**REV**	Corrections and expanded description of orientation vector.
NLSOLVE	**REV**	Corrected to reflect the secant modulus solution method.
RFORCE RFORCE1	**REV**	Expanded description of orientation vector.
RLOAD1 RLOAD2	**REV**	Clarification of use of enforced motion.
SETI SETR	**NEW**	Entries that allow integer and real sets to be specified in the Bulk Data packet.
SETOP	**NEW**	Allows set operations on integer sets defined by **SETI** Bulk Data entries. Both Union and Intersection are available.
SHOCK	**REV**	Correction to reference only **TABLED1** Bulk data entries.
TLOAD1 TLOAD2	**REV**	Clarification of use of enforced motion.

DMAP MODULES

MODULE NAME	STAT	DESCRIPTION
DBIN DBOUT DBPARM	**REV**	New modules used to import and export ***eBase*** entities when using DMAP sequences. These are often used in conjunction with the new Executive Control command **ENTITY**.

This page is intentionally blank.

TABLE OF CONTENTS

LIST OF FIGURES

This page is intentionally blank.

LIST OF TABLES

Chapter 1

EXECUTING THE PROGRAM

As is the case with all major software systems that are available across a broad spectrum of host computers and operating systems†, **UAI/*NASTRAN*** has features that are implemented differently on different computers. The most common differences are in the way that you execute **UAI/*NASTRAN*** and other UAI software products, the management of dynamic memory, and the manner in which files are handled during execution.

Information describing interfaces with third-party software such as MSC/PATRAN® and SDRC I-DEAS® is found in Chapter 31 of the **UAI/*NASTRAN User's Guide***.

†All computer models and operating system names are trademarks of their respective manufacturers and vendors.

Rev: V20.1

1
JCL

1.1 OVERVIEW

This section provides you with an overview of the areas of **UAI/*NASTRAN*** that are directly affected by your host computer and its operating system.

1.1.1 Executing UAI/*NASTRAN*

The manner in which you invoke a **UAI/*NASTRAN*** execution is completely dependent on the operating system of your host computer. Subsequent sections of this chapter describe this operation for the most common host computers upon which **UAI/*NASTRAN*** is currently available. You will note that Section 1.2 includes all of the host computers using the Unix operating system and its derivatives.

1.1.2 The UAI/*NASTRAN* Configuration and Preference Files

In general, UAI's suite of engineering software products uses computing resources intensively. As a result, there are a number of parameters that must be set to achieve optimal resource management on a given host computer. These parameters, taken as a group, are called the ***Configuration*** of the products. The configuration is provided through several files. These files include parameters which are used for controlling such things as database locations, physical file characteristics, memory utilization, and algorithm control.

For maximum flexibility, configurations may be controlled both by the site, i.e. the UAI System Support Specialist for larger companies, and the end user. Many different configurations may be defined for a site or a user. For example, when configuring **UAI/*NASTRAN***, the UAI System Support Specialist may create different configurations for very small and for very large analyses.

Among the most common reasons reasons for having a customized configuration are:

- To allocate large amounts of memory and CPU time limits, by default, when always executing large analyses.
- To define the locations of file systems when databases are expected to exceed 2GB in size.
- To make engineering options (e.g. **`AUTOSPC`** and **`AUTOOMIT`**) compatible with other NASTRAN variants.
- To select comprehensive data checking options which have more stringent tests than other NASTRAN variants (e.g. element warping and aspect ratio checks).

1.1.3 Executive Control Commands

Chapter 2 of this manual presents the **UAI/*NASTRAN*** Executive Control commands. These commands provide general information to **UAI/*NASTRAN*** during execution. While the great majority of these commands are implemented in a host-independent manner, there are two commands which do depend on your host computer. The first of these, **`ASSIGN`**, is used to attach physical files on your host computer to logical files within **UAI/*NASTRAN***. The second of these commands is **`INCLUDE`**. You use this command to insert a text file into your **UAI/*NASTRAN*** input data stream. Descriptions of all host-dependent data that are required are discussed in following sections.

1.1.4 Dynamic Memory

The architecture of **UAI/*NASTRAN*** allows the modeling and analysis of finite element models of virtually unlimited size. Most numerical calculations perform at maximum efficiency when all data for the operation fits in the ***working memory*** space of the program. Many operations may be performed even when all data that they require does not fit in memory by using what is called ***spill logic***. Spill logic simply involves the paging of data to and from disk storage devices as necessary. For very large jobs, spill commonly occurs. In such cases, providing **UAI/*NASTRAN*** with additional working memory can often improve performance. On the other hand, you do not want to give **UAI/*NASTRAN*** excess memory, because it will reduce resources that could be used for other processes on your system. Under certain circumstances, excess memory may actually degrade the performance of **UAI/*NASTRAN*** and, in extreme cases, even your computer system.

UAI/*NASTRAN* has a second independent dynamic memory which is used to operate on databases that are attached to the execution. This memory is typically much smaller than the working memory. The main factor influencing the amount of database memory required is the block size used by the active databases. This is described in detail in subsequent sections.

The working memory for **UAI/*NASTRAN*** is dynamically acquired during execution. The amount of space that is actually used by the program is typically controlled by the **UAI/*NASTRAN*** execution procedure or the **MEMORY** Executive Control command. Some host computers have alternate means of controlling this memory.

1.1.5 The *eBase* Database

With **UAI/*NASTRAN*** Version 11.0, UAI introduced the Engineering Database Management System, ***eBase***, into **UAI/*NASTRAN***. This advanced scientific database technology greatly enhances the data handling capabilities of **UAI/*NASTRAN*** while removing many of the inconveniences of the older I/O system which used sequential files.

The Three Types of Databases

There are three types of ***eBase*** databases. The first is the system database. This is used by **UAI/*NASTRAN*** to store items such as error message text and database schemata definitions. The second type is the ***run-time database***, or RUNDB. This database is used to store the relations and matrices which are used in performing your analysis task. At the end of your job, the RUNDB is deleted. The third type is the ***archival database***. This type of database is saved from one execution to the next. There are three archival databases. The first is the SOF database, used in performing Substructuring Analyses, and the second is the NLDB database, used when you perform Nonlinear Material or Geometry Analyses. The third database is the Archive database which is controlled by the **ARCHIVE** Case Control command. This database may contain the geometry and solution results for your run in easy-to-use relational form. The format, or ***schema***, of these relations is described in the **UAI/*NASTRAN*** Archive Database Manual.

The Logical and Physical Views of the Database

To fully understand the database technology, you must understand the two views of the database. Each database is called a ***logical database.*** This term is used because from an engineering viewpoint, the database is a single entity which is used in its entirety. The manner in which the logical database is stored on your host computer depends on the amount of data it contains and the availability of disk storage devices. The physical view is a mapping of a logical database to some

number of physical files on your host computer. It may be necessary for you to understand the physical model because, for very large analyses, it may be more efficient to organize the actual files in a manner that allows higher performance on your host computer.

The Physical Model

Each ***eBase*** database, regardless of its use, has two components manifested as a minimum of two physical files. The first of these components is called the INDEX component. This component is always a single physical file. It contains information which identifies and locates actual database entities. These entities themselves are stored in the DATA component. To provide the maximum flexibility for a wide variety of data storage requirements, the data components may be stored in a number of different physical files. Most database systems are organized in this manner, because the index component is generally small in size and referenced often, while the data component may be extremely large and not fit in a single file or even on a single disk drive.

ASSIGNing Databases

Each logical database must be defined using the Executive Control command **ASSIGN**. The general form of the **ASSIGN** command is:

```
ASSIGN logical_name [= phys_name] [ , { NEW } ][,USE = use][,REALLOC]
                                   [ { OLD }
                                   [ { TEMP }

        [,PASSWORD = pass][,IBLKSIZE = nwib][,DBLKSIZE = nwdb]

        [,ACCESS = access][,params]
```

The description of the ASSIGN command for databases, as well as other files, is found in Chapter 2 of this manual. Of interest here are the optional *params*. The meaning and availability of these *params* depends on the **UAI/*NASTRAN*** host computer. When available, these are described for each computer beginning in Section 1.2 of this chapter.

Database File Names

The naming of database files follows a convention that is different from that of other **UAI/*NASTRAN*** files. The file names are generated automatically at execution time. The conventions used are also described starting in Section 1.2 of this chapter.

Very Large Databases

You may be solving extremely large problems with **UAI/*NASTRAN***. In such cases it may be possible that a databases exceeds the capacity of a single disk drive. **UAI/*NASTRAN*** has made provision for this and you must contact your **UAI/*NASTRAN*** System Support Specialist for details describing the use of this advanced feature.

1.1.6 The INCLUDE Files

To simplify the creation of the **UAI/*NASTRAN*** input data stream, you may insert files directly into the input stream by using the **INCLUDE** command which may appear in any of the data packets. The general syntax of this command is:

```
INCLUDE filename [,params]
```

As in the case of the **ASSIGN** command, on some host computers there are additional params which may be used. These are also described starting in Sections 1.2 of this chapter. Note that **INCLUDE** commands may appear in any position within your input data stream.

1.1.7 UAI/*NASTRAN* Import/Export Files

There are a number of file-based operations that are frequently performed when using **UAI/*NASTRAN***. These are described in the following sections.

Using the INPUTT2/OUTPUT2 and INPUTT4/OUTPUT4 Modules

UAI/*NASTRAN* provides modules with which you may import data into, or export data from, the program in a form which may be interfaced to FORTRAN programs. Typically, you use these features for pre- and post-processing of data. When using the **INPUTT2** and **OUTPUT2** modules, the files are written using FORTRAN variable length, unformatted or binary records. The **INPUTT4** and **OUTPUT4** allow you to read or write data in either of two ways. You control your selection with the parameter **TYPE**. The two types, **FORMATTED** and **BINARY**, determine the type of FORTRAN I/O used to process the file.

Either the files used by these modules must be allocated and assigned by using the **ASSIGN** Executive Control command for a logical file with a **USE** parameter specifying the appropriate module, or they may use the default parameters available under automatic **ASSIGN**ment. A detailed description of the format of these files is found in Chapter 8 of this manual.

Using the SOFIN and SOFOUT Modules

The substructuring capability within **UAI/*NASTRAN*** uses an archival database called the SOF. There are two utility operations which you may perform on an SOF database. The first of these is called **SOFOUT**. This operation is used to export an SOF database from your **UAI/*NASTRAN*** job. The second operation is called **SOFIN**. This operation allows you to import a file which contains an SOF database that was exported during a previous execution.

These operations may be performed in either of two modes. The first mode is called the **INTERNAL** format. When you create an export file using **SOFOUT** with a **TYPE** of **INTERNAL**, the file is created with FORTRAN binary I/O. It may therefore be imported into another **UAI/*NASTRAN*** execution only on the same host computer or another computer which is fully compatible with the computer on which it was created. The second mode is called the **EXTERNAL** format. When you create an export file using **SOFOUT** with a **TYPE** of **EXTERNAL**, the file is created with formatted FORTRAN I/O. In this case, it may be imported into another **UAI/*NASTRAN*** execution on any host computer. As a result, the **EXTERNAL** format can be used to transfer SOF data between one type of host computer and another. This can be done by either creating a tape on your host which will be physically loaded on the other computer, or by transferring the data directly over a network.

The NASTPLOT File

UAI/*NASTRAN* provides extensive plotting capability both for structural plots and X-Y plots of solution results. These capabilities are described in detail in Chapters 5 and 6 of this manual. When you use either, or both, of these features, you may **ASSIGN** a file with a **USE** of **PLOT**, or use the automatic **ASSIGN**ment default. In a manner similar to that described in the previous two sections,

you may select a **TYPE** of either **FORMATTED** or **BINARY** for this file. You select the **BINARY** option when your NASTPLOT post-processor program will be executed on your **UAI/*NASTRAN*** host computer. If your NASTPLOT program resides on a different computer, then you must use the **FORMATTED** option to facilitate the transfer of data from the **UAI/*NASTRAN*** host to the NASTPLOT host. In addition, UAI provides certain display capabilities for each host computer. These are described in the remainder of this Chapter.

1.1.8 Host Computer Dependencies

The sections that follow provide detailed information describing the differences in **UAI/*NASTRAN*** execution procedures and commands which depend on your host computer system.

1.2 UNIX-BASED COMPUTERS

This section describes the host-dependent information that you need to execute **UAI/*NASTRAN*** on Unix-based computer systems. UAI supports a wide variety of these computers including those manufactured by Cray, DEC, HP, IBM, SGI, Sun and others. For a complete list of platforms, please contact UAI.

1.2.1 Executing UAI/*NASTRAN*

A ***csh*** script file, called **nastran**, is provided to execute **UAI/*NASTRAN***. To execute you enter:

```
nastran [ –m  memory [ K ][ W ] ] [-ps prefname] [-pu prefname]
        [ M ][ P ]                [-pl prefname] filelist
             [ B ]
```

where *memory* specifies the amount of memory that the job will use. Options allow you to use shorthand notation for large values and allocation types. The options **K** and **M** indicate that the memory value is specified in thousands or millions of units, respectively. The units may be specified in single precision words (**W**), bytes (**B**), or machine precision words (**P**). If none of these arguments are used, then memory is assumed to be single precision words. The *prefname* specifies the substitution string used to generate preference File names. You may specify a different string for the system (**-ps**), the user (**-pu**) and the local (**-pl**) preference files. If you have the unusual case where all of these files have the same name, you may use the option **-p** followed by the *prefname*. Finally, *filelist* specifies a list of one or more file names, separated by spaces, that contain **UAI/*NASTRAN*** input data streams. The actual file names must have the proper trailing component, which is usually **.d**. The script file will execute **UAI/*NASTRAN*** using each of the data files that you provide. Examples illustrating the use of the script are shown below.

1. Execute **UAI/*NASTRAN*** using the input file **test.d**

```
nastran test
```

2. Execute **UAI/*NASTRAN*** in the background for all of the input files in directory **/uai/demodata**.

```
nastran /uai/demodata/*.d &
```

3. Execute **UAI/*NASTRAN*** using the input file **test.d** and request one million words of memory.

```
nastran -m 1000000 test or
nastran -m 1mw test or
nastran -m 1000kw test
```

4. Suppose that you have created a Preference File name **my.pref**, execute **UAI/*NASTRAN*** using the input file **test.d** using these preferences.

```
nastran -p my test
```

1.2.2 UAI/*NASTRAN* File Names

When you execute the **nastran** script a number of files may be created which have names that are automatically generated by the program. These are described in this section.

Unique UAI/*NASTRAN* files

There are four unique files that are used frequently by **UAI/*NASTRAN***. These are unique in the sense the program will automatically define file names for these if you do not explicitly **ASSIGN** them. These files, and their default names, are shown in the table below:

FILE	May Override with ASSIGN Command?	Generated Name if ASSIGN Command is Not Used
The print file	**NO**	*filename*.prt
The log file	**NO**	*filename*.log
The **BULK** file	**YES**	*filename*.bulk
The **PUNCH** file	**YES**	*filename*.pch

The *filename* represents the name of the file containing the **UAI/*NASTRAN*** input data stream. The ***log file*** is a special file that contains the history of your execution. You may monitor the progress of your job by viewing the log file periodically. Upon completion of the job, the log file is appended to the print file, and then deleted.

Databases

You will recall from Section 1.1.4 that each database that you use during an execution is comprised of at least two physical files. The trailing components of these file names is always generated by **UAI/*NASTRAN***. When you **ASSIGN** a database with a status of **NEW** and provide a physical file name, *phys_name*, the program generates the file names:

```
phys_name.edb and phys_name.00
```

There may be times, most often in the case of the **RUNDB**, that you **ASSIGN** a database with a status of **TEMP**. In such cases, the program internally generates file names that are unique to your job. The detailed rules used to generate these names are given in the ***System Support Manual***. These simple rules pertain to the simplest and most used **ASSIGN**ments of databases. If you are using very large databases, then there are additional rules. These will be provided by your **UAI/*NASTRAN*** System Support Specialist.

1.2.3 ASSIGN and INCLUDE Command Parameters

There are no additional parameters for the **INCLUDE** command for Unix-based host computers. The **ASSIGN** command has two special parameters, **ILOC** and **DLOC**, that are used to control the location of the physical files comprising a database. Contact your **UAI/*NASTRAN*** System Support Specialist for a complete description of how these parameters are used.

1.2.4 Site Definition of Automatic ASSIGN Commands

UAI/*NASTRAN* provides a capability which allows an individual client site to define a set of **ASSIGN** commands which are used automatically by the program as needed. When this feature is used, it is not necessary for you to specify your own **ASSIGN** commands if the appropriate automatic ones are available. Contact your **UAI/*NASTRAN*** System Support Specialist for a complete list of automatic **ASSIGN**s available at your site.

1.2.5 The *eShell* Program

If your site has the ***eShell*** interactive ***eBase*** interface program, then to execute this program you enter:

```
eshell [-ps prefname] [-pu prefname] [-pl prefname] [database]
```

where:

pref_name Specifies the substitution strings used to generate the Preference File names.

database Is the name of a database to be opened with read access.

This command will execute ***eShell*** in the interactive mode and, optionally, open the database that you specify with read access. As with **UAI/*NASTRAN***, *prefname* specifies the substitution string used to generate Preference File names. You may specify a different string for the system (**-ps**), the user (**-pu**) and the local (**-pl**) preference files. If you have the unusual case where all of these files have the same name, you may use the option **-p** followed by the *prefname*.

Unless directed otherwise by ***eShell*** commands, all subsequent output will be sent to the terminal device. The ***eShell*** Tutorial Problem library is available. Contact your Systems Support Specialist to obtain the name of the directory where these problems may be found. A description of how you may use them is given in the ***eShell User's Manual***.

1.2.6 Automatic Preference Files

Both the **nastran** and ***eShell*** scripts provide arguments which allow you to specify the substitution string needed to generate Preference File names. If these arguments are not used, both of these programs will look for a Preference file named **uai.pref**. By default, the UAI installation directory will be searched for this file, then for a User Preference File in your home directory, and, finally, for a Local Preference File in your current working directory. This behavior may be changed with a series of parameters that are contained in the Host Computer section. Contact your Systems Support Specialist for a complete description of how these parameters are used.

1.2.7 The Plotting Programs

Four plotting programs, **tekplot**, **nastplotps**, **nastplotgl**, and **nastplot**, are provided. **nastplotps** may be used to create files using the PostScript language, and **nastplotgl** may be used to create files using the Hewlett-Packard graphics language, HP-GL. These files may then be routed to a printer or display device. **nastplot** is an interactive X-Window program that allows you to view and print your plots. Additionally, source code is provided in the form of program **tekplot** which provides your facility with a starting point for creating your own customized plotting program.

The Tektronix PLOT10 Plot Program

A Fortran program, **tekplot**, is provided in source code format, which you may modify and use to process **UAI/*NASTRAN*** plot files and create displays on graphics terminals connected to your host computer which support the Tektronix PLOT10 graphics instructions. Contact your **UAI/*NASTRAN*** System Support Specialist for additional information.

The PostScript Plot Program

The program **nastplotps** reads both binary and formatted plot files generated by **UAI/*NASTRAN*** and generates an Encapsulated PostScript file. This PostScript output can then be either sent to a printer or imported into a text formatting program which accepts Encapsulated PostScript input. Importing the plot only makes sense when the plot file contains a single frame or if you use the **-pn** option to explicitly create a single plot. The program allows you to select fonts, control paper size and to determine output orientation (landscape or portrait). Detailed documentation on these options is available by executing the following command with no arguments:

```
nastplotps
```

The on-line help is:

```
Usage: nastplotps options file_name_1 file_name_2 ...
          -b  = plot files are binary (default)
          -f  = plot files are formatted
          -nf = suppress frame around plot
         -pn# = only plot number # is processed
         -pw# = paper width (default -pw8.5)
         -mw# = unplottable margin width (default -mw0.25)
         -ph# = paper height (default -ph11.0)
         -mh# = unplottable margin height (default -mh0.25)
         -por = portrait orientation (default)
         -lan = landscape orientation
          -tx = typeface (default -tHelvetica)
```

The output of **nastplotps** is to Unix standard output. Normally, you should redirect standard output to a file or pipe it to a print spooling program as desired. The following illustrates a typical use of **nastplotps**:

```
nastplotps -f -lan mydata.plt | lpr -Pps
```

The HP-GL Plot Program

The program **nastplotgl** reads both binary and formatted plot files generated by **UAI/*NASTRAN*** and generates HP-GL commands. This HP-GL output can then be either sent to a printer or plotter. It may also be imported into a text formatting program which accepts GL input. Importing the plot only makes sense when the plot file contains a single frame or if you use the **-pn** option to explicitly create a single plot. Detailed documentation on the plotter options is available by executing the following command with no arguments:

```
nastplotgl
```

The on-line help is:

```
Usage: nastplotgl options file_name_1 file_name_2 ...
          -b  = plot files are binary (default)
          -f  = plot files are formatted
          -nf = suppress frame around plot
         -pn# = only plot number # is processed
         -pw# = paper width (default -pw8.5)
         -mw# = unplottable margin width (default -mw0.25)
         -ph# = paper height (default -ph11.0)
         -mh# = unplottable margin height (default -mh0.25)
```

The output of **nastplotgl** is to Unix standard output. Normally, you should redirect standard output to a file or pipe it to a print spooling program as desired.

The X-Window, Motif Interface Plot Program

For computer systems which support the X-Window system, the plotting program **nastplot** is provided. This program, which operates in the X-Window environment, uses a Motif interactive interface. **nastplot** provides the following functional capability for viewing and processing **UAI/*NASTRAN*** plot files:

- ❒ Automatic recognition and processing of binary or formatted plot files.
- ❒ Full support of the **LINESTYLE** command using user selectable display colors.
- ❒ Direct selection of display for any plot in the plot file.
- ❒ Zooming of the plot display.
- ❒ Export of plots to either a printer or a file, using either PostScript or HP-PCL display languages.

nastplot is executed with the command:

```
nastplot [ file_name ]
```

Detailed online help is provided by the **nastplot** program.

Special Versions of the nastplot Program

On HP/Apollo and Sun workstations special versions of **nastplot** are delivered which operate under the normal window system found on those computers, Display Manager and SunView, respectively.

1.2.8 Online Manuals

The entire suite of **UAI/*NASTRAN*** manuals is available online in the Adobe Portable Document Format (PDF). This allows you to view the documentation on any computer that has the Adobe® Acrobat® Reader 3.0. Readers for DEC, HP, IBM, MAC, PC, SGI, and Sun (OS and Solaris) were delivered with your system. Any other readers that become available can be downloaded from the Adobe Web site at **www.adobe.com**.

To use the documents, from the command line you enter:

```
uaidoc [manual_name]
```

If you omit the *manual_name*, then you will see a splash screen that allows you to navigate to the appropriate manual. You may also go directly to a manual by placing its name on the command lines. The names of the **UAI/*NASTRAN*** manuals are:

- ***Nastran_Reference_Manual***
- ***Nastran_Users_Guide***
- ***Nastran_Schemata_Manual***
- ***UAI_Unix_Support_Manual***

Check the UAI Web site at **www.uai.com** for any interim updates and additions to the electronic documentation.

1.3 DEC VAX SERIES COMPUTERS — VMS OPERATING SYSTEM

This section describes the host-dependent information that you need to execute **UAI/*NASTRAN*** on DEC VAX computers under the VMS operating system.

1.3.1 Executing UAI/*NASTRAN*

A command procedure, called **NASTRAN**, is provided to execute the program. To execute you enter:

```
NASTRAN { filename    }[/WSL=n][/AFT=time][ /PRT= { NO  } ] -
        { JID=filename}                           { YES }

        [/SPREF=pref_file][/UPREF=pref_file][/LPREF=pref_file]

        [/T=cpu_time][/QUE=queuename]
```

where:

filename — Specifies the file name that contains the **UAI/*NASTRAN*** input data stream. The actual file specifications must have a type of **.DAT**. If you omit the *filename*, you will be prompted for it.

n — Specifies the working set size limit for the run.

time — Specifies the time-of-day at which the **UAI/*NASTRAN*** execution will begin. The general form of *time* is:

dd-mmm-yyyy:hh:mm:ss.ss

Refer to your VAX/VMS DCL Dictionary Manual for a complete description of the format.

{ **NO** / YES } — Requests that the output and log files be saved or printed and deleted. If the default value, **NO**, is selected, then the files *filename*.**PRT** and *filename*.**LOG** will be saved in your directory. If you select **YES**, then both files will be printed on your line printer and then deleted.

pref_file — Preference file substitution strings.

cpu_time — Specifies the CPU time limit for the execution in the form:

hh:mm:ss

queuename — Specifies the name of the batch queue into which the job will be placed.

The *pref_file* specifies the substitution string used to generate Preference File names. You may specify a different string for the system (**SPREF**), the user (**UPREF**) and the local (**LPREF**) preference files. If you have the unusual case where all of these files have the same name, you may use the option **PREF** followed by the *pref_file*. This procedure submits a batch job which executes **UAI/*NASTRAN*** using the specified *filename*. Some of the results of this execution may create

output files. These are described in the next section. The keyword parameters used by the procedure may be specified in any order, but they must be separated by the slash character, "/", a comma, or a blank. Consider the following examples:

1. Execute **UAI/*NASTRAN*** for the input data stream contained in file **TEST.DAT**:

```
NASTRAN TEST
```

2. Submit a batch job that will execute **UAI/*NASTRAN*** using the file **TEST.DAT** after 10:00 PM and request that the output print file be printed and then deleted.

```
NASTRAN TEST/AFT=22:00:00/PRT=YES
```

1.3.2 UAI/*NASTRAN* File Names

When you execute the **NASTRAN** script a number of files may be created which have names that are automatically generated by the program. These are described in this section.

Unique UAI/*NASTRAN* files

There are four unique files that are used frequently by **UAI/*NASTRAN***. These are unique in the sense the program will automatically define file names for these if you do not explicitly **ASSIGN** them. These files, and their default names, are shown in the table below:

FILE	May Override with ASSIGN Command?	Generated Name if ASSIGN Command is Not Used
The print file	**NO**	*filename*.PRT
The UAI log file	**NO**	*filename*.SUM
The **BULK** file	**YES**	*filename*.BULK
The **PUNCH** file	**YES**	*filename*.PCH

The *filename* represents the name of the file containing the **UAI/*NASTRAN*** input data stream. The UAI ***log file*** is a special file that contains the history of your execution. Upon completion of the job, the log file is appended to the print file, and then deleted. VMS jobs also generate a log file which is named *filename*.LOG. Depending on the options specified at your site, this file may also contain information similar to that found in the UAI log file.

Databases

You will recall from Section 1.1.4 that each database that you use during an execution is comprised of at least two physical files. The trailing components of these file names is always generated by **UAI/*NASTRAN***. When you **ASSIGN** a database with a status of **NEW** and provide a physical file name, *phys_name*, the program generates the file names:

```
phys_name.EDB and phys_name.DF000
```

There may be times, most often in the case of the **RUNDB**, that you **ASSIGN** a database with a status of **TEMP**. In such cases, the program internally generates file names that are unique to your job. The detailed rules used to generate these names are given in the ***System Support Manual***. These simple rules pertain to the simplest and most used **ASSIGN**ments of databases. If you are using very large databases, then there are additional rules. These will be provided by your **UAI/*NASTRAN*** System Support Specialist.

1.3.3 Monitoring the Execution

There are two ways that you may monitor the progress of your batch job. The process name field of the VMS command

```
SHOW SYSTEM/BATCH
```

will display the filename being executed and the **UAI/*NASTRAN*** module that is currently being executed. If allowed by your **UAI/*NASTRAN*** System Support Specialist, you may also examine the batch log file which provides a history of the **UAI/*NASTRAN*** modules as they execute.

1.3.4 ASSIGN and INCLUDE Command Parameters

There are no additional parameters to the **INCLUDE** command for VAX/VMS host computers.

The **ASSIGN** Executive Control command has several parameters. The **NEWVER** parameter complements the **REALLOC** parameter in controlling the manner in which **UAI/*NASTRAN*** handles duplicate file when the **ASSIGN** command has a disposition of **NEW**. The **REALLOC** option causes the latest version of the existing file to be deleted and reallocated by the job. The **NEWVER** option, on the other hand, causes the existing version to be kept and a new version created. If you do not specify one of these options, your job will fail if a specified file already exists. Two other special parameters, **ILOC** and **DLOC**, are used to control the location of the physical files comprising a database. Contact your **UAI/*NASTRAN*** System Support Specialist for a complete description of how these parameters are used.

Examples of **ASSIGN** commands are given below:

1. Assign a file in your **UAI/*NASTRAN*** data stream which has the file name **DB1:[A]WING.OP2**. Assume the file, which is new, will be used to export data using the **OUTPUT2** module and you wish to use the logical name **OP2** for the file.

```
ASSIGN OP2=DB1:[A]WING.OP2,NEW,USE=OUTPUT2,REALLOC
```

2. Perform the same operation as above, but assume that you are saving different **OUTPUT2** results as different versions of the same file. Then the following **ASSIGN** command is used:

```
ASSIGN OP2=DB1:[A]WING.OP2,NEW,USE=OUTPUT2,NEWVER
```

If you already had two versions of this file, **WING.OP2;1** and **WING.OP2;2**, then the **OUTPUT2** results from the current execution will be placed in the file:

```
WING.OP2;3
```

1.3.5 Site Definition of Automatic ASSIGN Commands

UAI/*NASTRAN* provides a capability which allows an individual client site to define a set of **ASSIGN** commands which are used automatically by the program as needed. When this feature is used, it is not necessary for you to specify your own **ASSIGN** commands if the appropriate automatic ones are available. Contact your **UAI/*NASTRAN*** System Support Specialist for a complete list of automatic **ASSIGN**s available at your site.

1.3.6 Dynamic Memory

You generally define the amount of dynamic memory to be used by your **UAI/*NASTRAN*** job by using the **MEMORY** Executive Control command. Additionally, you may specify a working set limit, **WSL**, when you invoke the command procedure. This is an advanced feature that may impact the performance of your host computer. Contact your **UAI/*NASTRAN*** System Support Specialist for complete details.

1.3.7 The *eShell* Program

If your site has the ***eShell*** interactive ***eBase*** interface program, then to execute this program you enter:

```
eshell [ database ]
```

where:

database Is the name of a database to be opened with read access.

This command will execute ***eShell*** in the interactive mode and, optionally, open the database that you specify with read access. Unless directed otherwise by ***eShell*** commands, all subsequent output will be sent to the terminal device. The ***eShell*** Tutorial Problem library is available. Contact your Systems Support Specialist to obtain the name of the directory where these problems may be found. A description of how you may use them is given in the ***eShell*** User's Manual.

1.3.8 The Plotting Programs

Three plotting programs, **TEKPLOT**, **nastplotps**, and **nastplotgl**, are provided. **nastplotps** may be used to create files using the PostScript language, and **nastplotgl** may be used to create files using the Hewlett-Packard graphics language, HP-GL. These files may then be routed to a printer or display device. Additionally, source code is provided in the form of program **TEKPLOT** which provides your facility with a starting point for creating your own customized plotting program.

The Tektronix PLOT10 Plot Program

A Fortran program, **TEKPLOT**, is provided in source code format, which you may modify and use to process **UAI/*NASTRAN*** plot files and create displays on graphics terminals connected to your host computer which support the Tektronix PLOT10 graphics instructions. Contact your **UAI/*NASTRAN*** System Support Specialist for additional information.

The PostScript Plot Program

The program **NASTPLOTPS** reads both binary and formatted plot files generated by **UAI/*NASTRAN*** and generates an Encapsulated PostScript file. This PostScript output can then be either sent to a printer or imported into a text formatting program which accepts Encapsulated PostScript input. Importing the plot only makes sense when the plot file contains a single frame or if you use the **/PN** option to explicitly create a single plot. The program allows you to select fonts, control paper size and to determine output orientation (landscape or portrait).

A command procedure called **NASTPLOTPS** is provided to allow you to create these PostScript files. To execute, you enter:

```
NASTPLOTPS [/TYPE = {BINARY  }] [/ORIENT = {PORTRAIT  }] [ /NOFRAME ]
                   {FORMAT  }              {LANDSCAPE }
        [/PN = #][/PW = #][/MW = #][/PH = #][/MH = #]
        [TYPEFACE=tf][/OUTPUT=filename] plotfile_1 plotfile_2  ...
```

where:

{BINARY / FORMAT}	Specifies the plot file format. The default is BINARY.
{PORTRAIT / LANDSCAPE}	Specifies the paper orientation. The default is PORTRAIT.
/NOFRAME	Suppresses the frame around the plot.
/PN=#	Processes single plot with sequence number #.
/PW=#	Sets paper width. The default is 8.5 in.
/MW=#	Specifies unplottable margin width. Default 0.25 in.
/PH=#	Specifies paper height. The default 11.0 in.
/MH=#	Specifies unplottable margin height. The default 0.25 in.
/TYPEFACE=*tf]*	Selects a PostScript typeface. The default is Helvetica.
/OUTPUT=*filename*	Specifies the file name which will contain the resulting Encapsulated PostScript file. If omitted, output is routed to **SYS$OUTPUT**.
plotfile_i	Specifies the file names which contains your plot files created by a **UAI/*NASTRAN*** job.

Normally you should redirect **SYS$OUTPUT** to a file. This file may then be routed to the PostScript printer, if available, at your facility.

The HP-GL Plot Program

The program **NASTPLOTGL** reads both binary and formatted plot files generated by **UAI/*NASTRAN*** and generates HP-GL commands. This HP-GL output can then be either sent to a printer or plotter. It may also be imported into a text formatting program which accepts GL input. Importing the plot only makes sense when the plot file contains a single frame or if you use the **/PN** option to explicitly create a single plot. A command procedure called **NASTPLOTGL** is provided to allow you to create these files. To execute, you enter:

```
NASTPLOTGL [/TYPE = { BINARY } ]  [ /NOFRAME ] [/PN = #][/PW = #]
                   { FORMAT }
           [/MW = #][/PH = #][/MH = #] [/OUTPUT=filename ]
           plotfile_1  plotfile_2 ...
```

where:

{ BINARY } { FORMAT }	Specifies the plot file format. The default is **BINARY**.
/NOFRAME	Suppresses the frame around the plot.
/PN=#	Processes single plot with sequence number #.
/PW=#	Sets paper width. The default is 8.5 in.
/MW=#	Specifies unplottable margin width. The default 0.25 in.
/PH=#	Specifies paper height. The default 11.0 in.
/MH=#	Specifies unplottable margin height. The default 0.25 in.
/OUTPUT= *filename*	Specifies the file name which will contain the resulting Encapsulated PostScript file. If omitted, output is routed to **SYS$OUTPUT**.
plotfile_i	Specifies the file names which contains your plot files created by a **UAI/*NASTRAN*** job.

Normally you should redirect **SYS$OUTPUT** to a file. This file may then be routed to the PostScript printer, if available, at your facility.

1.3.9 Online Manuals

The entire suite of **UAI/*NASTRAN*** manuals is available online in the Adobe Portable Document Format (PDF). This allows you to view the documentation on any computer that has the Adobe® Acrobat® Reader 3.0. Readers for DEC, HP, IBM, MAC, PC, SGI, and Sun (OS and Solaris) were delivered with your system. Any other readers that become available can be downloaded from the Adobe Web site at **www.adobe.com**.

To use the documents, from the command line you enter:

```
uaidoc [manual_name]
```

If you omit the *manual_name*, then you will see a splash screen that allows you to navigate to the appropriate manual. You may also go directly to a manual by placing its name on the command lines. The names of the **UAI/*NASTRAN*** manuals are:

- ***Nastran_Reference_Manual***
- ***Nastran_Users_Guide***
- ***Nastran_Schemata_Manual***
- ***UAI_VMS_Support_Manual***

Check the UAI Web site at **www.uai.com** for any interim updates and additions to the electronic documentation.

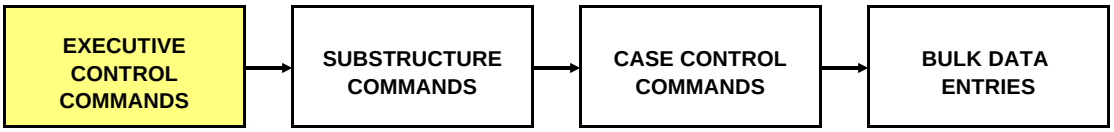
This page is intentionally blank.

Chapter 2

EXECUTIVE CONTROL COMMANDS

Executive Control commands provide **UAI/*NASTRAN*** with general information describing the analysis to be performed. The Executive Control command packet is the first in the input data stream as shown in Figure 2-1.

Figure 2-1. EXECUTIVE CONTROL PACKET LOCATION



In addition to Executive Control commands, this data packet may contain one or more subpackets that are used to modify Rigid Formats, to restart a previous analysis or to create a specialized DMAP program.

Rev: Ver20.1

2.1 THE EXECUTIVE CONTROL COMMANDS

The general form of the Executive Control command packet is:

```
command option_list
command option_list
   ...
[RESTART Subpacket]
[ALTER Subpacket] or [DMAP Subpacket]
CEND
```

The packet contains a number of *commands*, most of which have an additional *option_list*. Three optional subpackets may also be present. These are the DMAP Subpacket, the RESTART Subpacket, and the ALTER Subpacket, all of which are described in this chapter. The available commands are summarized in Table 2-1.

Table 2-1. SUMMARY OF EXECUTIVE CONTROL COMMANDS

COMMAND	DESCRIPTION
$	Indicates a comment entry.
ALTER	Edits a Rigid Format DMAP sequence.
APPROACH	Selects the analysis approach or discipline.
ASSIGN	Assigns physical file names.
BEGIN	Defines the start of a DMAP sequence.
CEND	Indicates the end of the Executive Control packet.
CHKPNT	Selects the CHECKPOINT option.
DATACHECK	Enables specific data checking features.
DIAG	Selects diagnostic output.
ECHOON and ECHOOFF	Enables and disables the echoing of Executive Control commands.
END	Defines the end of a DMAP sequence subpacket
ENDALTER	Defines the end of a Rigid Format ALTER packet.
ID	Defines the start of the Executive Control packet.
IMPORT	Imports a Rasna Appled Structure model.
INCLUDE	Inserts the contents of an external file into the input data stream.
LINES	Specifies the number of printed lines on a page.
MAXLINES	Limits the total number of lines of print.
MEMORY	Specifies the amout of memory **UAI/*NASTRAN*** will use.
PRINT CONFIGURATION	Requests a printed echo of your current Configuration.
SEQUENCE	Controls the GRID point sequencer.
SKIPON and SKIPPOFF	Disables or enables the processing of Executive Control commands.
SOLUTION	Selects a Rigid Format.
SOLVER	Controls the use of the Sparse Matrix Solver for static analysis.
TIME	Specifies the maximum allowable execution time.

2.2 EXECUTIVE CONTROL SUBPACKETS

There are three optional subpackets that may be included in the Executive Control packet:

- ❒ The ALTER Subpacket
- ❒ The DMAP Subpacket
- ❒ The RESTART Subpacket

Each of these is described below and examples of their use are given. However, first a word about some nomenclature in this manual.

Note that **UAI/*NASTRAN*** creates an output file, called the ***Print File***, which is formatted and suitable for routing to a standard, 132 character wide printer. At most installations, this file is not automatically routed to the printer. You may review the contents of this file using your standard text editor, and then dispose of the file as you wish. Thus, the generation of output for this file is often referred to as a ***Print Operation***. Also, an alternate form of output is available which once requested the generation of physical punched computer cards. Today, however, this output is written to a logical file that you have assigned with the option **USE=PUNCH**. This file is always formatted in 80 byte records and it may also be examined with a text editor.

2.2.1 The ALTER Subpacket

To modify a standard Rigid Format, for instance to print some intermediate results, an ALTER subpacket is used. The **ALTER** commands allow the deletion or insertion of new DMAP statements into the Rigid Format. For example, to print the stiffness matrix used for a static analysis, an ALTER subpacket might be:

```
ALTER 20
MATPRN KAA,,,,// $
ENDALTER
```

This subpacket may appear anywhere within the Executive Control packet. The ALTER subpacket always begins with an initial **ALTER** command. As many **ALTER** commands may be used as desired, but the DMAP statement numbers must appear in increasing sequence number. The **ALTER** commands are of the form:

```
ALTER i
or
ALTER j,k
```

In the first case, *i* is a DMAP statement number and the following DMAP statements will be inserted after statement *i*. In the second case, existing Rigid Format statements numbered *j* through *k* will be replaced by the DMAP statements which follow the **ALTER** command.

Note that only one ALTER subpacket may appear in the Executive Control packet. This means that **ALTER** commands may not be interspersed with other Executive Control commands.

The ALTER subpacket may be terminated by either an **ENDALTER** command or the **CEND** command. Thus, the ALTER subpacket is often placed at the end of the Executive Control packet and **CEND** is used as the terminator.

Chapter 8 contains documentation for DMAP statements which may be used in the ALTER subpacket. To get the DMAP listings for the Rigid Formats you use the Executive Control command **DIAG,14**. Finally, note that the **MULTI** Solution Sequence may not be modified by the user.

2.2.2 The DMAP Subpacket

To use the Direct Matrix Abstraction Program (DMAP) feature of **UAI/*NASTRAN***, detailed in Chapter 8, the **APP DMAP** Executive Control command is selected. A DMAP subpacket is then entered into the Executive Control packet. This subpacket must have the form:

```
BEGIN $
   ...
   ...
   ...
END $
```

This subpacket must also appear prior to the **CEND** command. Only one DMAP subpacket may be included in the Executive Control command packet.

2.2.3 The RESTART Subpacket

The RESTART subpacket consists of commands which are automatically generated by a previous **UAI/*NASTRAN*** execution in which a checkpoint operation was requested. The CHECKPOINT execution created these commands and stored them on the **UAI/*NASTRAN*** logical file which is assigned with **USE=DICT**. These commands are a table of contents of the RESTART File, which is a logical file that you have assigned with **USE=CHKPNT**, which was also created in this previous execution.

To perform a RESTART using your previous CHECKPOINT results, the RESTART subpacket must be inserted into the Executive Control packet along with an **ASSIGN** command for the RESTART File. The RESTART File, which is the same physical file that you used for the CHECKPOINT, is now assigned with the option **USE=RESTART**. The RESTART subpacket is normally inserted into the Executive Control packet by using the **INCLUDE** command.

An example of the proper use of these commands is shown next. First, the following Executive Control commands perform the CHECKPOINT operation:

```
ID CHECKPOINT,RUN
   APP DISP
   SOL 1
   CHKPNT YES
   ASSIGN DICT=DICT.DAT,NEW,USE=DICT
   ASSIGN NPTP=RESTARTFILE,NEW,USE=CHKPNT
CEND
```

Then these commands perform the RESTART operation:

```
ID RESTART,RUN
   APP DISP
   SOL 1
   ASSIGN OPTP=RESTARTFILE,OLD,USE=RESTART
   INCLUDE DICT.DAT
CEND
```

Note that the status of the RESTART File is **OLD**, indicating that it already exists. Also note that the **MULTI** Solution Sequence may not be restarted.

2.2.4 Configuration Parameters

As indicated in Chapter 1, **UAI/*NASTRAN*** is configured to perform optimally on a given host computer. This is done by using Configuration Files. Some of the items in the **[UAI/NASTRAN]** Component of the Configuration File control parameters are found in Executive Control commands. When this is the case, the item is noted for the specific command.

2.3 EXECUTIVE CONTROL COMMAND DESCRIPTIONS

Executive Control commands are free-field entries. In presenting the general syntax of each command and its options, the following conventions are used:

- Many Executive Control commands contain ***keywords***. All keywords are shown in capitalized, bold computer type such as: **ALTERLIB**. All such keywords must be entered exactly as they are specified, subject only to the exception that they may be abbreviated by their first four characters.
- Some commands contain ***parentheses***. These must be entered if an option requiring them is selected.
- Lower case italicized computer type, such as *altername*, indicates that you must provide a specific data value.
- Braces { } enclose a list of two or more options from which you may select one.
- Brackets [ ] indicate that the enclosed keywords and parameters may be omitted when you use the command.
- When a choice is to be made from a list, the default choice is presented in boldface type as in the following: { CHOICE1 / CHOICE2 / **DEFAULT** }
- A single command line may not exceed 72 characters. However, you may continue a command by ending the current record with a comma and continuing to the next record.

Additionally, command options have a valid data range and, in some cases, a default value. The following table defines the data range specifications found in this chapter.

DATA RANGE SPECIFIER	MEANING
Integer	The data must be an integer number in the range of indicated values.
Real	The data must be a real number in the range of indicated values.
Character	The data must be a string of characters beginning with a letter.
Keyword	The data must be a character string matching the first four or more unique characters of a keyword option.

Executive Control Command $ [Comment]

Defines the beginning of commentary text.

Command Syntax:

```
$ any_character_string
```


Examples:

1. Enter a comment line in the Executive Control packet:

```
APP DISP
$ EXECUTE A STATICS ANALYSIS
SOL 1
```

2. Place a comment after an Executive Control command:

```
TIME = 10 $ SPECIFY A TIME LIMIT RATHER THAN USE THE DEFAULT VALUE
```

Executive Control Command ALTER

Modifies the Direct Matrix Abstraction Program (DMAP) sequence of a Rigid Format.

Command Syntax:

```
ALTER line1  [ , line2 ]
```

Option	*Meaning*	*Data Range*
line1	Specifies a DMAP instruction statement number which, if *line2* is absent, defines a statement after which the ensuing DMAP commands are entered.	Integer>0
line2	Specifies a second DMAP statement number which, with *line1*, defines a range of instructions to be deleted and replaced by any following DMAP instructions.	Integer>0 *line2*≥*line1*

Remarks:

1. Successive **ALTER** commands must reference statement numbers which are in increasing order.
2. The ALTER subpacket is terminated by either an **ENDALTER** or **CEND** command.

Examples:

1. Insert DMAP statements after line 22:

```
ALTER 22
...
```

2. Delete line 5 of the selected Rigid Format:

```
ALTER 5,5
```

3. Replace lines 38 through 45 with one or more new DMAP statements.

```
ALTER 38,45
...
```

Executive Control Command APPROACH

Selects the analytical approach for the **UAI/*NASTRAN*** execution.

Command Syntax:

```
APP { DISPLACEMENT }
    { NONLINEAR   }
    { HEAT        }
    { DMAP        }
```

Option	*Meaning*	*Data Range*
DISPLACEMENT	Specifies that one of the Linear Structural Analysis Rigid Formats, or the Multidisciplinary Solution sequence, will be used.	Keyword
NONLINEAR	Specifies that a Nonlinear Structural Analysis Rigid Format will be used.	Keyword
HEAT	Specifies that one of the Heat Transfer Rigid Formats will be used.	Keyword
DMAP	Specifies that a Direct Matrix Abstraction Program (DMAP) will be provided by the user.	Keyword

Remarks:

1. The following table summarizes the **APP** command that is used in performing different types of analyses with **UAI/*NASTRAN***:

ANALYSIS TYPE	APP COMMAND	NOTE
Linear Structural Analysis Rigid Formats and Multidisciplinary Solution Sequence	APP DISP	Optional
Nonlinear Structural Analysis Rigid Formats	APP NONLINEAR	Required
Heat Transfer Rigid Formats	APP HEAT	Required
User-written DMAP program	APP DMAP	Required
User-written ALTERs that replace the automatically generated Substructuring DMAP Solutions [2]	APP DISP,SUBS,1	Required [3]

2. See Chapter 19 of the ***User's Guide*** for examples of using ALTERs with the Substructuring capability.

3. This form is only necessary if there is a conflict between your ALTERs and the Substructuring ALTERs. If this is the case, the command is used to remove all of the automatically generated ALTERs. As a result, you must provide all of the DMAP.

Rev: V20.1

Executive Control Command ASSIGN

Selects **UAI/*NASTRAN*** data files prior to execution.

Command Syntax:

For Databases:

ASSIGN *logical_name* [= *phys_name*] [,{ NEW / **OLD** / TEMP }][,USE=*use*][,REALLOC]

[,PASSWORD = *pass*][,IBLKSIZE = *nwib*][,DBLKSIZE = *nwdb*]

[,ACCESS = { READ / WRITE / **ADMIN** }] [,*params*]

For Other Files:

ASSIGN *logical_name* [= *phys_name*] [,{ NEW / **OLD** / TEMP }][,USE = *use*][,REALLOC]

TYPE = [,{ **BINARY** / FORMATTED }] [,BLKSIZE = *nwblk*][,*params*]

Option	*Meaning*	*Data Range*
logical_name	Defines a logical **UAI/*NASTRAN*** file name. [1]	Character
phys_name	Is the operating system dependent physical file name that you wish assigned to the *logical_name* file. If not provided, then *logical_name* is used.	[2]
{ NEW / **OLD** / TEMP }	Defines the status of the file. The file may be **NEW**, in which case it is allocated at run-time, an existing or **OLD** file, which is the default, or a **TEMP** file which is deleted at the end of the run. [3]	Keyword
use	Defines how the assigned file will be used. If not provided, then *logical_name* is used.	[4]
REALLOC	Requests that a ***new*** physical file be ***reallocated*** if it already exists. [5]	Keyword
pass	Specifies the database password. [6]	Character

nwib	Index file blocksize, in words. Used for databases only. [7]	Integer>0
nwdb	Data file blocksize, in words. Used for databases only. [7]	Integer>0
{ READ / **WRITE** / ADMIN }	Specifies the type of access that will be allowed for a database with a status of **OLD**. [6]	Character
params	Host computer dependent parameters.	[10]
{ **BINARY** / FORMATTED }	Specifies the type of output that will be written on the file. [8,9]	Character
nwblk	Sequential file blocksize, in words. [7]	Integer>0

Remarks:

1. The **ASSIGN** command is optional, it overrides any default assignments made in the **<Assign Processing>** group of the **[UAI/NASTRAN]** Configuration Section. You may use the command:

   PRINT CONFIGURATION

   to obtain the defaults. Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

2. The logical file name *logical_name* is used in DMAP instructions which you use to create files that will be exported from, or imported into, **UAI/*NASTRAN***, and it is also used directly in certain other commands. Otherwise it may be one of the *use* names, removing the need for the specification of the *use* parameter. See the examples below.

2. Physical file names vary depending on your host computer. See Chapter 1 for details.

3. Additional options may be available for different host computers. See Chapter 1.

4. The table below summarizes the available **UAI/*NASTRAN*** file *uses:*

USE	FILE CONTAINS OR WILL CONTAIN:	USED WHEN:
ARCHIVE	ARCHIVE Database	When requesting data archival with the **ARCHIVE** Case Control command.
BULK	Bulk Data **PUNCH** requests	Any Bulk Data **PUNCH** requests are made.
CHKPNT	CHECKPOINT information	Performing a **CHECKPOINT** execution.
DICT	The CHECKPOINT dictionary	Performing a **CHECKPOINT** execution.
IMPORT	**UAI/*NASTRAN*** ARCHIVE Database	Importing an existing model.
INPUTT1 INPUTT2 INPUTT3 INPUTT4 INPUTT5	**UAI/*NASTRAN*** Import File	Reading files that were created in earlier **UAI/*NASTRAN*** jobs by using modules **OUTPUT1**, **OUTPUT2**, **OUTPUT4** or **OUTPUT5**, or were created by an external FORTRAN program.
NLDB	Nonlinear Database	Performing Nonlinear Analyses.
OUTPUT1 OUTPUT2 OUTPUT4 OUTPUT5	**UAI/*NASTRAN*** Export File	Writing files using modules **OUTPUT1**, **OUTPUT2**, **OUTPUT4**, or **OUTPUT5**.

USE	FILE CONTAINS OR WILL CONTAIN:	USED WHEN:
PLOT	**UAI/*NASTRAN*** plot directives	**OUTPUT(PLOT)**, **OUTPUT(XYPLOT)**, or Substructuring **PLOT** commands are present.
PUNCH	Data for which **PUNCH** is requested	Any **PUNCH** requests are made for any solution results.
RESTART	RESTART information	**RESTART**ing a previous job. You must also **INCLUDE** the **DICT** file from the **CHECKPOINT** run.
RUNDB	Run-time Database	Running any **UAI/*NASTRAN*** job.
SOF	Substructure Database	Performing Substructuring Analyses.
SOFIN	Substructure Database	Importing an SOF Substructure database that was previously exported with module **SOFOUT**.
SOFOUT	Substructure Database	Exporting the SOF Substructure database.

5. If you specify **NEW** for a file that already exists, and you do not include the **REALLOC** parameter, your job will be terminated.

6. Passwords are used, but they are not required, only when **USE** is **RUNDB**, **ARCHIVE**, **SOF** or **NLDB**. For databases with a **STATUS** of **NEW**, the same password is used for the **READ**, **WRITE** and **ADMIN** privileges. The ***eShell*** command:

   SET PASSWORD

   may be used to change any or all of the passwords as desired. For **OLD** databases, the password must match the access type specified by the **ACCESS** parameter.

7. You may provide default value for this command in the **<Computing Resources>** group of the **[eBase:applib]** Section of your Configuration.

8. This parameter is only needed when the file **USE** is **PLOT**, **INPUTT4**, **INPUTT5**, **OUTPUT4**, **OUTPUT5**, **SOFIN**, and **SOFOUT**.

9. When a **BINARY** type is specified, you may combine output from modules OUTPUT2 and OUTPUT4 on the same file.

10. Other host computer dependent parameters may be available. See Chapter 1 for details.

Examples:

---

1. Any of the following **ASSIGN** commands may be used to define a new OUTPUT2 file to be saved for post-processing:

```
ASSIGN USER1=MYFILE,NEW,USE=OUTPUT2,REALLOC

ASSIGN OUTPUT2=MYFILE,NEW,REALLOC

ASSIGN OUTPUT2=MYFILE,NEW,USE=OUTPUT2,REALLOC

ASSIGN MYFILE,NEW,USE=OUTPUT2,REALLOC
```

2. You may use any of the following commands to create a new Substructure Database:

```
ASSIGN SOF=MYSOF,NEW,REALLOC

ASSIGN MYSOF,NEW,USE=SOF,REALLOC
```

Executive Control Command BEGIN

Defines the beginning of a DMAP Subpacket and selects compiler options.

Command Syntax:

```
BEGIN [ { GO   } ] [ , ERR = level ] [ , { NOLIST } ] [ , { NODECK } ]
      [ { NOGO } ]                     [ { LIST   } ]   [ { DECK   } ]

                                       [ , { NOREF } ] [ , { NOWARN } ] $
                                         { REF   }       { WARN   }
```

Option	*Meaning*	*Data Range*
{ **GO** } { NOGO }	Requests DMAP program compilation and execution, or termination of **UAI/*NASTRAN*** after compilation.	Keyword
ERR=*level*	Defines the error *level* at which suspension of execution will occur. *level=0* selects the warning error level, *level=1* the potentially fatal error level, and a *level=2* the fatal error level. [1]	0≤Integer≤2 Default=2
{ LIST } { **NOLIST** }	Selects or deselects DMAP program listing option.	Keyword
{ DECK } { **NODECK** }	Selects or deselects writing the DMAP program to the PUNCH file. [2]	Keyword
{ REF } { **NOREF** }	Selects or deselects creating a DMAP cross-reference listing.	Keyword
{ WARN } { **NOWARN** }	Selects or deselects warning message print. If **WARN** is selected, then warning and potentially fatal error messages are printed, otherwise they are not. [1]	Keyword

Remarks:

1. If you ALTER a Rigid Format, then the **NOWARN** default is changed to **WARN**.
2. You must use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **DECK** option.
3. All Rigid Formats start with a **BEGIN** command. To request other than the default compiler options, you replace the **BEGIN** command. For example:

```
ALTER 1,1
BEGIN NOGO, REF $
```

Executive Control Command CEND

Defines the end of the Executive Control command packet.

Command Syntax:

```
CEND
```


Remarks:

1. This command is mandatory and must be the last input record in the Executive Control command packet.

Executive Control Command CHKPNT

Requests the creation of a CHECKPOINT file which can be used later during a RESTART execution.

Command Syntax:

```
CHKPNT  { YES }
        { NO  }
```


Remarks:

1. Unless defaults are provided in your Configuration, you must use the **ASSIGN** Executive Control command to define two logical files, one with **USE=DICT** and one with **USE=CHKPNT** if you use this command.
2. The RESTART dictionary, contained on the logical file assigned with **USE=DICT**, is used in subsequent RESTARTs.

   For example, the following commands could be used to perform a CHECKPOINT operation:

```
ASSIGN CHKPNT=MY_CHK_FILE,NEW,USE=CHKPNT,REALLOC
ASSIGN DICT=MY_DICT_FILE,NEW,USE=DICT,REALLOC
CHKPNT YES
```

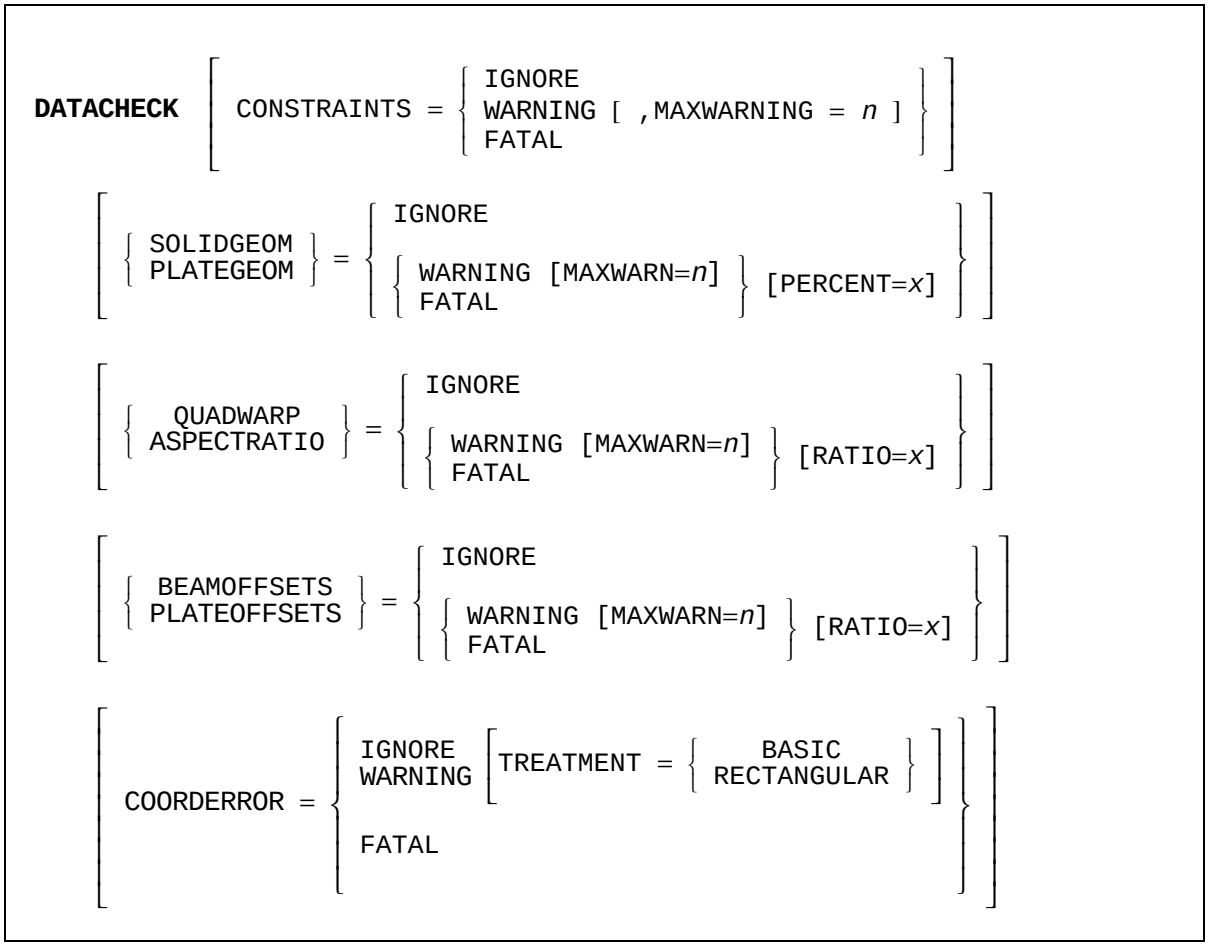
   And then these commands would be used to perform a RESTART execution:

```
ASSIGN RESTART=MY_CHK_FILE,USE=RESTART
INCLUDE MY_DICT_FILE
```

Executive Control Command DATACHECK

Specifies parameters which control how **UAI/*NASTRAN*** checks the validity of certain user input or finite element model data.

Command Syntax:



Option	*Meaning*	*Data Range*
{ IGNORE / WARNING / FATAL }	Selects the treatment of possible user input errors of the selected type. [1]	Character
n	Specifies the maximum number of warning messages to be printed for the selected option.	Integer
x	Meaning depends on the **DATACHECK** option. [2]	Real
{ RECTANGULAR / BASIC }	Specifes the type of treatment to correct GRID point coordinates located on the z-axis of non-rectangular systems.	Character

Remarks:

1. You may provide default values for all data specified on this command using the Configuration. Defaults are found in the **<Data Checking>** group of the **[UAI/NASTRAN]** Section of your Configuration. You can print the contents of your Configuration with the command:

   PRINT CONFIGURATION

   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

2. The table below describes the meaning of *x* for the various **DATACHECK** options.

OPTION	NOTES	MEANING OF *x*
ASPECTRATIO	3	Specifies the aspect ratio tolerance for QUAD4, QUADR, TRIA3 and TRIAR plate elements.
BEAMOFFSETS		Specifies the maximum ratio of the offset of a one dimensional element to its length. Elements of type BAR and BEAM are checked.
CONSTRAINTS	4	
COORDERROR	5	
PLATEGEOM	6	Specifies the percentage (**0.0-100.0**) of allowable negative volume for poorly shaped QUAD4, QUADR and QUAD8 plate elements.
PLATEOFFSETS		Specifies the maximum ratio of the offset of a two-dimensional element to its thickness. Elements of type QUAD4, QUAD8, QUADR, TRIA3 and TRIAR.
QUADWARP	7	Specifies the ratio of allowable warping for QUAD4 and QUADR plate elements.
SOLIDGEOM	6	Specifies the percentage (**0.0-100.0**) of allowable negative volume for poorly shaped HEXA, PENTA and TETRA solid elements.

3. The *x* argument specifies the maximum aspect ratio that will be considered reasonable. For QUAD4 and QUADR elements, two calculations are performed. The first is the ratio of the average lengths of the opposite sides as measured in the element coordinate system. The second is the ratio of the diagonals of the element. The maximum of these ratios is taken as the error measure. For TRIA3 and TRIAR elements, the measure is defined as the ratio of the longest side to the perpendicular distance from the longest side to the opposite GRID point. You may have elements whose *x* exceeds the specified value **IGNORE**d, or you may have **WARNING** messages printed, or you may have the program terminated with the **FATAL** option. When selecting **WARNING**, you may also specify the maximum number of such messages.

4. Undefined GRID points may be easily referenced when using the **THRU** option on many Bulk Data entries. Therefore, this control, when set to **IGNORE**, allows you to make liberal use of the **THRU** option without concern for **UAI/*NASTRAN*** to terminate execution. Additionally, you may select to have warning messages relative to such non-existant GRID points printed and simultaneously limit the number of such warning messages which will be printed.

5. Previous versions of **UAI/*NASTRAN***, as do other NASTRAN versions, allow you to locate GRID points on the z-axis of cylindrical and spherical coordinate systems even when the output coordinate system was not rectangular. The documentation warns against this practice, but the program did not check for this situation. Results from this practice are not predictable. This control allows you to override the recommended default condition of a **FATAL** error for this situation.

6. Values such as 5% negative volume for the plate elements and 15% negative volume for the solid elements are recommended by UAI as reasonable tolerances to allow poorly shaped finite elements to exist in difficult modeling situations. Such default values are originally set by UAI in the **UAI/*NASTRAN*** Cnfiguration. However, UAI does not recommend that poorly shaped elements be used in areas of high or critical stress.

7. The warping ratio is value is defined as:

$$Warp = \frac{d}{\sqrt{A}}$$

where d is the distance of the GRID point from the mean plane of the element and A is its area.

Examples:

1. Request data checking of negative volume for all elements and terminate your job if these errors are detected.

```
DATACHECK SOLID=FATAL,PLATE=FATAL
```

2. Request data checking for beam and plate offset values, set the condition to warning only and print the first 25 errors encountered for each.

```
DATACHECK BEAMOFF WARN(MAXWARN=25) PLATEOFF WARN(MAXWRN=25)
```

Executive Control Command DIAGNOSTIC

Requests the PRINT of additional information or requests executive operations to be performed.

Command Syntax:

DIAG *n1* [,*n2*,...]

Option	*Meaning*	*Data Range*
ni	Specifies a diagnostic code. [1]	Integer>0

Remarks:

1. The table below summarizes the available **UAI/*NASTRAN*** diagnostic codes:

ni	RESULTING DIAGNOSTIC
5	Do not print **BEGIN** time on-line for each functional module.
6	Print **END** time on-line for each functional module
8	Print matrix trailers as they are generated. Matrix trailers provide a summary of matrix characteristics as the matrices are generated
9	Suppress echo of CHECKPOINT dictionary.
10	Use an alternate nonlinear transient loading technique: replace $\{N_{\mathbf{n+1}}\}$ by $1/3\ \{N_{\mathbf{n+1}} + N_{\mathbf{n}} + N_{\mathbf{n-1}}\}$
12	Print eigenvalue extraction diagnostics for complex inverse power methods
14	Print the Rigid Format.
16	Trace real inverse power eigenvalue extraction operations
17	Punch the DMAP sequence that is compiled.
19	Print **MPYAD**, **READ** and **SDCOMP** module messages.
21	Print the set definition table.
22	Print the degree of freedom definition table.
23	Print the DMAP **ALTER**s generated during Substructuring.
24	Punch the DMAP **ALTER**s generated during Substructuring.
28	Print a DMAP cross reference table.
29	Use the alternate **PUNCH** format for compatibility with some commercial post-processors.
30	Print I/O statistics in the LOG file.

ni	RESULTING DIAGNOSTIC
40	Punch the view factor matrices generated in Heat Transfer analysis with radiation.
41	Print a GAP element Open/Closed status table.
42	Print table data block trailers as they are generated.
43	Sets the **NOGO** flag which terminates execution after the compilation of the DMAP sequence. This is primarily used when modifying automatically generated Substructuring ALTERs.
46	Suppress checking against remaining CPU time for a job.
49	Use the alternate free field Bulk Data processing for compatibility with the MSC/NASTRAN program.
50	Print results of DMAP **EQUIV** and **PURGE** statements; similar to **DIAG 8** and **42**.
53	Allow multiple elements of different types to have the same identification number.
62	Print detail information regarding GAP element state changes during iteration.
64	Allow multiple, additive **TIC** Bulk Data entries.
70	Print Fluid-Structure interface connectivity table.

2. Unless defaults are provided in your Configuration, when you select **DIAG 17** or **DIAG 24**, you must use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH**.

3. You may provide default diagnostic values in the **<Print File Controls>** group of the **[UAI/NASTRAN]** Section of your Configuration. You may print the contents of your Configuration with the command:

   PRINT CONFIGURATION

   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

Executive Control Commands ECHOOFF and ECHOON

Disables or enables the normal echoing of Executive Control commands. When enabled, the Executive Control commands are written on the PRINT file.

Command Format:

```
ECHOOFF
ECHOON
```


Remarks:

1. The ECHO of consecutive Executive Control commands may be removed from the PRINT file by insertion of the **ECHOOFF** command into the Executive Control packet. Echo may then be reactivated by use of the **ECHOON** command.
2. As many pairs of **ECHOOFF**, **ECHOON** commands may be used as desired.
3. **ECHOON** is automatically enabled at the completion of processing the Executive Control packet.

Executive Control Command END

Defines the end of the DMAP subpacket.

Command Syntax:

```
END $
```

Remarks:

1. This command is required as the last command of a DMAP sequence (not as the last command of a DMAP ALTER, see **ENDALTER**).
2. For specific instructions related to DMAP usage, see Chapter 8.
3. The **END $** statement cannot be altered into the middle of a Rigid Format. To terminate the job, use either the

   ```
   EXIT $
   ```

   statement or the

   ```
   JUMP FINIS $
   ```

   statement.

Executive Control Command ENDALTER

Defines the end of an ALTER subpacket.

Command Syntax:

```
ENDALTER
```


Remarks:

1. This command is optional when an ALTER to a Rigid Format DMAP sequence is supplied if the ALTER packet immediately preceeds the **CEND** command.
2. This command may be used only once. When used, it terminates the ALTER subpacket.

Executive Control Command ENTITY

Allows you to define groups of ***eBase*** database entities, and to write them to, or read them from, specified ARCHIVE databases.

Command Syntax:

```
ENTITY [ GROUP  entity_name [,entity_name,...] IN group_name
       [ ASSIGN { entity_name } [,{ entity_name } ,... ] ON logical_eb_name
       [        { group_name  }   { group_name  }       ]
       [
```

Option	*Meaning*	*Data Range*
entity_name	A list one one or more *entity_name*s that appear in a DMAP Rigid Format solution sequence. [1]	Character
group_name	A name assigned to the *entity_name*s. [2,3]	Character
logical_eb_name	Logical name of a previously **ASSIGN**ed ARCHIVE database. [4]	Character

Remarks:

1. Any *entity_name*s may be listed, those not appearing in the DMAP Rigid Format sequence will be ignored.
2. Multiple commands may be used to define additional entities in the same group.
3. When you use **ENTITY ASSIGN**, then each of the specified entities is:

   WRITTEN to the ARCHIVE database **EACH** time it appears in the DMAP Rigid Format sequence if the ARCHIVE database **STATUS** is **NEW** [4,5]

   READ from the ARCHIVE database **EACH** time it appears in the DMAP Rigid Format sequence if the ARCHIVE database **STATUS** is **OLD** [4,5]

   As a result, if you use the **ENTITY ASSIGN** command in a DMAP Rigid Format sequence that performs looping, then each entity in the loop will be written to the database each time it appears. Previous versions are deleted, and, at the end of the job, only the last state of the entity is saved. This may cause a problem if you then use the database in a new job. To avoid this automatic use of the **ENTITY ASSIGN**, you may explicitly write and read entities using the DMAP modules **DBIN** and **DBOUT**. [4]

New: V20.1

4. When using an ***eBase*** ARCHIVE database, unless defaults are provided in your Configuration File, you must use the **ASSIGN** Executive Control command to define a logical file with **USE=ARCHIVE** and a **STATUS** of **NEW** when writing entities to a database, and a **STATUS** of **OLD** when reading entities from the database.

5. In order to control the specific entity that you are writing to, or reading from, an ARCHIVE database, you may use the **DBIN** and **DBOUT** DMAP modules. These are described in Chapter 8 of this manual and in Chapter 31 of the ***User's Guide***. The following example illustrates this symbolically:

   Consider the following DMAP segment (assume it is part of a DMAP Rigid Format solution sequence):

```
            ...
20          MOD1        A/B/...
21          MOD2        A/C/...
22          LABEL       TOP
23          MOD3        B,C/D/...
24          REPT        TOP,PARM $
25          MOD4        D/...
            ...
```

   If the following commands are placed in the job:

```
ASSIGN DB,NEW,USE=ARCHIVE
ENTITY GROUP B,D IN MYGROUP
ENTITY ASSIGN MYGROUP TO DB
```

   then the resulting ARCHIVE database will include two entities, **B** and **D**. The contents of **D** will reflect the data for the last iteration of the loop between the **LABEL** and **REPT** commands. So, if the loop was executed four times, then the data in **D** are only those computed in the final loop. If you want to save (and later use) entities computed **DURING** the looping, then you must use the **DBOUT** module. This must be ALTERed into the appropriate location in the DMAP Rigid Format sequence. For the above example, this would be:

```
ASSIGN DB,NEW,USE=ARCHIVE
...
ALTER 23
DBOUT    /D/*ARCHIVE*/PARM $
CEND
...
```

   This input will result in the entity **D** being saved each time through the loop. The entity name will be given a subscript based on the current value of the loop parameter **PARM**. To use a specific value of the entity for further computations, say the second set of values, you could **ASSIGN** the ARCHIVE database, remove the DMAP lines that are not needed, and restore the data from the ARCHIVE:

```
ASSIGN DB,OLD,USE=ARCHIVE
...
ALTER 1,24
DBIN     /D/*ARCHIVE*/2 $
CEND
...
```

   There is an additional module named **DBPARM** which may be used to save the value of DMAP parameters on the database. This is also described in Chapter 8 of this manual.

New: V20.1

Examples:

1. Request that every occurrence of the entities **KGG**, **MGG** and **BGG** be written to the ARCHIVE database **MYMAT**:

```
ASSIGN MYMAT,NEW,USE=ARCHIVE
ENTITY GROUP KGG,MGG,BGG TO MATRIX
ENTITY ASSIGN MATRIX ON MYMAT
```

2. Use the matrices from Example 1 in a subsequent job:

```
ASSIGN MYMAT,OLD,USE=ARCHIVE
ENTITY GROUP KGG,MGG,BGG TO MATRIX
ENTITY ASSIGN MATRIX ON MYMAT
```

3. Save the matrix **KGGX** after DMAP statement 167 in a DMAP Rigid Format solution sequence:

```
ASSIGN MYMAT,NEW,USE=ARCHIVE
ENTITY ASSIGN KGGX ON MYMAT
ALTER 167
DBOUT KGGX/'MYMAT'
CEND
```

4. Use the matrix **KGGX** saved in Example 3 after DMAP statement 167 in a DMAP Rigid Format solution sequence:

```
ASSIGN MYMAT,OLD,USE=ARCHIVE
ENTITY ASSIGN KGGX ON MYMAT
ALTER 167
DBIN KGGX/'MYMAT'
CEND
```

New: V20.1

Executive Control Command ID

Provides an identifier for your job.

Command Syntax:

```
ID  string1 , string2
```


Option	*Meaning*	*Data Range*
stringi	Any identifier of eight or fewer characters beginning with a letter.	Character

Remarks:

1. This command is optional, but if present, it must be the first Executive Control command.
2. If you do not use the **ID** command, **UAI/*NASTRAN*** will provide the default data:

   ```
   ID DEFAULT,UAINAST
   ```

3. The *stringi* data defined during a CHECKPOINT are automatically written to the CHECKPOINT file and are placed at the front of the CHECKPOINT dictionary file for security purposes. A RESTART may only be performed if these *stringi* data agree when the contents of the CHECKPOINT dictionary file and the RESTART File are compared.

Executive Control Command IMPORT

Imports external model files into **UAI/*NASTRAN***.

Command Syntax:

For UAI/*NASTRAN* ARCHIVE Databases

IMPORT ARCHIVE DATABASE FROM *logical_name*

Option	*Meaning*	*Data Range*
ARCHIVE DATABASE	The model data is obtained from a **UAI/*NASTRAN*** ARCHIVE database. [1,2]	Character
logical_name	Selects a **UAI/*NASTRAN*** logical file name. [1]	Character

Remarks:

1. When using an ***eBase*** ARCHIVE database, unless defaults are provided in your Configuration, you must use the **ASSIGN** Executive Control command to define a logical file with **USE=ARCHIVE** and a Status of **OLD**.
2. See Chapter 2 of the ***User's Guide*** for information on how the ***eBase*** ARCHIVE database is used in restarting your **UAI/*NASTRAN*** job.

Executive Control Command INCLUDE

Selects an external file which is expanded into the **UAI/*NASTRAN*** input data stream.

Command Syntax:

```
INCLUDE  { ['] phys_name ['][,params] }
         {   ALTERLIB ( alternamе )   }
```

Option	*Meaning*	*Data Range*
phys_name	Specifies the host computer dependent name of the physical file containing data that you wish to have included in the input data stream. [1,2]	Character
alternamе	Is the name of a DMAP ALTER packet which is stored in the **UAI/*NASTRAN*** ALTER Library. [3]	Character
params	Selects host computer dependent parameters.	[4]

Remarks:

1. Physical file names vary depending on your host computer. See Chapter 1 for details
2. The character string that you specify for *phys_name* is always converted to upper case characters by **UAI/*NASTRAN***. If your host computer has case-sensitive file names, and if you wish to use lower case characters, then you must enclose *phys_name* in single quotation marks, sometimes called tics.
3. The contents of the **UAI/*NASTRAN*** ALTER Library are documented in Chapter 9. **UAI/*NASTRAN*** ALTER Library packets do not contain **ENDALTER** commands. Therefore, this form of the **INCLUDE** command must always be placed at the end of the Executive Control packet, and may only be followed by additional ALTER statements, **ENDALTER** or **CEND**.
4. Other host computer dependent parameters may be available. See Chapter 1 for details.
5. Any number of **INCLUDE** commands may appear in Executive Control command packet.
6. An **INCLUDE** file may not contain any other **INCLUDE** commands.

Executive Control Command LINES

Defines the number of lines of output per printed page.

Command Syntax:

```
LINES = nline
```

Option	*Meaning*	*Data Range*
nline	Specifies the number of printed lines per page. [1,2]	Integer>0

Remarks:

1. You may provide default value for this command in the **<Print File Controls>** group of the **[UAI/NASTRAN]** Section of your Configuration. You may print the contents of your Configuration with the command:

   PRINT CONFIGURATION

   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

2. For 11 inch paper, 50 lines per page is recommended; for 8½ inch paper, 35 is recommended.

Executive Control Command MAXLINES

Specifies the maximum number of lines of printed output. When this limit is exceeded, the **UAI/*NASTRAN*** execution is terminated.

Command Syntax:

```
MAXLINES = maxline
```

Option	*Meaning*	*Data Range*
maxline	Specifies the maximum number of output lines that you wish to allow. [1,2]	Integer>0

Remarks:

1. You may provide default value for this command in the **<Print File Controls>** group of the **[UAI/NASTRAN]** Section of your Configuration. You may print the contents of your Configuration with the command:

   PRINT CONFIGURATION

   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

2. **UAI/*NASTRAN*** terminates execution after *maxline* lines have been printed.

Executive Control Command MEMORY

Specifies the amount of memory **UAI/*NASTRAN*** will use for the internal storage of data.

Command Syntax:

2 EXEC

```
MEMORY [ { work_mem          } ]
       [ { WORKING = work_mem } ]

                    [ EBASE = eb_mem ] [ PHYSICAL = phys_mem ]
```

Option	*Meaning*	*Data Range*
work_mem	Specifies the working memory size. [1,2]	Integer>0
eb_mem	Specifies the ***eBase*** database memory size. [1,3]	Integer>0
phys_mem	Specifies the real physical memory memory size. [1,4]	Integer>0

Remarks:

1. The units of the working memory size are determined by the two optional command arguments.

   ```
   [ { M } ]  [ { W } ]
   [ { K } ]  [ { B } ]
              [ { P } ]
   ```

   The first argument indicates an order of magnitude for *memory_space*, **M** for millions, **K** for thousands. The second argument indicates the unit specifier as single precision words (**W**), bytes (**B**), or computer precision words (**P**). If neither is present, then *memory_space* is taken to be ***single precision computer words***.

2. The working memory may be increased for large problems to reduce the amount of physical I/O. Note, however, that this may cause increased paging on host computers with virtual memory operating systems. Contact your **UAI/*NASTRAN*** Support Specialist for additional details.

3. The ***eBase*** memory is a separate memory pool used by the database during execution. Normally the default value in the delivered Configuration is sufficient, but if you use block sizes larger than the default for any database, this value may need to be increased.

4. The physical memory is used to control certain advanced algorithms. Contact your **UAI/*NASTRAN*** Support Specialist for additional details.

5. For some host computers, there may be a relationship between memory requested by your Job Control Language (JCL) and this command. See Chapter 1 for details.

6. You may provide default value for this command in the **<Computing Resources>** group of the **[UAI/NASTRAN]** Section of your Configuration. You may print the contents of your Configuration with the command:

   PRINT CONFIGURATION

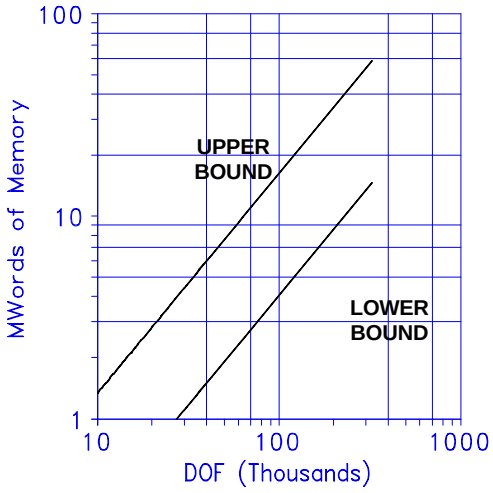
   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

7. **UAI/*NASTRAN*** has two high-performance solvers which take advantage of the latest developments in sparse matrix algorithm technology. The first of these is the symmetric matrix decomposition used in static analyses, and the second is the Lanczos eigenextraction method. This latter method is used for extracting a modest number of eigenvalues from very large systems. When these solvers are used, memory requirements may become significant. The figures below give upper and lower bound estimates for the amount of memory that you should specify on your **MEMORY** Executive Control Command. Although the eigensolver takes slightly more memory, about 20%, the same figures may be used to approximate the requirements for either solver. Note, that in the case of the Linear Solver, if you do not specify enough memory for the new algorithm, the program will revert to the old solution algorithm. This is not the case for Lanczos — the job will terminate. These curves have been created using a representative sample of real analysis jobs. They are intended only to be used as guidelines — a specific job may take significantly more or less memory than indicated.

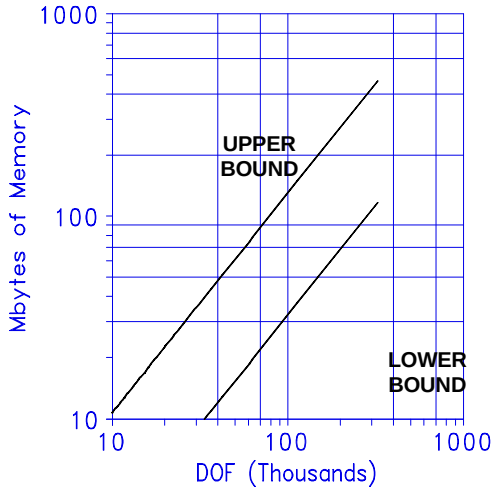
Examples:

1. Execute **UAI/*NASTRAN*** using 12 million words of working memory:

   MEMORY = 12000000; or **MEMORY** = 12000KW; or **MEMORY** = 12MW



SOLVER MEMORY — Cray



SOLVER MEMORY — Others

Executive Control Command PRINT CONFIGURATION

Request printing of your current **UAI/*NASTRAN*** Configuration.

Command Syntax:

```
PRINT CONFIGURATION
```

Remarks:

1. This command is provided to allow you to determine the default values of numerous parameters which you may wish to modify for your specific **UAI/*NASTRAN*** execution.

Executive Control Command SECONVERT

Selects the Superelement-to-Substructuring convertor which creates UAI/*NASTRAN* substructuring input files from an MSC/NASTRAN Superelement input file.

Command Syntax:

```
SECONVERT [ = [ { YES } ] [ , OUTPUT = { BULK  } ] ]
          [   { NO  } ]   [            { SPLIT } ]
```

Option	*Meaning*	*Data Range*
{ **YES** } { NO }	Selects or deselects the execution of the convertor.	Keyword
{ **BULK** } { SPLIT }	Selects the manner in which the output substructuring files are processed. Either all substructures are placed on a single **BULK** file, or the data packets are **SPLIT** into separate files.	Keyword

Remarks:

1. The convertor processes the MSC/NASTRAN Superelement Case Control command **SUPER**. The options **ALL**, a Superelement id and a Superelement set id are all processed. Any load sequence number present is ignored. Superelement commands **LOADSET**, **SEALL**, **SEDR**, **SEDV**, **SEEXCLUDE**, **SEKREDUCE**, **SELGENERATE**, **SELREDUCE**, **SEMGENERATE**, **SEMREDUCE**, **SERESP**.
2. The following table summarizes the MSC/NASTRAN Superelement Bulk data entries that are processed by the convertor:

BULK DATA	DESCRIPTION
DTI, SETREE	Defines the Superelement tree connectivity. Required for multilevel analyses.
SESET, GRID	Defines the interior points of a Superelement
SEBSET,1 SECSET,1	Moves boundary degrees of freedom from fixed to free
SEQSER1	Defines modal degrees of freedom for component mode synthesis. Although GRID points may be used, for clarity it is advised that scalar points or GRID points with a single degree of freedom be used as modal degrees of freedom.

The entries **LSEQ**, **SEQSET**, **CSUPER**, **CSUPEXT**, **SESUP**, **SEELT**, **SEQSEP**, **SEUSET**, **SEUSET1** are not processed.

Example:

1. Request that the convertor run and produce separate files for each Substructure.

```
SECONVERT = YES , OUTPUT=SPLIT
```

Rev: V20.1

Executive Control Command SEQUENCE

Selects GRID point resequencing to minimize CPU time using techniques to minimize either the stiffness matrix wavefront or bandwidth.

2 EXEC

Command Syntax:

```
SEQUENCE [ = [ { YES } ] [ , CRITERIA = { RMSWAVE   } ] [ , { MPC   } ] ]
         [     { NO  } ]   { BANDWIDTH }                   { NOMPC }

         [ , PRINT = { SUMMARY } ] [ , PUNCH ] ]
                     { DETAIL  }
                     { NONE    }
```

Option	*Meaning*	*Data Range*
{ **YES** } { NO }	Selects or deselects the sequencing of GRID points. [1,2]	Keyword
{ **RMSWAVE** } { BANDWIDTH }	Selects the criterion for determining the internal sequence. [3]	Keyword
{ MPC } { **NOMPC** }	Selects or deselects the inclusion of MPC and Rigid Element data in the resequencing.	Keyword
{ **SUMMARY** } { DETAIL } { NONE }	Selects the sequencer print option. you may choose **SUMMARY** tables or a more **DETAIL**ed report of sequencing statistics report, which is not recommended. **NONE** disables the request.	Keyword
PUNCH	Requests that sequence Bulk Data, **SEQGP**, be written to the BULK file. [4]	Keyword

Remarks:

1. **UAI/*NASTRAN*** automatically resequences the input GRID point data to maximize the efficiency of matrix operations. To deselect this option, you must either enter a **SEQUENCE = NO** command in the Executive Control packet or have Bulk Data entries **SEQGP** present in your input file.
2. This option is automatically deselected if any **SEQGP** Bulk Data entries are present in the input file.

REV: V20.1

3. The default sequencing criterion, **RMSWAVE** (minimize root-mean-square matrix wavefront), is recommended for all analyses on computers with scalar CPU processors. The alternate method, **BANDWIDTH** (minimize matrix bandwidth), is recommended for use on vector processor computers under certain modeling conditions. These conditions are those modeling situations where a significant static condensation operation or dynamic reduction operation will occur. Typical examples for this case include the user specification of an a-set prior to a Givens or modified Givens eigenvalue analysis, the use of an a-set in a statics PHASE 1 Substructuring Analysis, or use of Dynamic Reduction in either Normal Modes or Dynamic Response analysis. The following table summarizes these conditions.

Condition	Scalar CPU	Vector CPU
Default Value	RMSWAVE	RMSWAVE
Static Analysis	RMSWAVE	RMSWAVE
Static or Dynamic Reduction	RMSWAVE	BANDWIDTH

4. Unless defaults are provided in your Configuration, you must use the **ASSIGN** Executive Control command to define a logical file with **USE=BULK** if you use the **PUNCH** option.

5. You may provide default value for this command in the **<Matrix Conditioning>** group of the **[UAI/NASTRAN]** Section of your Configuration. You may print the contents of your Configuration with the command:

   ```
   PRINT CONFIGURATION
   ```

   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

Examples:

1. Execute **UAI/*NASTRAN*** without using the GRID point sequencer:

   ```
   SEQUENCE = NO
   ```

2. Run the GRID point sequencer and obtain a detailed report of its activity. and additionally request that the **SEQGP** Bulk Data entries be written to the BULK file:

   ```
   SEQUENCE = YES,PRINT = DETAIL,PUNCH
   ```

3. Perform normal modes analysis with dynamic reduction and an a-set on a vector processor computer:

   ```
   SEQUENCE CRITERIA=BANDWIDTH
   ```

Executive Control Commands SKIPON and SKIPOFF

Disables and enables the processing of Executive Control commands.

Command Syntax:


```
SKIPON
SKIPOFF
```

Remarks:

1. The processing of Executive Control commands is suspended when a **SKIPON** command is encountered in the Executive Control packet. All following commands are echoed but not processed until a **SKIPOFF** command is encountered. ECHOing of skipped commands may be controlled by the **ECHOOFF** and **ECHOON** commands.
2. As many pairs of **SKIPON**, **SKIPOFF** commands may be used as desired.
3. **SKIPOFF** is automatically enabled at the completion of processing the Executive Control packet.

Executive Control Command SOLUTION

Selects either a DMAP Rigid Format Solution by its solution identification number or solution name, or the Multidisciplinary Solution Sequence.

Command Syntax:

```
SOL { id      }
    { rfname  }
    { CHECKOUT }
```

Option	*Meaning*	*Data Range*
id	Identification number of a Rigid Format. [1]	Integer>0
rfname	Name of Rigid Format. [1]	Character
CHECKOUT	Requests that the preface and data checking modules be executed, and then the job terminated. [2]	Keyword

Remarks:

1. The *rfname* that is used depends upon your solution **APP**roach. The values of *id* for the Displacement, Heat and Nonlinear approaches are shown in the following table:

id	*rfname*		
	APP DISPLACEMENT	**APP HEAT**	**APP NONLINEAR**
–	MULTI [3]		
1	STATICS	STATICS	STATICS
2	INERTIA RELIEF		
3	MODES NORMAL MODES REAL EIGENVALUES	STEADY STATE	
4	DIFFERENTIAL STIFFNESS		
5	BUCKLING		
7	DIRECT COMPLEX EIGENVALUES		
8	DIRECT FREQUENCY RESPONSE		
9	DIRECT TRANSIENT RESPONSE	TRANSIENT	
10	MODAL COMPLEX EIGENVALUES		
11	MODAL FREQUENCY RESPONSE		

Rev: V20.1

id	*rfname*		
	APP DISPLACEMENT	**APP HEAT**	**APP NONLINEAR**
12	MODAL TRANSIENT RESPONSE		
13	DIFFERENTIAL STIFFNESS MODES		
14	CYCLIC STATICS		
15	CYCLIC MODES		
16	CYCLIC BUCKLING		
17	SHOCK		
51	SENSITIVITY STATICS		
52	SENSITIVITY MODES		

2. The **CHECKOUT** option is useful in the early stages of model development. It allows you to simply check the consistency of your input data and then terminate the job. It avoids the need to use a DMAP ALTER to accomplish the same result.

   If you have an **ARCHIVE** command in your Case Control packet, then the model geometry will be written to the ARCHIVE database. This database may then be used, assuming all data are correct, to perform a restart.

3. When you select **SOL MULTI**, then you may perform analysis and design optimization for the **STATICS**, **INERTIA RELIEF**, **NORMAL MODES**, **DIRECT FREQUENCY RESPONSE** and **MODAL FREQUENCY RESPONSE** disciplines. In addition, you may perform sensitivity analyses. All disciplines are controlled by using the Case Control command **CASE**.

4. The **SOL** command is mandatory unless you are using **APP DMAP**.

Executive Control Command SOLVER

Selects the desired solver algorithm for static analysis.

Command Syntax:

```
SOLVER { BEST     }
       { SPARSE   }
       { STANDARD }
```

Option	*Meaning*	*Data Range*
BEST	Allows **UAI/*NASTRAN*** to select the optimum static analysis solution algorithm. [1]	Keyword
SPARSE	Requests use of the Sparse Matrix solution algorithm. [1]	Keyword
STANDARD	Requests use of the Standard Matrix solution algorithm. [1]	Keyword

Remarks:

1. This command allows you to select a particular static analysis solution algorithm. This might be necessary for special situations involving limited memory conditions or benchmark performance studies.
2. You may provide default value for this command in the **[eBase:matlib]** Section of your Configuration. You may print the contents of your Configuration with the command:

   ```
   PRINT CONFIGURATION
   ```

   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

Executive Control Command TIME

Specifies the maximum time, in minutes, that will be allotted to the execution of the **UAI/*NASTRAN*** program.

Command Syntax:

```
TIME  cpu_max
```

Option	*Meaning*	*Data Range*
cpu_max	Specifies the maximum number of CPU minutes for execution.	Integer>0

Remarks:

1. You may provide default value for this command in the **<Computing Resources>** group of the **[UAI/NASTRAN]** Section of your Configuration. You may print the contents of your Configuration with the command:

   ```
   PRINT CONFIGURATION
   ```

   Your System Support Specialist can provide you with more information about this advanced **UAI/*NASTRAN*** feature.

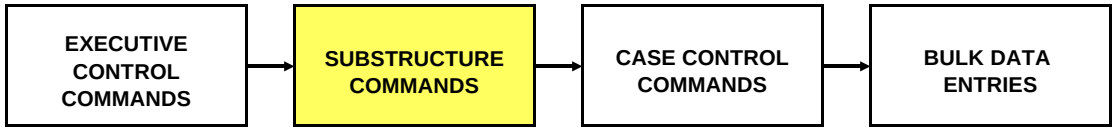
2. The *cpu_max* that you specify should be less than the time allotted the entire execution which is often specified through an operating system Job Control Language statement.

Chapter 3

SUBSTRUCTURE COMMANDS

Substructure Control commands are used to control the execution of the **UAI/*NASTRAN*** substructuring and modal synthesis capability. If this optional capability is used, the Substructure Control data packet is the second in the input data stream as shown in Figure 3-1.

Figure 3-1. SUBSTRUCTURE CONTROL PACKET LOCATION



The Substructure Control commands provide you with a powerful and flexible tool for performing complex substructure analyses. The commands direct the execution of **UAI/*NASTRAN*** through automatically generated DMAP ALTERs to the Rigid Format that you use.

3.1 THE SUBSTRUCTURE COMMANDS

The general form of the Substructure Control Command packet is:

```
                 { PHASE1 }
SUBSTRUCTURE     { PHASE2 }
                 { PHASE3 }

   COMMAND option_list
      SUBCOMMAND option_list
      SUBCOMMAND option_list
      ...
   COMMAND option_list
   ...
ENDSUBS
```


The packet must begin with the **SUBSTRUCTURE PHASEi** selection command and it must end with the **ENDSUBS** command. The packet itself is then defined by the substructuring operations that you wish to perform during the current **UAI/*NASTRAN*** exection. Within the packet are a number of *Substructure Control commands*. Each command may include an *option_list*, which is a list of optional operands to the command. Additionally, some of the commands have one or more *subcommands* associated with them. As before, these subcommands may also have their own *option_list*. The available Substructure Control commands are summarized in Table 3-1, where they are grouped in three functional categories. The use of these commands is discussed in detail in the ***User's Guide***.

Table 3-1. SUMMARY OF SUBSTRUCTURE CONTROL COMMANDS

COMMAND TYPE	COMMAND NAME	COMMAND DESCRIPTION
CONTROL COMMANDS	ENDSUBS	Ends substructure commands.
	OPTIONS	Defines matrix options.
	SUBSTRUCTURE	Selects the substructure processing phase.
SUBSTRUCTURE OPERATIONS	BASIC	Creates a Basic Substructure.
	COMBINE	Combines two or more substructures.
	CREDUCE	Performs a complex modal reduction.
	EQUIV	Creates a secondary (and image) substructure(s).
	MASSBAL	Prints substructure weight and balance data.
	MRECOVER	Recovers modal solution results.
	MREDUCE	Performs a real modal reduction.
	PLOT	Plots an undeformed substructure.
	RECOVER	Recovers PHASE 2 or PHASE 3 solution results.
	REDUCE	Performs a static reduction on a substructure.
	SOLVE	Performs the solution of a substructure.
SOF OPERATIONS	CHECK	Verifies the contents of an SOF backup.
	DELETE	Removes selected data from the SOF.
	DESTROY	Removes complete substructures from the SOF.
	EDIT	Removes related data from the SOF.
	SOFIN	Imports substructure data.
	SOFOUT	Exports substructure data.
	SOFPRINT	Prints an SOF Table of Contents and other data.
MISCELANEOUS OPERATIONS	ECHOON ECHOOFF	Enables and disables the echoing of the Substruture Command packet.
	SKIPON SKIPOFF	Enables and disables the processing of the Substruture Command packet.

3.2 AUTOMATICALLY GENERATED DMAP ALTERS

Each of the substructuring commands produce a set of DMAP ALTERs which are automatically inserted into the Rigid Format previously selected by the Executive Control command **SOL**.

It is possible to modify these automatically generated ALTERs, or to add your own additional ALTER statements. However, any additional ALTERs may not overlap those that are automatically generated. The DMAP statement numbers that are modified by substructuring, and therefore cannot be included directly or indirectly in your ALTER packet, are presented for each Rigid Format in Chapter 9. You may Print and Punch the automatically generated ALTER packets by using the Executive Control command:


```
DIAG 23,24
```

The ***User's Guide*** presents details for modifying the automatically generated ALTERs and merging your ALTERs with them for any special requirements that you may have.

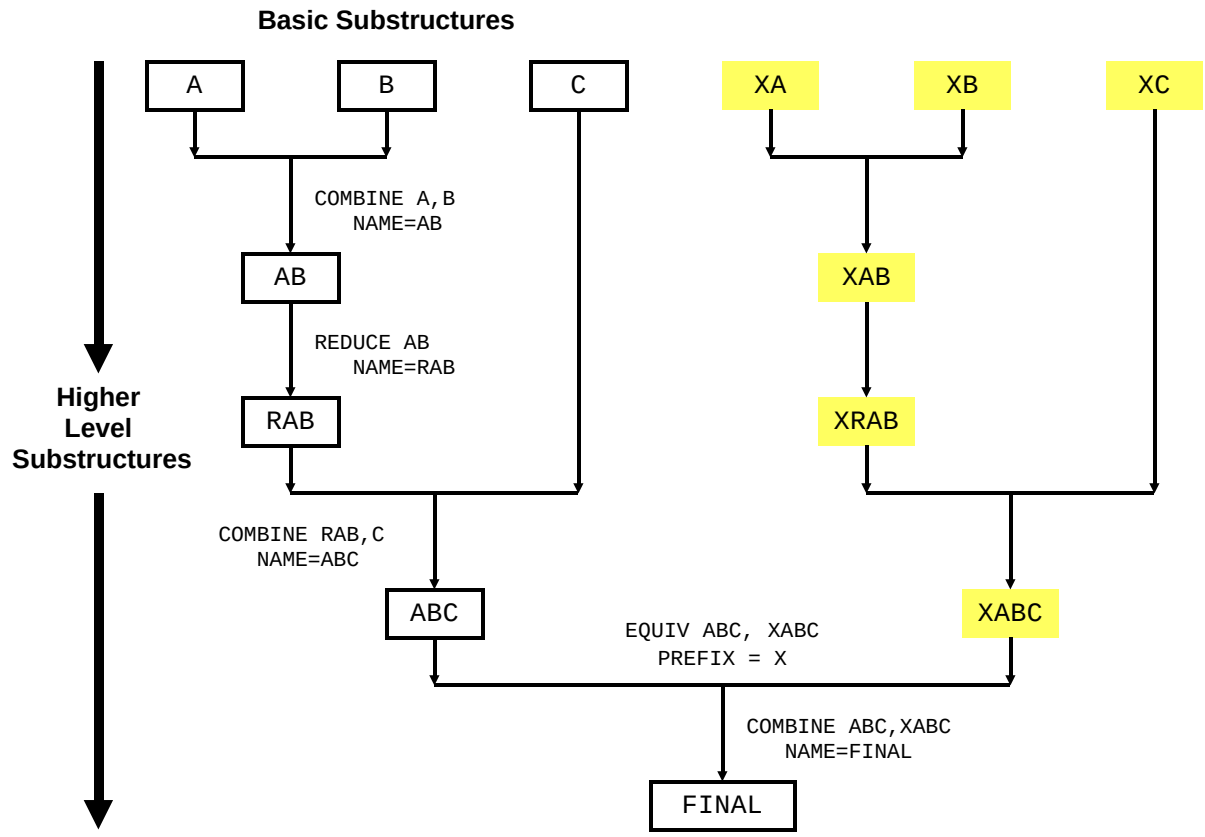
3.3 SUBSTRUCTURE TERMINOLOGY REVIEW

You will find an extensive description of the substructuring and modal synthesis capabilities in the **UAI/*NASTRAN User's Guide***. This section reviews some important terminology which is encountered in the command descriptions found at the end of this chapter.

The SOF Database. Is a database which contains all of the information about your substructuring analysis. Each substructure is defined by specific database entities which are called ***Items***.

Substructuring Tree. The substructuring operations that you perform create a logical ***Analysis Tree*** which defines the relationships between the different substructures. Most often, your analysis planning should include the definition of the tree prior to any executions. An example tree, and the Substructure Control commands used to generate it, are shown in Figure 3-2.

Figure 3-2. SUBSTRUCTURING ANALYSIS TREE



Substructure. Every node in your substructuring tree is called a ***Substructure***. There are, however, special terms used to further describe substructures. These terms generally relate to the manner in which the substructure was created.

Basic Substructure. When you execute PHASE 1 of substructuring, you create ***Basic Substructures***. These form the basic building blocks of the substructuring process. Substructures **A**, **B**, and **C** in the tree of Figure 3.2 represent Basic Substructures. The remaining types of substructures are then created by performing operations on other existing substructures.

Higher Level and Lower Level Substructures. When you **COMBINE** two or more substructures or perform one of the **REDUCE** operations on a substructure, your new substructure is called a ***Higher Level Substructure***. This indicates that it has been synthesized from other simpler ***Lower Level Substructures***. In the example tree, **AB** has been created by combining the Lower Level Substructures **A** and **B**. Similarly, substructure **RAB** represents a reduction of **AB**, **RAB** thus becomes a Higher Level Substructure of which **AB** is a Lower Level Substructure.

Component Substructures. The Lower Level Substructures are also called ***Component Substructures***. If they also happen to be Basic Substructures, then they are called ***Basic Components***. This is important because, as you will see, nearly all substructuring Bulk Data entries reference data which may only be contained in Basic Components.

Primary, Secondary and Image Substructures. The **EQUIV** operation makes a copy of a substructure. This new copy is called a ***Secondary Substructure***, whereas the original one is called the ***Primary Substructure***. In the example, **XABC**, the lightly shaded box, is a Secondary Substructure of **ABC**, its Primary Substructure. When you perform the **EQUIV** on a Higher Level Substructure, then the entire subtree is copied. In the example, the substructures with dark shading, **XA**, **XB**, **XC**, **XAB**, and **XRAB**, define this subtree. These substructures are called ***Image Substructures***. Note that the names of all of the image substructures were automatically generated by concatenating the **PREFIX** specified in the **EQUIV** command with the names of the components of the primary substructure.

3.4 SUBSTRUCTURE CONTROL COMMAND DESCRIPTIONS

The remainder of this chapter provides you with a description of the Substructure Control commands including their syntax and examples of their use. All Substructure Control commands are entered in free-field format. In presenting the general formats for each command and its options, the following conventions are used:

- Many Substructure Control commands contain ***keywords***. All keywords are shown in capitalized computer type such as: ALTER. All such keywords must be entered exactly as they are specified, subject only to the exception that they may be abbreviated by their first four characters.

- Some commands contain ***parentheses***. These must be entered if an option requiring them is selected.

- Lower case italicized computer type, such as *comp_name*, indicates that you must provide a specific data value.

- Braces { } enclose a list of two or more options from which you may select one.

- Brackets [ ] indicate that the enclosed keywords and parameters may be omitted when you use the command.

- When a choice is to be made from a list, the default choice is presented in boldface type as in the following: { CHOICE1 / CHOICE2 / **DEFAULT** }

- The special symbol ♦ indicates a ***required*** subcommand.

- A single command line may not exceed 72 characters. However, you may continue a command by ending the current record with a comma and continuing to the next record.

Additionally, command options have a valid data range and, in some cases, a default value. The following table defines the data range specifications found in this chapter.

Data Range Specifier	Meaning
Integer	The data must be an integer number in the range of indicated values.
Real	The data must be a real number in the range of indicated values.
Character	The data must be a string of characters beginning with a letter.
Keyword	The data must be a character string matching the first four or more unique characters of a keyword option.
Name	The data must be a character string providing a substructure name.

This page is intentionally blank.

Substructure Command BASIC

Assigns a name to a Basic Substructure during a PHASE 1 execution.

Command Syntax:

```
BASIC = name
```

Subcommands:

```
SAVEPLOT   = plotsid
GRIDOFFSET = offset
OVERWRITE
```

Option	*Meaning*	*Data Range*
name	Specifies the name that you wish assigned to the Basic Substructure which is being created in PHASE 1. [1]	Name
plotsid	Specifies the plot set identification number used to define the set of elements of Basic Substructure *name* to be saved in PHASE 1 for subsequent plotting in PHASE 2. [2]	Integer>0
offset	Specifies an offset value to be added to all GRID point identification numbers for the substructure model. [3]	Integer>0
OVERWRITE	Specifies that previously existing model data will be deleted before the Basic Substructure is created. [4]	Keyword

Remarks:

1. A separate PHASE 1 execution is required for each Basic Substructure.
2. Only one **SAVEPLOT** subcommand may be defined for a Basic Substructure. If you use the **SAVEPLOT** subcommand, the elements selected in the plot set will be used when plotting substructure *name* or any other substructure of which *name* is a component.
3. The **GRIDOFFSET** subcommand allows you to modify the external GRID point identification numbers for the substructure. This is used to insure unique identification numbers during a PHASE 2 **SOLVE** operation when new model or loads data are being added to the model.
4. Prior to creating the Basic Substructure, the entire substructure tree is analyzed to make certain that the model data does not exist. If you wish to re-create your model, then you can either manually **DESTROY** it, or you can accomplish the same result by simply using the **OVERWRITE** subcommand. If you do this, then all substructures of which *name* is a component will also be destroyed.

Substructure Command CHECK

Lists all substructure items contained on an import file previously exported using the **SOFOUT** command with **INTERNAL** format.

Command Syntax:

CHECK *logical_name*

Subcommands: None

Option	*Meaning*	*Data Range*
logical_name	Logical name of the import file. [1]	Character

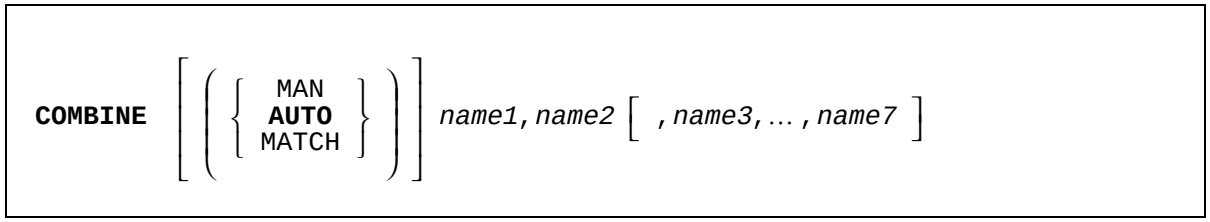
Remarks:

1. You must use the **ASSIGN** Executive Control command to define the *logical_name* with **USE=SOFIN** or **USE=SOFOUT**.
2. When you perform a **CHECK**, you obtain a listing of the data contained on the specified *logical_name*. These data include the substructure name, item name, and the date and time the item was written using the **SOFOUT** command.

Substructure Command COMBINE

Creates a new substructure by combining two to seven existing substructures.

Command Syntax:



Subcommands:

- **NAME** = *newname*
- **TOLERANCE** = *tol*
  OUTPUT = *opt1* [,*opt2*,...]
  CONNECT = *n*
  COORD = *choice*
  COMPONENT = *cname*
  TRANSFORM = *tid*
  SYMTRANSFORM = [X][Y][Z]
  LOCAL = *action*

Option	*Meaning*	*Data Range*
{ MAN / **AUTO** / MATCH }	Selects the combination mode as **AUTO**matic, **MAN**ual, or **MATCH**ing. [1]	Keyword
namei	Specifies the names of from two to seven substructures which will be combined.	Name
newname	Specifies the name of the resulting combined substructure.	Name
tol	Defines the limit of distance between points which will be automatically connected.	Real > 0.0
opti	Requests selected output. [2]	Integer>0
n	Provides the set identification number of manual GRID point connection and release information specified using Bulk Data entries **CONCT**, **CONCT1**, **RELES** and **RELES1**. [1]	Integer>0
choice	Specifies a coordinate system for automatically generating **GTRAN** data when connection points have conflicting local coordinate systems. [3]	[3]
cname	Specifies a component substructure, one of the *namei*, to which transformation commands which follow will be applied.	Name

REV: V20.1

tid	Set identification number of **COORD2R** and **GTRAN** Bulk Data entries. [4]	Integer>0
[X][Y][Z]	Defines an axis, or set of axes, normal to the plane(s) of symmetry in the new Basic coordinate system. [5]	Keyword
action	Specifies a local coordinate system processing option. [6]	Keyword

Remarks:

1. There is a hierarchy of processing associated with GRID point connection processing. This hierarchy is described as follows:

   - Automatic connections are made first based on geometry, only if the **MAN** option is not selected with the **COMBINE** command.
   - **CONCT** Bulk Data define the initial connections if the **MAN** option is selected.
   - **RELES** and **RELES1** Bulk Data override any connections generated by either the automatic technique or defined with **CONCT** Bulk Data.
   - **CONCT1** Bulk Data override any effects of **RELES** Bulk Data.
   - The **MATCH**ing method is equivalent to the **MAN**ual option. **CONCT** Bulk Data entries are automatically generated to connect matching GRID point identification numbers in the substructures. This option should only be used when the GRID point numbering is unique throughout the substructures being combined. This is the common occurance in Superelement models.

2. The following output requests are available for the COMBINE operation. Shaded options are recommended for general use.

CODE	OUTPUT ITEM
2	SOF Database Table of Contents
3	**CONCT1** Bulk Data Summary
4	**CONCT** Bulk Data Summary
6	**GTRAN** Bulk Data Summary
9	**RELES** Bulk Data Summary
10	Connectivity map of the final combined substructure which includes only connected points. (See 12 below)
11	Summary of automatically generated connections in terms of internal point numbers.
12	Complete connectivity map of the final combined substructure. This map defines each internal point in terms of the GRID point ID and component substructure it represents.
13	The EQSS Item for the resulting substructure.
14	The BGSS Item for the resulting substructure.
15	The CSTM Item for the resulting substructure.
16	The PLTS Item for the resulting substructure.
17	The LODS Item for the resulting substructure.

3. The **COORD** command selects a coordinate system for resolving connections that have conflicting local coordinate systems. If the command is not used, such mismatches result in a fatal error.

   If there is ***any*** conflict between the coordinate systems of the points being combined, then all points are transformed to the same coordinate system. If you select the command **COORD = BASIC**, then they are all transformed to the Basic system, and if you select an explict coordinate system, *cid*, then they are all tranformed to that system.

4. The **COORD2R** Bulk Data define the orientation of the original Basic coordinate system of a component substructure in terms of the new Basic coordinate system. The output of all GRID points in the original Basic coordinate system will be transformed to the new Basic coordinate system. *Note that the more restrictive **TRANS** Bulk Data entry still functions correctly to insure compatibility of earlier substructuring jobs.*

   Points with output defined in local coordinate systems will not be transformed unless otherwise specified using the **LOCAL** subcommand or **GTRAN** Bulk Data entries. This means that such local coordinate systems will rotate in space during the **COMBINE** operation retaining their same relative motion to the body of the substructure.

5. The **SYMTRANFORM** request is primarily used to produce symmetric reflections of a structure. This is usually preceded by an **EQUIV** command to produce a new, unique substructure name for the second, symmetric substructure. Note that the displacements and nodal coordinates in these directions are reversed in sign and that the results for the reflected substructure will be expressed in a left-handed coordinate system whenever either one or three axes of symmetry exists.

6. The **LOCAL** command controls overall handling of any local coordinate systems in the component substructure. The default condition is equivalent to the command **LOCAL=NOTRANS**, which causes all local coordinate system motions to remain fixed in their original relative directions to the structure. The command **LOCAL=BASIC** causes all local coordinate system motions to be transformed to the Basic coordinate system of the new substructure. This procedure will always assure that GRID points which should be combined, and which have their motions defined in incompatible local coordinate systems, will be correctly joined with compatible motions in the new Basic coordinate system. **GTRAN** data may be used for individual GRID points to override an overall **LOCAL** command.

Examples:

1. Perform an automatic **COMBINE** of substructures **PANEL** and **SPAR** resulting in a new substructure called **SECTION**. Use a connection **TOLE**rance of **0.0001**.

```
COMBINE PANEL SPAR
   TOLE = .0001
   NAME = SECTION
```

2. Perform an automatic combine of the three substructures **TANK1**, **TANK2** and **BULKHD**. Specify a connection tolerance of **0.01** and a transformation for substructure **TANK1** which is defined by a **TRANS** Bulk Data entry whose identification number is 4:

```
COMBINE TANK1, TANK2, BULKHD
   NAME = TANKS
   TOLE = .01
   COMPONENT TANK1
      TRAN = 4
```

Rev: V20.1

3. Combine two substructures, name **LWING** and **RWING**, and name the resulting substructure **WING**. Perform a symmetric reflection of **LWING** about the Y-axis during the operation.

```
COMBINE LWING, RWING
   NAME = WING
   COMPONENT LWING
      SYMT = Y
```

Substructure Command CREDUCE

Performs a complex modal synthesis reduction on a specified component substructure.

Command Syntax:

```
CREDUCE  name
```

Subcommands:

- **NAME** = *newname*
- **BOUNDARY** = *bsid*
  FIXED = *fsid*
- **METHOD** = *methid*
  RANGE = *f1,f2*
  NMAX = *nmodes*
  GRIDOFFSET = *offset*
  GPARAM = *g*
  RSAVE
  OUTPUT = *opt1 [,opt2,...]*

Option	*Meaning*	*Data Range*
name	Specifies the name of the substructure to be reduced.	Name
newname	Assigns a name to the resulting substructure. [1]	Name
bsid	Selects the identification number of **BDYC** Bulk Data entries which define the boundary degrees of freedom. [1]	Integer>0
fsid	Selects the identification number of **BDYC** Bulk Data entries which define degrees of freedom temporarily fixed during mode extraction.	Integer>0
method	Selects an **EIGC** Bulk Data entry which controls the eigenvalue extraction. [2]	Integer>0
f1,f2	Specifies a natural frequency range, in Hz, for selecting modal coordinates. [2]	Real≥0.0
nmodes	Specifies the number of modal coordinates to be selected. [2]	Integer>0
offset	Specifies an offset to be added to all GRID point identification numbers that are used as modal degrees of freedom. [3]	Integer>0
g	Selects a structural damping parameter.	Real>0.0
RSAVE	Requests that the LMTX item be saved on the SOF Database. [4]	Keyword
opti	Requests selected output. [5]	Integer>0

Remarks:

1. The new substructure, *newname*, will contain only those physical degrees of freedom which are included in the boundary set, *bsid*. Because this substructure is be defined by complex matrices, you may only use it in subsequent **COMBINE** and **SOLVE** operations. Furthermore, the **SOLVE** operation must be performed with Rigid Format 8, Direct Frequency Response Analysis.

2. When you specify a natural frequency **RANGE**, then all of the eigenvectors of modes within the range will be used as generalized coordinates for the reduction. There must be at least one frequency in the range. Similarly, if you specify **NMAX**, then the first *nmodes* flexible eigenvectors will be used as generalized coordinates unless there are actually fewer extracted by the selected eigenvalue procedure. In this case, the number extracted will be used. If you do not use either of these subcommands, then all of the modes extracted, based on your **EIGC** Bulk Data entry, will be used. You may not use both of these subcommands.

3. The **GRIDOFFSET** subcommand allows you to specify the new external GRID point identification numbers for the substructure. This is used to insure unique identification numbers during a PHASE 2 **SOLVE** operation when new model or loads data are being added to the model.

4. An important result of the **CREDUCE** operation is the lower triangular factor, LMTX, of the decomposed matrix representing the degrees-of-freedom which have omitted during the reduction process. This matrix is both costly to compute and its storage often represents more than half of the space on your SOF Database. If your host computer has a large amount of available disk space, then you may use the **RSAVE** subcommand to save the LMTX on the Database. On the other hand, if disk space is at a premium, you do not use the **RSAVE** command. Later, if you are performing data recovery for the omitted points, the matrix decomposition will be redone as necessary.

5. The following output requests are available for the **CREDUCE** operation. Shaded options are recommended for general use.

CODE	OUTPUT ITEM
1	Current problem summary
2	Boundary set summary
3	Summary of GRID point identification numbers in the boundary set
4	The EQSS item for the substructure being reduced
5	The EQSS item for the new substructure
6	The BGSS item for the new substructure
7	The CSTM item for the new substructure
8	The PLTS item for the new substructure
9	The LODS Item for the new substructure
10	Modal dof set summary
11	Fixed set summary
12	Summary of GRID point identification numbers in each fixed set

6. You may perform a **CREDUCE** operation in both PHASE 1 and PHASE 2 and at any level of the substructuring process.

Example:

1. Perform a **CREDUCE** of substructures **FUSLAGE** resulting in a new substructure called **RFUSE**. Assume that a boundary set is specified by **BDYC** Bulk Data entries having an identification number of **100**, and use all eigenvectors whose frequencies are between **0.0** and **1000.0** Hz as generalized coordinates.

```
CREDUCE FUSLAGE
   NAME = RFUSE
   BOUNDARY = 100
   RANGE = 0.0,1000.0
```

Substructure Command DELETE

Deletes individual substructure items from the SOF Database.

Command Syntax:

```
DELETE  name item1 [,item2,...,item5]
```

Subcommands: None

Option	*Meaning*	*Data Range*
name	Specifies the name of the substructure for which items will be deleted. [1]	Name
itemi	Specifies the SOF Database item names that will be deleted. [2]	Name

Remarks:

1. Any secondary substructures or image substructure which have been created by an **EQUIV** of substructure *name* are unaffected by the **DELETE** operation.
2. You may select subscripted item names by specifying:

   item[*sub_val*] or *item*[*]

   where *sub_val* is the selected subscript and * is a wildcard signifying all subscripts.
3. See also the **EDIT** and **DESTROY** commands.

Example:

1. Remove the items **SOLN**, **PVEC[3]**, and **HORG** of substructure **FUSLAGE** from the SOF.

```
DELETE FUSLAGE SOLN,PVEC[3],HORG
```

Substructure Command DESTROY

Removes data for a substructure, and all substructures of which it is a component, from the SOF Database.

Command Syntax:

DESTROY *name*

Subcommands: None

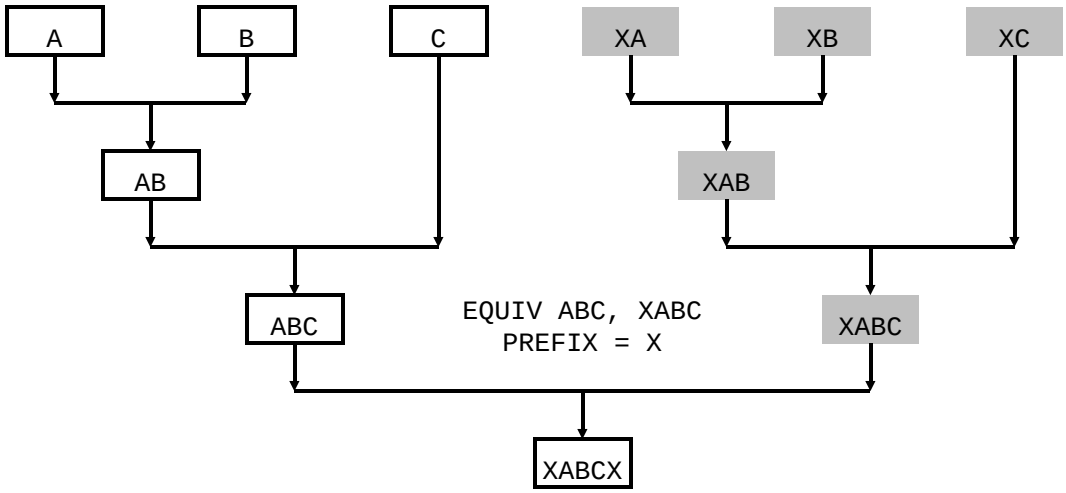
Option	*Meaning*	*Data Range*
name	Specifies a substructure name whose data will be destroyed. [1,2]	Name

Remarks:

1. *name* may not reference an image substructure.
2. If substructure *name* includes subscripted SOF items, all items are destroyed with this command. You use the **DELETE** command to selectively remove data.
3. In addition to the substructure being destroyed, *name*, the following data are also removed from the SOF Database:
   - ☐ All substructures of which *name* is a component.
   - ☐ All secondary substructures and their associated image substructures for which *name* is the primary substructure.
   - ☐ All secondary substructures and their associated image substructures of which *name* is a component.
3. See also the **DELETE** and **EDIT** commands.

Examples:

Consider the following substructuring analysis tree:



1. Referencing the substructure tree shown above, the command:

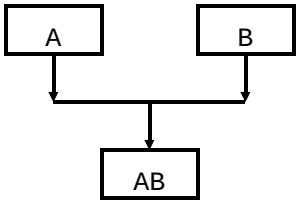
   DESTROY A

   will result in the removal of all substructures in the tree with the exception of the Basic Substructures **B** and **C**.

2. Again referencing the substructure tree, the command:

   DESTROY C

   Results in the following tree:



Substructure Commands ECHOOFF and ECHOON

Disables or enables the normal echo of Substucture commands.

Command Syntax:

```
ECHOOFF
ECHOON
```

Remarks:

1. The echo of consecutive Substructure commands may be removed from the print file by insertion of the **ECHOOFF** command into the Substructure Control packet. Echo may be reactivated by use of the **ECHOON** command.
2. As many pairs of **ECHOOFF**, **ECHOON** commands may be used as desired.
3. **ECHOON** is automatically invoked for the Case Control packet at the completion of processing the Substructure Command packet. **ECHO** of the Case Control packet is then controlled by Case Control commands **ECHOOFF** and **ECHOON**.

Substructure Command EDIT

Removes selected substructure data from the SOF Database.

Command Syntax:

```
EDIT (code) name
```

Subcommands: None

Option	*Meaning*	*Data Range*
name	Specifies the substructure from which data items will be removed.	Name
code	Specifies a code defining the SOF Database items to be edited. [1,2]	Integer>0

Remarks:

1. The table below defines codes for editing the indicated SOF Database items. To combine two or more requests, simply add the codes together.

code	ITEMS REMOVED
1	Stiffness matrix – KMTX
2	Mass matrix – MMTX
4	Load data – LODS, LOAP, PVEC, PAPP
8	Solution data – UVEC, QVEC, SOLN
16	Transformation matrices defining next level – HORG, UPRT, POVE, POAP, LMTX, GIMS, HLFT
32	All items for the substructure
64	Appended load data – LOAP, PVEC, PAPP
128	Damping matrices – K4MX, BMTX
256	Modal reduction data – LAMS, PHIS, PHIL
512	Total transforms only – HORG, HLFT

2. If substructure *name* includes subscripted SOF items, all items are deleted with this command.
3. You are cautioned not to remove the transformation matrix data. These matrices are required for the recovery of the solution results.
4. To insure the integrity of your SOF Database, you should consider also using the **SOFOUT** utility prior to performing any **EDIT** operations
5. See also the **DELETE** and **DESTROY** commands.

Substructure Command EQUIV

Creates a new substructure, or substructure tree, as a copy of an existing one.

Command Syntax:

```
EQUIV pname,ename
```

Subcommands:

```
{ PREFIX } = p_or_s_char
{ SUFFIX }

GRIDOFFSET = offset
```

Option	*Meaning*	*Data Range*
pname	Specifies an existing substructure name. [1]	Name
ename	Assigns a new substructure name. [2]	Name
p_or_s_char	Specifies a single letter used as the prefix or suffix for the image substructures created. [3]	Character
offset	Specifies an increment that will be added to all GRID point identification numbers in the secondary substructure, and any image substructures generated by the operation. [5]	Integer>0

Remarks:

1. The existing substructure is called the ***primary substructure***.
2. The new substructure created by this command is called a ***secondary substructure***. If *pname* has been created by other substructuring operations, then all of its component substructures and their data items are also copied. The component substructures of the secondary substructure are called ***image substructures***. Any number of secondary substructures may be equivalenced to a primary substructure.
3. If you do not specify a **PREFIX** or **SUFFIX**, then the prefix letter **E** will be used by default. Remember that if *pchar* is a prefix, then it is added as the first character of the name of each image substructure. If these names become longer than eight characters, they are truncated on the right. When *pchar* is a suffix, it is added to the end of the name of each substructure. If the name becomes longer than eight characters, then the rightmost character is replaced by the **SUFFIX** and a warning message is issued. ***Repeated use of* EQUIV *commands may lead to duplicate substructure names, proceed with caution.***
4. If you have performed one of the **SOLVE** operations on the primary substructure, the SOF Database solution items, **SOLN**, **UVEC**, and **QVEC**, are not copied for the secondary substructure or any of its images.
5. The **GRIDOFFSET** subcommand allows you to modify the external GRID point identification numbers for the substructure. This is used to insure unique identification numbers during a PHASE 2 **SOLVE** operation when new model or loads data are being added to the model.

Substructure Command MASSBAL

Requests weight and mass equilibrium data to be calculated and printed for a substructure.

Command Syntax:

```
MASSBAL  name
```

Subcommands:

```
RNAME = refname
RGRID = pointid
```


Option	*Meaning*	*Data Range*
name	Selects a substructure for which the weight will be computed.	Name
refname	Specifies a reference Basic Substructure which is a component of substructure *name*.	Name Default=*name*
pointid	Specifies the identification number of a reference GRID point contained in substructure *refname*.	Integer>0 Default=0

Remarks:

1. The rigid body mass properties are calculated with respect to the origin of the substructure Basic coordinate system or with respect to the specific reference GRID point, which is part of a component Basic Substructure.
2. If **PARAM,WTMASS** is specified, it is used by the **MASSBAL** command. The substructure mass matrix will be scaled by the inverse of the **WTMASS** parameter for the weight and mass equilibrium calculations only.

Substructure Command MRECOVER

Recovers modal displacements and boundary forces for substructures reduced to modal coordinates.

Command Syntax:

MRECOVER *reduced_name*

Subcommands:

{ **SAVE** / **PRINT** } = *comp_name*

SORT = { **MODES** / SUBSTRUCTURE }

{ **DISP** / **SPCF** } [([{ **PRINT** / NOPRINT } ,] [PUNCH,] [{ **RECT** / POLAR } ,] [{ **SORT1** / SORT2 }])] = { ALL / *sid* / **NONE** }

MODES = { **ALL** / *sid* }

RANGE = *f1* [,*f2*]

BASIC = *bas_name*

UIMPROVE

Option	*Meaning*	*Data Range*
reduced_name	Specifies the name of a reduced substructure from which the solution results are to be recovered. [1]	Name
comp_name	Specifies the name of a component substructure for which the solution results are to be recovered. [2,3,4,5]	Name
{ **MODES** / SUBSTRUCTURE }	Selects the solution results sort order. [6,7]	Character
{ ALL / *sid* / **NONE** }	Selects **DISP**lacement or **SPCF**orce output for **ALL** points, or provides a set identification number *sid* which contains a list of the points for which output will be generated. If **NONE** is specified, output will not be generated. [8,9]	Keyword or Intger>0
{ **ALL** / *sid* }	Selects either **ALL** modes, or provides a set identification number *sid* which contains a list of the modes for which output will be generated. [10]	Keyword or Intger>0
f1[,*f2*]	Specifies a range of frequencies for which output will be generated. [10,11]	Real>0.0

bas_name	Specifies the name of a Basic Component of *reduced_name* to which specific output requests apply. [12]	Name

Remarks:

1. An **MRECOVER** operation may only be performed on a substructure which has been reduced using the **MREDUCE** or **CREDUCE** commands.
2. **SAVE** will save the recovered solution results for substructure *comp_name* on the SOF Database. **PRINT** will save and print the solution results.
3. If output requests are not present, the **PRINT** subcommand is equivalent to **SAVE** and output will not be printed.
4. If the solution results already exist on the SOF Database, they may be printed without the cost of regeneration with the **PRINT** subcommand.
5. For efficiency, you should order multiple **SAVE** or **PRINT** subcommands so as to trace one branch of the substructuring tree at a time, starting from the solution structure.
6. If **MODES** is specified, all output requests for each mode will appear together. If **SUBSTRUCTURE** is specified, all output requests for each Basic Substructure will appear together.
7. The **SORT** subcommand should only appear after a **PRINT** subcommand. Any **SORT** subcommands appearing after a **BASIC** subcommand will be ignored.
8. All set definitions must appear in the Case Control packet.
9. You may specify print thresholds for all printout. If the absolute value is less than the threshold, the value will be set to zero. The following thresholds may be specified using **PARAM** Bulk Data entry:

   - **UTHRESH** - displacement, velocity and acceleration threshold.
   - **PTHRESH** - load threshold.
   - **QTHRESH** - reaction force threshold.

10. If both a **MODES** subcommand and a **RANGE** subcommand are present, all restrictions must be satisfied for any output to be generated.
11. If only *f1* is present, the frequency range is assumed to be zero to *f1*.
12. The output requests appearing after a **BASIC** subcommand are honored specifically for the corresponding *bas_name*, and override all other similar requests.
13. If the **UIMPROVE** request is present, an improved displacement vector will be generated. This vector will contain the effects of inertia and damping forces.

Example:

```
MRECOVER SOLSTRCT
  PRINT ABDC
    SORT = SUBSTRUCTURE
    DISP = ALL $ Provides defaults for ABDC output
    BASIC A
      DISP = 5 $ Provides specific requests for Basic Substructure A
    BASIC C
      SPCF = 10 $ Provides specific requests for Basic C
  SAVE ABC
```

Substructure Command MREDUCE

Performs a modal synthesis reduction on a specified component substructure.

Command Syntax:

```
MREDUCE  name
```

Subcommands:

```
♦  NAME       = newname
♦  BOUNDARY   = bsid
   FIXED      = fsid
♦  METHOD     = methid
   RANGE      = f1,f2
   NMAX       = nmodes
   GRIDOFFSET = offset
   RESIDUAL
   RSAVE
   OUTPUT     = opt1 [,opt2,...]
```

Option	*Meaning*	*Data Range*
name	Specifies the name of the substructure to be reduced.	Name
newname	Assigns a name to the resulting substructure.	Name
bsid	Selects the identification number of **BDYC** Bulk Data which define one or more sets of boundary degrees of freedom. [1,3]	Integer>0
fsid	Selects the identification number of **BDYC** Bulk Data which define one or more sets of degrees of freedom temporarily fixed during modal extraction. [2,3]	Integer>0
method	Selects an **EIGR** Bulk Data entry which controls the eigenvalue extraction. [4]	Integer>0
f1,f2	Specifies a frequency range, in Hz, for selecting modal coordinates. [4]	Real
nmodes	Specifies the number of modal coordinates to be selected. [4]	Integer>0
offset	Specifies an offset value to be added to all GRID point identification numbers that are created for the modal degrees of freedom. [5]	Integer>0
RESIDUAL	Requests addition of inertia relief deflection shapes. [6]	Keyword
RSAVE	Requests that the LMTX item be saved on the SOF Database. [7]	Keyword
opti	Requests selected output. [8]	Integer>0

Remarks:

1. The new substructure, *newname*, will contain only those physical degrees of freedom which are included in the boundary set, *bsid*.
2. The degrees of freedom in the **FIXED** set, *fsid*, are constrained only during the modal extraction process.
3. The reduced substructure, *newname*, is defined by the union of the physical boundary degrees of freedom, the generalized coordinates computed by the modal extraction subject to the fixed degrees of freedom, and any interia relief generalized coordinates. See the ***UAI/NASTRAN User's Guide*** for a complete discussion of **MREDUCE**.
4. When you specify a natural frequency **RANGE**, then all of the eigenvectors of modes within the range will be used as generalized coordinates for the reduction. There must be at least one frequency in the range. Similarly, if you specify **NMAX**, then the first *nmodes* flexible eigenvectors will be used as generalized coordinates unless there are actually fewer extracted by the selected eigenvalue procedure. In this case, the number extracted will be used. If you do not use either of these commands, then all of the modes extracted, based on you **EIGR** Bulk Data entry, will be used.
5. The **GRIDOFFSET** subcommand allows you to specify the new external GRID point identification numbers for the substructure. This is used to insure unique identification numbers during a PHASE 2 **SOLVE** operation when new model or loads data are being added to the model.
6. If you use the **RESIDUAL** subcommand, the six rigid body motions about the overall Basic coordinate system of *name* are used to define the inertia relief deflection shapes. These shapes are used as generalized coordinates in addition to the modal coordinates.
7. An important result of the **MREDUCE** operation is the lower triangular factor, LMTX, of the decomposed matrix representing the degrees-of-freedom which are omitted during the reduction process. This matrix is both costly to compute and its storage often represents more than half of the space on your SOF Database. If your host computer has a large amount of available disk space, then you may use the **RSAVE** subcommand to save the LMTX on the database. On the other hand, if disk space is at a premium, you do not use the **RSAVE** command. Later, if you are performing data recovery for the omitted points, the matrix decomposition will be redone as necessary.
8. The following output is available for the **MREDUCE** operation. Shaded options are recommended for general use.

opti	OUTPUT ITEM	*opti*	OUTPUT ITEM
1	Current problem summary	7	The CSTM item
2	Boundary set summary	8	The PLTS item
3	Summary of GRID point identification numbers in each boundary set	9	The LODS Item
4	The EQSS item for the structure being reduced	10	Modal dof set summary
5	The EQSS item	11	Fixed set summary
6	The BGSS item	12	Summary of GRID point identification numbers in each fixed set

You may perform an **MREDUCE** operation in both PHASE 1 and PHASE 2 and at any level of the substructuring process.

Substructure Command OPTIONS

Selects the matrices to be created during the current job.

Command Syntax:

OPTIONS *opt*1 [, *opt*2 ...]

Subcommands: None

Option	*Meaning*	*Data Range*
opti	Selects one or more matrix options. [1,2]	Keyword

Remarks:

1. The available matrix options are shown in the table below:

opti	DESCRIPTION	*opti*	DESCRIPTION
K	Stiffness Matrices	PA	Appended Load Vectors
M	Mass Matrices	B	Viscous Damping Matrices
P	Load Matrices	K4	Structural Damping Matrices

2. The default depends on the **UAI/*NASTRAN*** Rigid Format as given in the following table:

RIGID FORMAT	DEFAULT
1	K,P
2	K,M,P
3	K,M
8	K,M,P,B,K4
9	K,M,P,B,K4

3. The matrix options that you select with this command remain enabled for all subsequent substructuring operations until they are changed.
4. The use of the **OPTION** command is discussed in detail in the **UAI/*NASTRAN User's Guide***.

Substructure Command PLOT

Requests undeformed plots of a substructure which may be composed of several component substructures.

Command Syntax:

PLOT *name*

Subcommands: None

Option	*Meaning*	*Data Range*
name	Selects the substructure to be plotted.	Name

Remarks:

1. **PLOT** commands must be included in the usual manner in the Case Control command packet.

Substructure Command RECOVER [PHASE 2]

Recovers displacements and boundary forces on specified substructures in a PHASE 2 execution.

Command Syntax:

RECOVER *sub_name*

Subcommands:

3 SUBS

For all disciplines:

SUBSCRIPT = *sub_val*
SAVE = *comp_name*
PRINT = *comp_name*
BASIC = *bas_name*

{ **DISP** / **SPCF** / **OLOAD** } [([{ **PRINT** / NOPRINT } ,] [PUNCH,] [{ **RECT** / POLAR } ,] [{ **SORT1** / SORT2 }])] = { ALL / *sid* / **NONE** }

For Static Analysis only:

SORT = { **SUBCASE** / SUBSTRUCTURE }

SUBCASES = { **ALL** / *case_set_id* / NONE }

For Normal Modes only:

SORT = { **MODES** / SUBSTRUCTURE }

MODES = { **ALL** / *case_set_id* / NONE }

RANGE = *f1* [,*f2*]

For all Dynamic Response Disciplines:

{ **VELO** / **ACCE** } [([{ **PRINT** / NOPRINT } ,] [PUNCH,] [{ **RECT** / POLAR } ,] [{ **SORT1** / SORT2 }])] = { ALL / *sid* / **NONE** }

UIMPROVE [14]

For Transient Response Analyses:

SORT = { **TIME** / SUBSTRUCTURE }

STEPS = { **ALL** / *time_set_id* / NONE }

RANGE = *t1* [,*t2*]

For Frequency Response Analyses:

SORT = { **FREQ** / SUBSTRUCTURE }

STEPS = { **ALL** / *freq_set_id* / NONE }

RANGE = *f1* [,*f2*]

SUBCASES = { **ALL** / *case_set_id* / NONE }

Option	*Meaning*	*Data Range*
sub_name	Name of a previously **SOLVE**d substructure from which the solution results are to be recovered.	Name
sub_val	Subscript value.	Integer
comp_name	Name of a component substructure of *sub_name* for which results are to be recovered. [1,2,3,4,5]	Name
{**ALL** / *out_set_id* / NONE}	Selects requested output for **ALL** points, or provides a set identification number *out_set_id* which contains a list of the points for which output will be printed. If **NONE** is specified, then output is not generated. [6,7,8]	Keyword or Integer>0
bas_name	Specifies the name of a component Basic Substructure of the selected *comp_name* to which specific output requests apply. [9]	Name
{**SUBCASE** / SUBSTRUCTURE}	Selects output sort order for Static analyses. [10,12]	Keyword
{**MODES** / SUBSTRUCTURE}	Selects output sort order for Normal Modes analyses. [10,12]	Keyword
{**TIME** / SUBSTRUCTURE}	Selects output sort order for Transient Response analyses. [11,12]	Keyword
{**FREQ** / SUBSTRUCTURE}	Selects output sort order for Frequency Response analyses. [11,12]	Keyword
{**ALL** / *case_set_id* / NONE}	Selects, for Static or Frequency Response analyses, subcases for which results will be **SAVE**d or **PRINT**ed. You may select **ALL** subcases or you may specify the identification number of a set which gives a list of specific subcases. If **NONE** is selected, then output will not be generated.	Keyword or Integer>0
{**ALL** / *mode_set_id* / NONE}	Selects, for Normal Modes analyses, modes for which results will be **SAVE**d or **PRINT**ed. You may select **ALL** modes or you may specify the identification number of a set which gives a list of specific modes. If **NONE** is selected, then output will not be generated.	Keyword or Integer>0
{**ALL** / *time_set_id* / NONE}	Selects, for Transient Response analyses, times for which results will be **SAVE**d or **PRINT**ed. You may select **ALL** times or you may specify the identification number of a set which gives a list of specific times. If **NONE** is selected, then output will not be generated. [13]	Keyword or Integer>0
{**ALL** / *freq_set_id* / NONE}	Selects, for Frequency Response analyses, frequencies for which results will be **SAVE**d or **PRINT**ed. You may select **ALL** frequencies or you may specify the identification number of a set which gives a list of specific frequencies. If **NONE** is selected, then output will not be generated. [13]	Keyword or Integer>0
f1[,f2]	Selects, for Normal Modes or Frequency Response analyses, a range of frequencies for which solution results will be **SAVE**d or **PRINT**ed. If only *f1* is present, the frequency range is assumed to be zero to *f1*. [13]	Real>0.0

t1[,t2]	Selects, for Transient Response analyses, a range of times for which solution results will be **SAVE**d or **PRINT**ed. If only *t1* is present, the time range is assumed to be zero to *t1*. [13]	Real>0.0

Remarks:

1. *comp_name* may be the same as *sub_name*.
2. **SAVE** will save the recovered solution results for substructure *comp_name* on the SOF Database. **PRINT** will save and print the solution results.
3. If output requests are not present, the **PRINT** subcommand is equivalent to **SAVE** and output will not be printed.
4. If the solution results already exist on the SOF Database, they may be printed without the cost of regeneration with the **PRINT** subcommand.
5. For efficiency, you should order multiple **SAVE** or **PRINT** subcommands so as to trace one branch of the substructuring tree at a time, starting from the solution structure.
6. For dynamic analysis, the printed loads output will include dynamic loads only for the solution substructure in the same run that the solution was obtained. For any lower level substructures or on any run after the solution, only static loads will be printed.
7. All set definitions must appear in the Case Control packet.
8. You may specify print thresholds for all printout. If the absolute value is less than the threshold, the value will be set to zero. The following thresholds can be input using **PARAM** Bulk Data entries:

   - **UTHRESH** - Displacement, velocity, and acceleration.
   - **PTHRESH** - Load threshold.
   - **QTHRESH** - Reaction force threshold.

9. The output requests appearing after a **BASIC** subcommand are honored specifically for the corresponding *bas_name*, and override all other similar requests.

10. When you are performing Static or Normal Modes analyses, you must specify the **SORT1** sort order. You may control further sorting with the **RECOVER** subcommand **SORT**. The sort order of results requested in this manner are shown below:

STATIC ANALYSES		NORMAL MODES ANALYSES	
SORT =		SORT =	
SUBSTRUCTURE	SUBCASE	SUBSTRUCTURE	MODES
SUBSTRUCTURE A SUBCASE 1 GRID 1 GRID 2 ... SUBCASE 2 ... ... SUBSTRUCTURE B	SUBCASE 1 SUBSTRUCTURE A GRID 1 GRID 2 ... SUBSTRUCTURE B ... ... SUBCASE 2	SUBSTRUCTURE A MODE 1 GRID 1 GRID 2 ... MODE 2 ... ... SUBSTRUCTURE B	MODE 1 SUBSTRUCTURE A GRID 1 GRID 2 ... SUBSTRUCTURE B ... ... MODE 2 ... ...

11. When you are performing Transient Response or Frequency Response analyses, you must specify the **SORT2** sort order. You may control further sorting with the **RECOVER** subcommand **SORT**. The sort order of results requested in this manner are shown below:

TRANSIENT RESPONSE ANALYSES		FREQUENCY RESPONSE ANALYSES	
SORT =		SORT =	
SUBSTRUCTURE	TIME	SUBSTRUCTURE	MODES
SUBSTRUCTURE A TIME 1 GRID 1 GRID 2 ... TIME 2 ... ... SUBSTRUCTURE B	TIME 1 SUBSTRUCTURE A GRID 1 GRID 2 ... SUBSTRUCTURE B ... ... SUBCASE 2	SUBSTRUCTURE A MODE 1 GRID 1 GRID 2 ... MODE 2 ... ... SUBSTRUCTURE B	MODE 1 SUBSTRUCTURE A GRID 1 GRID 2 ... SUBSTRUCTURE B ... ... MODE 2 ... ...

12. The **SORT** subcommand should only appear after a **PRINT** subcommand. Any **SORT** commands appearing after a **BASIC** subcommand will be ignored.

13. If both a **MODES** (or **STEPS**) subcommand and a **RANGE** subcommand appear for dynamic response analyses, both requests must be satisfied for any output to be generated.

14. If the **UIMPROVE** request is present, an improved displacement vector will be generated. This vector will contain the effects of inertia and damping forces.

Examples:

1. Assume that you have **SOLVE**d substructure **X**, which has components **A** and **B**, in a previous execution. Recover solution results and print all of the displacements and applied loads for the component substructures:

```
...
SUBSTRUCTURE PHASE2
   RECOVER X
      PRINT A
         DISP = ALL
         OLOAD = ALL
      PRINT B
         DISP = ALL
         OLOAD = ALL
ENDSUBS
BEGIN BULK
...
```

2. Using the same information in example 1, suppose you want to print the displacements in **SET 101** for component **B** and do not want the applied loads printed for component **A**:

```
...
SUBSTRUCTURE PHASE2
   RECOVER X
   PRINT A
      DISP = ALL
      OLOAD = NONE   Required to override OLOAD=ALL
   PRINT B
      DISP = 101    Required to override DISP=ALL
      OLOAD = ALL
ENDSUBS
SET 101 = 1 THRU 100
BEGIN BULK
...
```

Substructure Command RECOVER [PHASE 3]

Identifies the Basic Substructure for a PHASE 3 execution.

Command Syntax:

RECOVER = *name*

Subcommands:

SUBSCRIPT = { *sub_val* / **ALL** }

CASEID = { **PHASE2** / ORDINAL / POSITIONAL }

OFFSET = *off_id*

OUTPUT = *opt1* [,*opt2*,...]

Option	*Meaning*	*Data Range*
name	Specifies the name of the Basic Substructure for which detailed data recovery is being performed in a PHASE 3 operation. [1,2,3]	Name
sub_val	If multiple solutions have been performed, i.e. you have used the **SUBSCRIPT** subcommand of PHASE 2 **RECOVER**, then *sub_val* specifies the solution to be recovered. To recover all solutions, you use the **ALL** option, or you may use multiple **SUBSCRIPT** commands.	Integer>0
{ **PHASE2** / ORDINAL / POSITIONAL }	Specifies how the PHASE 3 SUBCASE identification numbers are interpreted. They may be the **SUBCASE** identification numbers from PHASE 2, or will reference the numeric, or **ORDINAL**, sequence of the PHASE 2 solutions, or they may be in a one-to-one correspondence with them. [4,5]	Keyword
off_id	Offset to be added to the SUBCASE identification numbers from the PHASE 2 solution if **CASEID=PHASE2** is selected. [5]	Integer>0
opti	Requests selected output. [6]	Integer>0

Remarks:

1. A separate PHASE 3 execution is required for each Basic Substructure for which *g-set* and element results are desired. However, you may also perform data recovery for any, or all, basic substructures which were **EQUIV**alenced to name in the same run by using multiple **RECOVER** commands.

2. The PHASE 3 execution may be performed as a RESTART from a previously CHECKPOINTED PHASE 1 execution, or it may be performed using the Bulk Data input file used previously in a PHASE 1 execution.

3. If *name* is an Image Substructure, either RESTART from the primary Basic Substructure CHECKPOINT data or use the primary Basic Substructure Bulk Data packet.

4. There are three ways to reference the SUBCASEs during the PHASE 3 recovery operations. The first option, which is the default, is **PHASE2**. This means that the PHASE 3 SUBCASEs reference the PHASE 2 SUBCASE identification numbers directly.

   The second, or **ORDINAL**, option indicates that the SUBCASEs will be selected in the order which they appeared in the PHASE 2 solution. The PHASE 2 SUBCASE identification numbers are ignored, instead the PHASE 2 solutions are referenced in PHASE 3 by SUBCASE identification numbers which are ordered from 1 to n. Note that omitting **SUBCASE 2**, for example, results in no output for the second PHASE 2 solution.

   The third and final option, **POSITIONAL**, is available for compatibility with versions of **UAI/*NASTRAN*** prior to 11.8. With this option, the PHASE 3 SUBCASE identification numbers are irrelevant. The first SUBCASE simply corresponds to the first PHASE 2 solution vector, the second SUBCASE to the second vector, and so on.

5. In some cases, there may be duplicated SUBCASE identification numbers in different solutions. Then the special subcommand, **OFFSET**, allows you to reassign the PHASE 3 SUBCASE identification numbers so that they are unique. (See Example 2 below)

6. The following output request is available for the PHASE 3 RECOVER operation

opti	OUTPUT ITEM
1	Prints a table of the PHASE2 and PHASE3 SUBCASE Identification number relationships.

Examples:

1. Consider the following two PHASE 2 **SOLVE** and **RECOVER** operations for substructure **NAME1**:

FIRST SOLUTION	SECOND SOLUTION
... **SUBSTRUCTURE PHASE 2** **SOLVE** NAME1 **RECOVER** NAME1 **SUBSCRIPT** = 1 **SAVE** NAME1 **ENDSUBS** **SUBCASE** 100 **LOAD** = 100 **SUBCASE** 200 **LOAD** = 200 ...	... **SUBSTRUCTURE PHASE 2** **SOLVE** NAME1 **RECOVER** NAME1 **SUBSCRIPT** = 2 **SAVE** NAME1 **ENDSUBS** **SUBCASE** 300 **LOAD** = 300 **SUBCASE** 400 **LOAD** = 400 ...

The PHASE 3 recovery for all solution data for each of the options described above are shown in the following table.

WHEN CASEID IS:		
PHASE2	**ORDINAL**	**POSITIONAL**
SUBSTRUCTURE PHASE 3 **RECOVER** NAME1 **SUBSCRIPT** = ALL **ENDSUBS** **SUBCASE** 100 ... **SUBCASE** 200 ... **SUBCASE** 300 ... **SUBCASE** 400	**SUBSTRUCTURE PHASE 3** **RECOVER** NAME1 **CASEID** = ORDINAL **SUBSCRIPT** = ALL **ENDSUBS** **SUBCASE** 1 ... **SUBCASE** 2 ... **SUBCASE** 3 ... **SUBCASE** 4	**SUBSTRUCTURE PHASE 3** **RECOVER** NAME1 **CASEID** = POSITIONAL **SUBSCRIPT** = ALL **ENDSUBS** **SUBCASE** 11 ... **SUBCASE** 201 ... **SUBCASE** 333 ... **SUBCASE** 456 ... ...

2. This example shows the use of the **OFFSET** subcommand. Consider the following two PHASE 2 **SOLVE** and **RECOVER** operations for substructure **NAME1** consisting of two identical halves named **LEFT** and **RIGHT**, where **RIGHT** is an image of **LEFT** created by the **EQUIV** operation, and solution results from both of these substructures is desired in one PHASE 3 execution:

FIRST SOLUTION	SECOND SOLUTION
... **SUBSTRUCTURE PHASE 2** **SOLVE** NAME1 **RECOVER** NAME1 **SUBSCRIPT** = 1 **SAVE** LEFT **SAVE** RIGHT **ENDSUBS** **SUBCASE** 100 **LOAD** = 100 **SUBCASE** 200 **LOAD** = 200 ...	... **SUBSTRUCTURE PHASE 2** **SOLVE** NAME1 **RECOVER** NAME1 **SUBSCRIPT** = 2 **SAVE** LEFT **SAVE** RIGHT **ENDSUBS** **SUBCASE** 300 **LOAD** = 300 **SUBCASE** 400 **LOAD** = 400 ...

There are four solution vectors for **LEFT** and four for **RIGHT** arising from these solutions. This results in duplicate SUBCASE identification numbers. The **OFFSET** subcommand then allows you to reference the PHASE 2 SUBCASE identification numbers with unique identifiers.

This is shown below:

```
SUBSTRUCTURE PHASE 3
RECOVER LEFT
  SUBSCRIPT = ALL
RECOVER RIGHT
  SUBSCRIPT = ALL
  OFFSET = 1000
ENDSUBS
$
$   THE FOLLOWING THREE SUBCASES ARE FOR "LEFT"; NOTE THAT SUBCASE
$   300 RESULTS ARE NOT REQUESTED FOR "LEFT"
$
SUBCASE 100
   ...
SUBCASE 200
   ...
SUBCASE 400
   ...
$
$   THE FOLLOWING THREE SUBCASES ARE FOR "RIGHT" THE VALUE OF
$   "OFFSET" HAS BEEN ADDED TO THE PHASE2 SUBCASE IDENTIFICATION
$   NUMBERS. NOTE THAT THE PHASE2 SUBCASE 200 (1200 INCLUDING THE
$   OFFSET) IS NOT REQUESTED FOR "RIGHT"
$
SUBCASE 1100
   ...
SUBCASE 1300
   ...
SUBCASE 1400
   ...
...
```

Substructure Command REDUCE

Performs a Guyan reduction, or static condensation, on a specified substructure.

Request Command Syntax:

```
REDUCE  name
```

Subcommands:

```
♦  NAME     = newname
♦  BOUNDARY = bsid
   OUTPUT   = opt1 [,opt2,...]
   RSAVE
```


Option	*Meaning*	*Data Range*
name	Specifies the name of the substructure to be reduced.	Name
newname	Assigns a name to the resulting substructure.	Name
bsid	Selects the identification number of **BDYC** Bulk Data which define one or more sets of boundary degrees of freedom. [1,3]	Integer>0
RSAVE	Requests that the LMTX item be saved on the SOF Database. [3]	Keyword
opti	Requests selected output. [2]	Integer>0

Remarks:

1. The new substructure, *newname*, will contain only those physical degrees of freedom which are included in the boundary set, *bsid*.
2. The following output requests are available for the **REDUCE** operation. Shaded options are recommended for general use.

opti	OUTPUT ITEM	*opti*	OUTPUT ITEM
1	Current problem summary	5	The EQSS item
2	Boundary set summary	6	The BGSS item
3	Summary of GRID point identification numbers in each boundary set	7	The CSTM item
4	The EQSS item for the reduced structure	8	The PLTS item

3. An important result of the **REDUCE** operation is the lower triangular factor, LMTX, of the decomposed matrix representing the degrees-of-freedom which are omitted during the reduction process. This matrix is both costly to compute and its storage often represents more than half of the space on your SOF Database. If your host computer has a large amount of available disk space, then you may use the **RSAVE** subcommand to save the LMTX on the Database. On the other hand, if disk space is at a premium, you do not use the **RSAVE** command. Later, if you are performing data recovery for the omitted points, the matrix decomposition will be redone as necessary.

Substructure Commands SKIPON and SKIPOFF

Disables and enables the processing of Substructure Control commands.

Command Syntax:

```
SKIPON
SKIPOFF
```

Remarks:

1. The processing of Substructure Control commands is suspended when the **SKIPON** command is encountered in the Substructure Control packet. All following commands are echoed but not processed until a **SKIPOFF** command is encountered. Echoing of skipped commands may be controlled by the **ECHOOFF** and **ECHOON** commands.
2. As many pairs of **SKIPON, SKIPOFF** commands may be used as desired.
3. **SKIPOFF** is automatically invoked for the Case Control packet at the completion of processing the Substructure Command packet. Skipping of data in the Case Control packet is then controlled by the Case Control commands **SKIPON** and **SKIPOFF**.

Substructure Command SOFIN

Copies all or selected portions of an SOF Database from an import file which was created on another computer of the same or a different type.

Command Syntax:

```
SOFIN [ ( { INTERNAL } ) ] logical_name
      [ ( { EXTERNAL } ) ]
```


Subcommands:

```
POSITION = { REWIND     }
           { NOREWIND   }

ITEMS    = { ALL        }
           { MATRICES   }
           { PHASE3     }
           { TABLES     }
           { item_name  }

NAMES    = { sub_name   }
           { WHOLESOF   }
```

Option	*Meaning*	*Data Range*
{ INTERNAL } { **EXTERNAL** }	Specifies whether the import file was exported from a different, **EXTERNAL**, computer type or the same, **INTERNAL**, computer type.	Keyword
logical_name	Logical name of the import file. [1]	Character
{ REWIND } { **NOREWIND** }	Specifies the initial file position. You may **REWIND** the file prior to reading or you may begin reading at the current position with **NOREWIND**. [2]	Keyword
{ **ALL** } { MATRICES } { PHASE3 } { TABLES } { *item_name* }	Identifies the data items which are to be copied from the import file for the substructures, *name*, specified in all **NAMES** subcommands. You may select **ALL** items, all **MATRICES**, all **TABLES**, the **PHASE3** items UVEC, QVEC and SOLN, or individual *item_name*s. [3]	Keyword or Character
{ *sub_name* } { WHOLESOF }	Identifies a substructure for which data will be read. You may select the entire SOF Database, **WHOLESOF**, or a specific substructure name, *sub_name*. [3]	Keyword or Name

Remarks:

1. You must use the **ASSIGN** Executive Control command to define the *logical_name* with **USE=SOFIN**.

2. When you are reading all of an import file, you must use **POSITION=REWIND**. If you select the **EXTERNAL** file format, then the subcommand **POSITION=NOREWIND** results in the positioning of the file to the end-of-file.

3. You may include as many as five **NAMES** subcommands for each **SOFIN** command, where each references a different substructure name. The **ITEMS** that you have selected apply to **all** of these substructures. If you wish to write different **ITEMS** for different substructures, then you must use different **SOFIN** commands.

4. The files that you export with **SOFOUT** may be imported using the **SOFIN** Substructure Control command.

Examples:

1. Import a file which has been exported from a different computer type and contains your entire SOF Database:

```
ASSIGN SOF1=MYSOF.DAT,NEW,REALLOC,USE=SOF
ASSIGN IMPORT=IMSOF.DAT,OLD,USE=SOFIN,TYPE=FORMATTED
...
SOFIN IMPORT
   POSITION = REWIND
```

   Note that for importing in the **EXTERNAL** format you must **ASSIGN** the logical file with the **TYPE=FORMATTED** parameter.

2. Import a file which has been exported from the same computer type, and contains all of the SOF Database for substructures **WING** and **TAIL**:

```
ASSIGN SOF1=MYSOF.DAT,OLD,USE=SOF
ASSIGN IMPORT=IMSOF.DAT,OLD,USE=SOFIN,TYPE=BINARY
...
SOFIN (INTERNAL) IMPORT
   POSITION = REWIND
   NAMES = WING
   NAMES = TAIL
```

   Because the same items are being imported for both substructures, only one **SOFIN** command is required. Also notice that for importing in the **INTERNAL** format that you must **ASSIGN** the logical file with the **TYPE=BINARY** parameter.

3. Import a file which was created in internal format, and contains the solution items for **WING** and the table items for **TAIL**:

```
ASSIGN SOF1=MYSOF.DAT,OLD
ASSIGN IMPORT=IMSOF.DAT,OLD,USE=SOFIN
...
SOFIN (INTERNAL) IMPORT
   POSITION = REWIND
   NAMES = WING
   ITEMS = SOLN
SOFIN (INTERNAL) IMPORT
   POSITION = NOREWIND
   NAMES = TAIL
   ITEMS = TABLES
```

   Note that because the **ITEMS** for the two substructures are different, two **SOFIN** commands are required. Naturally, the second of these specifies **POSITION=NOREWIND**.

Substructure Command SOFOUT

Copies all or selected portions of an SOF Database to an export file suitable for importing to another computer of the same or a different type.

Command Syntax:

```
SOFOUT [ ( { INTERNAL } ) ] logical_name
       [   { EXTERNAL } ]
```


Subcommands:

```
POSITION = { REWIND    }
           { NOREWIND  }
           { EOF       }

ITEMS    = { ALL       }
           { MATRICES  }
           { PHASE3    }
           { TABLES    }
           { item_name }

NAMES    = { sub_name  }
           { WHOLESOF  }
```

Option	*Meaning*	*Data Range*
{ INTERNAL } { **EXTERNAL** }	Specifies whether the file will be used to import data to a different, **EXTERNAL**, computer type or the same, **INTERNAL**, computer type.	Keyword
logical_name	Logical name of the export file. [1]	Character
{ REWIND } { **NOREWIND** } { EOF }	Specifies the initial file position. You may **REWIND** the file prior to writing; you may begin output at the current position with **NOREWIND**; or you may position the file to the point immediately preceeding the end-of-file mark, **EOF**. [2]	Keyword
{ **ALL** } { MATRICES } { PHASE3 } { TABLES } { *item_name* }	Identifies the data items which are to be copied to the file for the substructures, *sub_name*, specified in all **NAMES** subcommansd. You may select **ALL** items, all **MATRICES**, all **TABLES**, the **PHASE3** items UVEC, QVEC and SOLN, or individual *item_name*s. [3]	Keyword or Character
{ *sub_name* } { WHOLESOF }	Identifies a substructure for which data will be written. You may select the entire SOF Database, **WHOLESOF**, or a specific substructure name, *sub_name*. [3]	Keyword or Name

Remarks:

1. You must use the **ASSIGN** Executive Control command to define the *logical_name* with **USE=SOFOUT**, and **TYPE=BINARY** for **INTERNAL** files or **TYPE=FORMATTED** for **EXTERNAL** files.

2. When you are creating a new export file, you **must** use **POSITION=REWIND**. If you select the **EXTERNAL** file format, then the subcommands **POSITION=NOREWIND** and **POSITION=EOF** both result in the positioning of the file to the end-of-file.

3. You may include as many as five **NAMES** subcommands for each **SOFOUT** command, where each references a different substructure name. The **ITEMS** that you have selected apply to **all** of these substructures. If you wish to write different **ITEMS** for different substructures, then you must use different **SOFOUT** commands.

4. The files that you export with **SOFOUT** may be imported using the **SOFIN** Substructure Control command.

Examples:

1. Create an export file suitable for importing on a different computer type for your entire SOF Database:

```
ASSIGN SOF1=MYSOF.DAT,OLD,USE=SOF
ASSIGN EXPORT=EXSOF.DAT,NEW,REALLOC,USE=SOFOUT,TYPE=FORMATTED
...
SOFOUT EXPORT
   POSITION = REWIND
```

   Note that for exporting in the **EXTERNAL** format you must **ASSIGN** the logical file with the **TYPE=FORMATTED** parameter.

2. Create an export file containing all of the SOF Database for substructures **WING** and **TAIL** which will be imported on the same computer type:

```
ASSIGN SOF1=MYSOF.DAT,OLD,USE=SOF
ASSIGN EXPORT=EXSOF.DAT,NEW,REALLOC,USE=SOFOUT,TYPE=BINARY
...
SOFOUT (INTERNAL) EXPORT
   POSITION = REWIND
   NAMES = WING
   NAMES = TAIL
```

   Because the same items are being exported for both substructures, only one **SOFOUT** command is required. Also notice that for exporting in the **INTERNAL** format you must **ASSIGN** the logical file with the **TYPE=BINARY** parameter.

3. Create an export file, in internal format, which contains the solution items for **WING** and the table items for **TAIL**:

```
ASSIGN SOF1=MYSOF.DAT,OLD
ASSIGN EXPORT=EXSOF.DAT,NEW,REALLOC,USE=SOFOUT
...
SOFOUT (INTERNAL) EXPORT
   POSITION = REWIND
   NAMES = WING
   ITEMS = SOLN
SOFOUT (INTERNAL) EXPORT
   POSITION = NOREWIND
   NAMES = TAIL
   ITEMS = TABLES
```

   Note that because the **ITEMS** for the two substructures are different, two **SOFOUT** commands are required. The second of these specifies **POSITION=NOREWIND** so that the data will be appended to the export file.

Substructure Command SOFPRINT

Prints a table of contents or selected items from the SOF Database.

Command Syntax:

```
SOFPRINT { name,item1 [,item2,item3,item4,item5 ] }
         { TOC                                    }
```

Subcommands: None

Option	*Meaning*	*Data Range*
name	Name of the substructure for which data are to be printed.	Name
itemi	SOF Database item names. [1]	Character
TOC	Requests the table of contents only.	Keyword

Remarks:

1. You may select subscripted item names by specifying:

   item[*sub_val*]

   where *sub_val* is the selected subscript.

2. On the page heading for the table of contents, the labels are defined as follows:

COLUMN	MEANING	CREATED BY
SS	Secondary substructure number (successor).	EQUIV
PS	Primary substructure number (predecessor).	EQUIV
LL	Lower level substructure number.	CREDUCE, MREDUCE, REDUCE
CS	Combined substructure number.	COMBINE
HL	Higher level substructure number.	CREDUCE, MREDUCE, REDUCE
TYPE	Substructure type: **B** Basic Substructure **C** Combined substructure **R** Guyan reduced substructure **M** Real modal reduced substructure **CM** Complex modal reduced substructure Any of the above **TYPE**s will have a prefix **I** if it is an image substructure.	BASIC COMBINE CREDUCE EQUIV MREDUCE REDUCE

Substructure Command SOLVE

Requests that an analytical solution be performed for a specified substructure.

Command Syntax:

```
SOLVE  name
```

Subcommands:

```
OVERWRITE
MODEL = grid_type
```


Option	*Meaning*	*Data Range*
name	Specifies the name of the substructure to be analyzed using the analytical discipline selected with the **SOL** Executive Control command.	Name
OVERWRITE	Specifies that previously existing solution data will be deleted before this solution is performed. [1]	Keyword
grid_type	Specifies the type of GRID point identification numbers used in the solution model. [2]	Character

Remarks:

1. Prior to solution, the entire substructure tree is analyzed to make certain that no previous solution data exists. If you wish to re-solve your model, then you can either manually **DELETE** all of the appropriate solution items, or you can accomplish the same result by simply using the **OVERWRITE** subcommand.

2. When you perform the **SOLVE** operation, your model may reference either external or internal GRID point identification numbers. (See Chapter 19 of the User's Guide for a complete discussion of this.) The type of identifiers used is important if you are adding any element or load data during the solution. The **MODEL** subcommand allows you to specify the type of identifier using the following options:

grid_type	DESCRIPTION
GRID	Use external identification numbers if possible, otherwise use internal numbers. (Default)
EXTGRID	Use external identification numbers, if not possible, issue an error message.
UEXTGRID	Use external identification numbers and do not allow an internal point to have several external identification numbers. If so, issue an error message.
INTGRID	Use internal identification numbers.
SCALAR	Use scalar points for compatibility with **UAI/*NASTRAN*** Versions 11.7 and earlier.

3. The **UAI/*NASTRAN*** Rigid Formats that support substructuring are 1, 2, 3, 8, 9, 11 and 12.

4. A procedure is available to perform nonlinear material analyses with substructuring models. This procedure is documented in the **UAI/*NASTRAN*** User's Guide.

5. The **SOLVE** command should always be followed by a **RECOVER** commnad to insure that the solution data are saved on the SOF Database.

6. The **SOLVE** command may only be used in PHASE 2 executions.

Substructure Command SUBSTRUCTURE

Initiates the Substructure Control data packet and defines the PHASE of the analysis.

Command Syntax:

SUBSTRUCTURE { PHASE1 / PHASE2 / PHASE3 }

Subcommands: None

Option	*Meaning*	*Data Range*
PHASE1	Indicates that you are performing a PHASE 1 execution in which you create a new Basic Substructure and, optionally, perform any selected operations on substructures that have already been placed on your SOF Database. [1]	Keyword
PHASE2	Indicates that you are performing a PHASE 2 execution in which you request any number of substructure synthesis operations on substructures that have already been placed on your SOF Database, and solve a requested analytical discipline. [1]	Keyword
PHASE3	Indicates that you are performing a PHASE 3 execution to recover analytical results at the Basic Substructure level. [1]	Keyword

Remarks:

1. The table below summarizes the substructuring commands that may be used in each PHASE:

COMMAND	PHASE1	PHASE2	PHASE3
BASIC	■		
CHECK	□	■	■
COMBINE	□	■	
CREDUCE	□	■	
DELETE	□	■	■
DESTROY	□	■	■
EDIT	□	■	■
EQUIV	□	■	
MASSBAL	□	■	■
MRECOVER	□	■	

COMMAND	PHASE1	PHASE2	PHASE3
MREDUCE	□	■	
OPTIONS	□	■	■
PLOT	□	■	■
RECOVER		■	■
REDUCE	□	■	
SOFIN	□	■	■
SOFOUT	□	■	■
SOFPRINT	□	■	■
SOLVE		■	
SUBSTRUC	■	■	■

■ May be used. □ May be used after **BASIC**

2. This command must be the first command in the Substructure Control packet.

This page is intentionally blank.

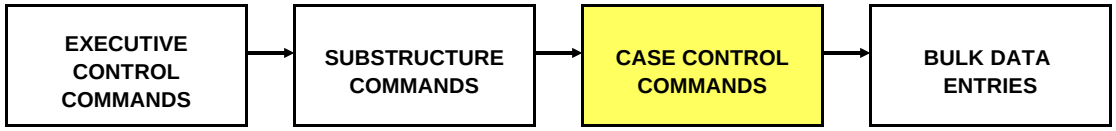
Chapter 4

CASE CONTROL COMMANDS

Case Control commands are used to select specific items from the Bulk Data packet, as described in Chapter 7, to define unique CASES (or SUBCASES) for which solutions will be obtained, and to select the solution results to be printed (written to a file in a printable format) or written to a file in binary or some other format. Note that in **UAI/*NASTRAN*** the command **CASE**, not found in older versions of NASTRAN, has been introduced. For compatibility, the older **SUBCASE** command may be used interchangeably with the newer **CASE** command.

The Case Control command packet is the second in the input data stream unless the substructuring feature is being used, in which case it is the third packet. This is shown in Figure 4-1.

Figure 4-1. CASE CONTROL PACKET LOCATION



This packet immediately follows the Executive Control command **CEND**, or, if substructuring is selected, it immediately follows the Substructuring Control command **ENDSUBS**. The Case Control command packet ends when the command **BEGIN BULK** is encountered. **UAI/*NASTRAN*** provides both ***Structural*** plotting and ***X-Y*** plotting capabilities. Although the plotting commands are actually part of the Case Control command packet, they are described separately in Chapters 5 and 6. Plotting commands must always follow all other Case Control commands.

Note that **UAI/*NASTRAN*** creates an output file, called the ***Print File***, which is formatted and suitable for routing to a standard 132 character wide line printer. At most installations, this file is not automatically routed to the printer. You may review the contents of this file using your standard text editor, and then dispose of the file as you wish. Thus, the generation of output for this file is often referred to as a ***Print Operation***. Also, an alternate form of output is available which once requested the generation of physical punched computer cards. Today, however, this output is written to a file that you have assigned with the option **USE=PUNCH**. This file is always formatted in 80 byte (character) records, and it may easily be examined, or modified, with a text editor.

4.1 CASE AND SUBCASE DEFINITION

There are six commands and three subcommands that you may use to define CASEs (or the older SUBCASEs, which have been retained for compatibility) for solution or output requests in **UAI*/NASTRAN***. SUBCASEs are used to define combinations of different load conditions, boundary conditions and output requests, while CASEs are used to select different analysis disciplines. The definition of these, and their related commands, are defined in Table 4-1

Table 4-1. COMMANDS FOR CASE AND SUBCASE DEFINITION

COMMAND	DESCRIPTION
CASE (or SUBCASE)	Defines the start of a CASE when using the **SOL MULTI** solution sequence for Multidisciplinary Analysis and Design, Aerodynamics and Flutter Analysis, Hydroelastic and Acoustic analysis, or automatic static or modal reduction.
USING	References a previous CASE to obtain a set of modes for use in the current CASE. Available only in **SOL MULTI**. Used for Modal Frequency or Transient Response, Hydroelastic and Acoustic analysis, and Flutter Analysis.
REPCASE	Defines an output request SUBCASE.
SUBCASE	Defines the start of a SUBCASE.
SUBCOM	Defines a linear combination of SUBCASES.
SUBSEQ	Defines the coefficients for a SUBCOM combination
SYM	Defines the start of a symmetry SUBCASE.
SYMCOM	Defines a symmetry combination SUBCASE.
SYMSEQ	Defines the coefficients for a SYMCOM combination.

4.1.1 Cases in the MULTI Solution Sequence

The solution sequence **SOL MULTI** greatly expands the manner in which CASEs are used in **UAI*/NASTRAN***. **SOL MULTI** allows the following uses of CASEs beyond those of the standard Rigid Formats. The specialized features are briefly enumerated:

- ❒ Multiple analysis CASEs are allowed with the ability to change analysis discipline, i.e. statics, modes, dynamic response, etc., between CASEs.
- ❒ Design Optimization of the structural model may be performed, simultaneously using the information from multiple, and different discipline based, CASEs, as mentioned above.
- ❒ Structural modes, created in a **CASE MODES**, may be utilized in a subsequent dynamic response, fluid-structure interaction, or flutter CASE, by referencing them with a **USING** Command.
- ❒ Reduced models, represented by mass and stiffness matrices, may be created using either automatic static condensation techniques (using **CASE REDUCE**) or modal reduction techniques (using **CASE MREDUCE**). The reduced model may be exported using **DMIG** Bulk Data entries or by saving the matrices on an ARCHIVE database.

- An aerodynamic database may be created and stored for later use with the **CASE AERO** command. The **USING** command is used during a **CASE FLUTTER** to refer to these aerodynamic data.
- A combination of three specialized CASEs, **CASE FLUID MODES**, **CASE STRUCTURAL MODES** and **CASE COUPLED MODES**, may be used sequentially to utilize a modal synthesis approach to create a generalized modal model of a fluid-structure interaction system. Such a modal model may the be used in a later dynamic response CASE by application of the **USING** Command.

Detailed explanations for using these specialized features of the **SOL MULTI** Case Control commands are presented in the appropriate chapters in the ***User's Guide***.

4.2 DATA SELECTION

Each CASE defined in the Case Control command packet may be used to select different loading conditions, constraint sets or solution control parameters. The commands used for these purposes are summarized in this section.

4.2.1 Load Selection

The commands shown in Table 4-2 are used to select static and dynamic loads.

4.2.2 Temperature Field Selection

Temperature fields must be defined when performing thermal loads/structural analyses or heat transfer analyses. They are also required when using temperature dependent material properties. The commands used are shown in Table 4-3.

4.2.3 Constraints and Partitioning

The selection of boundary conditions and dependency relations are also made with Case Control commands. These, along with automatic constraint and reduction features are summarized in Table 4-4.

4.2.4 Dynamics Control and Matrix Selection

Table 4-5 summarizes the Case Control commands used for dynamic analyses.

4.2.5 Multidisciplinary Design Optimization Control

Table 4-6 summarizes the commands used for Multidisciplinary Design Optimization (MDO) Analyses.

4.2.6 Nonlinear Analysis Control

There are several Case Control commands used only in nonlinear structural analysis, both material and geometric. Some of these commands are listed under other categories; others are unique to nonlinear analysis. All commands that are specific to nonlinear analysis are summarized in Table 4-7.

4.2.7 Aerodynamic Analysis Control

Table 4-8 summarizes the commands used for Aerodynamic and Flutter (Aeroelastic) Analyses. They have been organized in a manner that shows which commands are used in which discipline.

4.2.8 Fluid-Structure Interaction with Modal Synthesis

Table 4-9 summarizes the commands used for performing Fluid-Structure Interaction Analyses with Modal Synthesis. They have been organized in a manner that shows which commands are used in which discipline.

Table 4-2. COMMANDS FOR LOAD SELECTION

COMMAND	DESCRIPTION
DEFORM	Selects preloading due to element deformation.
DLOAD	Selects combined dynamic loading.
DSCO	Specifies load increments for Differential Stiffness analysis.
LOAD	Selects static loading conditions.
LOADCYH	Selects harmonic loads for Cyclic Symmetry.
LOADCYN	Selects physical loads for Cyclic Symmetry.
NONLINEAR	Selects Nonlinear Transient analysis loading.

Table 4-3. COMMANDS FOR TEMPERATURE FIELD SELECTION

COMMAND	DESCRIPTION
TEMP	Selects temperatures for loads and materials.
TEMP(ESTI)	Selects estimated temperatures for Nonlinear Static Heat Transfer analyses.
TEMP(LOAD)	Selects temperature set for loads.
TEMP(MAT)	Selects temperature set for material properties.

Table 4-4. COMMANDS FOR CONSTRAINT SELECTION

COMMAND	DESCRIPTION
AUTOOMIT	Selects the automatic reduction option.
AUTOREDUCE	Selects automatic static reduction.
AUTOSING	Selects singular matrix decomposition options.
AUTOSPC	Selects automatic matrix singularity removal.
AXISYM	Selects boundary conditions for Axisymmetric Harmonic analyses.
DSYM	Selects symmetry planes for Cyclic Symmetry analyses.
MPC	Selects the multipoint constraint set.
NLREDUCE	Selects automatic nonlinear partitioning.
SPC	Selects the single-point constraint set.

Table 4-5. COMMANDS FOR DYNAMICS CONTROL

COMMAND	DESCRIPTION
B2GG, K2GG M2GG	Selects g-set direct input damping, stiffness, and mass matrices.
B2PP, K2PP M2PP	Selects p-set direct input damping, stiffness, and mass matrices.
CMETHOD	Selects a complex eigenextraction method.
DYNRED	Selects dynamic reduction control parameters.
FREQUENCY	Selects a set of frequencies for Frequency Response analyses.
IC	Selects initial conditions for Transient Response analyses.
METHOD	Selects a real eigenextraction method.
SDAMPING	Selects modal damping values.
TFL	Selects dynamic transfer functions.
TSTEP	Selects integration time steps for Transient analyses.

Table 4-6. COMMANDS FOR MDO

COMMAND	DESCRIPTION
DESCON	Selects design constraints for Sensitivity analyses and MDO.
MAXRETAIN	Specifies the maximum number of retained design constraints.
OBJECTIVE	Selects the objective function for design.

Table 4-7. COMMANDS FOR NONLINEAR MATERIAL ANALYSIS

COMMAND	DESCRIPTION
AUTOREDUCE	Selects automatic nonlinear partitioning.
NLFORCE	Requests nonlinear element internal forces.
NLPRINT	Selects intermediate print of solution convergence information.
NLRESTART	Controls a nonlinear restart.
NLSAVE	Defines nonlinear load increment output.
NLSOLVE	Selects nonlinear solution control data.
NLSTRAIN	Requests strains in nonlinear elements.
NLSTRESS	Requests stresses in nonlinear elements.
NLTYPE	Selects the type of nonlinear analysis.
STEP	Defines the start of a nonlinear load STEP.

Table 4-8. COMMANDS FOR AERODYNAMIC ANALYSIS CONTROL

COMMAND	DESCRIPTION
CASE AERO	Selects the Aerodynamic Analysis discipline.
ARCHIVE	Requests that solution results be written to the ARCHIVE database for future use.
KFREQ	Specifies a list of Reduced Frequencies numbers for which the solution will be computed.
MACH	Specifies a list of Mach numbers for which the solution will be computed.
PRESSURE	Specifies the a list of pressures to be computed.
SYMMETRY	Selects a symmetry condition.
CASE FLUTTER	Selects the Unsteady Aeroelastic (Flutter) Analysis discipline.
DENS	Selects data defining the density-altitude relation.
FLPRINT	Selects Flutter Analysis print options.
FLSOLVE	Selects aerodynamic solution control data.
HSELECT	Specifies a list of Mode numbers to be omitted from generalized coordinates.
KLIST	Defines a list of "soft point" reduced frequencies.
KSELECT	Specifies a list of Reduced Frequencies numbers for which the solution will be computed.
MSELECT	Specifies a list of Mach numbers for which the solution will be computed.
NROOT	Specifies the number of flutter roots to extract.
USING MODES	Selects modes computed in a previous Normal Modes case for use as generalized coordinates.
USING AERO	Selects previously computed Aerodynamic solution results.
VLIST	Defines a set of velocities.

Table 4-9. COMMANDS FOR FSI - MODAL SYNTHESIS

COMMAND	DESCRIPTION
CASE FLUID MODES	Selects the fluid mode computation discipline.
CASE STRUCTURE	Selects the structural mode computation discipline.
CASE COUPLED MODES	Selects the coupled fluid-structure mode computation discipline.
USING FLUID MODES	Selects modes computed in a previous Normal Modes case for the fulid. These are used as generalized coordinates.
USING STRUCTURAL MODES	Selects modes computed in a previous Normal Modes case for the structure. These are used as generalized coordinates.

4.3 OUTPUT SELECTION

A wide variety of output is available from **UAI/*NASTRAN***. Solution results can be printed and/or saved on external files for any set of GRID points or elements that are selected. The following sections describe the available options.

4.3.1 Output Control and Titling

The general commands given in Table 4-10 are available for controlling output.

Table 4-10. COMMANDS FOR GENERAL OUTPUT SELECTION

COMMAND	DESCRIPTION
ECHOBULK	Specifies echo options for the Bulk Data packet.
LABEL	Information appearing on the third titling line.
NLPRINT	Selects intermediate print for nonlinear analyses.
NLSAVE	Defines nonlinear load increment output.
OUTPUT(PLOT)	Start of structural plotter command subpacket.
OUTPUT(XYPLOT)	Start of X-Y plotter command subpacket.
PLOTID	Titling information for plots.
SUBTITLE	Information appearing on the second titling line.
TITLE	Information appearing on the first titling line.

4.3.2 Defining Output Sets

Output may be requested for various sets of GRID points or elements. In transient and frequency response analyses, output may be requested at various time or frequency values. The commands which control the definition of sets are shown in Table 4-10.

Table 4-11. COMMANDS FOR SET DEFINITION

COMMAND	DESCRIPTION
OFREQ	Defines list of frequencies for which output will be generated.
OMODES	Defines list of mode numbers for which output will be generated.
OTIME	Defines list of times for which output will be generated.
SET	Defines a list of identifiers referenced by other input data.

4.3.3 Solution Results

The final solution results depend upon the type of analysis that is performed. The specific output available for a given analysis is described in the User's Guide. Output may be requested for quantities given at GRID points or individual finite elements. When you perform Axisymmetric Harmonic analysis, output may be selected for specific harmonics. Similarly, when using the Cyclic Symmetry capabilities, output may be selected by model segment. Substructuring provides output selection for individual substructures. When using the Design Sensitivity feature, you may obtain output for the design constraints and the sensitivities of each constraint to each design variable. When performing Multidisciplinary Design Optimization, the design iteration history is given automatically during execution.

The **ARCHIVE** command may be used to save the finite element geometry data, the solution results, or both, on an ARCHIVE database. This database may then be queried with ***eShell*** to perform data correlation and report generation. Additionally, the **ARCHIVE** command is an integral part of the Flutter capability. You use the database to store the results of the Aerodynamic solution for subsequent use in Flutter Analysis.

The available output quantities, and the commands that you use to request these data, are presented in Table 4-12.

4.3.4 Exporting Data

You may often wish to export certain **UAI/*NASTRAN*** data and solution results. The commands used for this are summarized in Table 4-12.

Table 4-12. COMMANDS FOR DATA EXPORT

COMMAND	DESCRIPTION
ARCHIVE	Requests that solution results and geometry data be saved on the ARCHIVE database. These data may then be used by ***eShell*** and ***DataMaster***.
EXPORT REDUCED MODEL	Requests that a reduced model be exported as **DMIG** Bulk Data entries.
POST	Controls interfaces to commercial post-processing programs including MSC/PATRAN, I-DEAS, FEMAP and ***RenderMaster***.

Table 4-13. COMMANDS FOR SOLUTION RESULTS

COMMAND	DESCRIPTION
ACCELERATION	Requests accelerations in the physical set.
BMFORCE	Requests that equivalent beam forces (moments, shears, axial loads, and torques) be computed for pseudo-elements, composed of collections of solid elements.
DISPLACEMENT	Requests displacements in the physical set.
EKE	Requests element kinetic energy
ELEMSUM	Prints summary of element data.
EQUILIBRIUM	Requests static equilibrium summary.
ESE	Requests element strain energies.
FLUX	Requests element thermal flux and gradients.
FORCE	Requests element forces.
GPFORCE	Requests GRID point forces.
GPKE	Requests GRID point kinetic energy
GPSTRAIN	Requests GRID point strains.
GPSTRESS	Requests GRID point stresses.
HARMONICS	Requests harmonic output for Axisymmetric analysis or gives the number of harmonics for Cyclic Symmetry analysis.
HOUTPUT	Requests output for Cyclic Symmetry in harmonic form.
MESHERR	Requests mesh error estimates.
MPCFORCE	Request multipoint constraint forces.
NLFORCE	Requests forces in nonlinear elements.
NLLOAD	Requests summary of nonlinear loads.
NLSTRAIN	Requests strains in nonlinear elements.
NLSTRESS	Requests stresses in nonlinear elements.
NOUTPUT	Requests output for Cyclic Symmetry in the physical set.
OLOAD	Requests echo of GRID point applied loads.
PRESSURE	Requests hydrostatic or acoustic pressure at GRID points.
SACCELERATION	Requests solution set accelerations.
SDISPLACEMENT	Requests solution set displacements.
SENSITIVITY	Requests design constraint sensitivity results.
SPCFORCE	Requests single-point constraint forces.
STRAIN	Requests element strains.
STRESS	Requests element stresses.
SVELOCITY	Requests solution set velocities.
THERMAL	Requests output temperatures.
VELOCITY	Requests velocities in the physical set.

4.4 DEFINING ANALYSIS CASES

As indicated earlier, Case Control commands are used for a variety of purposes including: definition of loading conditions; definition of boundary conditions; and output selection. There are minor differences in Case Control between the DMAP-based solution algorithms and the **MULTI** Solution Sequence. These are described briefly in the following sections.

4.4.1 CASE (or SUBCASE) Specifications

In general, when you use DMAP-based Rigid Formats, a separate CASE must be defined for each unique combination of loading conditions and constraint selections. CASEs may also be defined to request different solution results from the same analysis, but you need not specify the type of analysis, e.g. modes, since the Rigid Format only applies to a single analysis discipline or to a normal modes extraction followed by CASEs of a modal analysis type which uses those modes as generalized degrees of freedom, i.e. Modal Frequency Response analysis.

Complex eigenvalue analysis and frequency response also require separate CASEs for each unique set of direct input matrices. Case Control commands are structured so that a minimum amount of repetition is necessary. All commands placed before the first CASE will be in effect unless overridden within an individual CASE. Each Rigid Format has differing Case Control requirements. These, along with examples, are described in the ***User's Guide***.

For **SOL MULTI** CASEs, similar rules apply. Each of the disciplines may have its own Case Control requests. Where meaningful, for example, boundary condition commands such as **SPC** and **MPC**, may be placed above the CASE level. When this is done, the commands will be applied to all of the CASEs. The major difference between the Case Control for **MULTI** and for DMAP-based solutions is the fact that **MULTI** allows any combination of Statics, Normal Modes, Transient and Frequency Response analyses, Aerodynamic analyses, and Flutter analyses to be performed in a single program execution; therefore, you must specify the analysis discipline as part of the **CASE** command.

There are several Case Control commands which may only be used with **MULTI**, including MDO, as shown in Table 4.6, Aerodynamic analysis, as shown in Table 4.8, and Fluid-Structure Interaction (FSI) analysis as shown in Table 4.9.

A major feature of **MULTI** is the ability to use the modes generated in a **CASE MODES** and the solution vectors generated in a **CASE STATICS** as generalized coordinates for a subsequent Modal Frequency Response analysis. Similarly, a Flutter analysis may use modes from a previous **CASE MODES** and Aerodynamic results from a previous **CASE AERO**. See the **CASE** and **USING** commands for additional details.

4.5 MINIMAL REQUIRED CASE CONTROL COMMANDS

Each analysis discipline in **UAI/*NASTRAN*** requires certain minimal data to execute properly. The following table provides you with the Case Control commands needed to perform the specified analytical discipline, either using DMAP Rigid Formats, or the multidisciplinary solution sequence, **MULTI**.

DISCIPLINE	REQUIRED DATA	
	DMAP Rigid Formats	SOL MULTI
STATICS	LOAD[1]	LOAD[1]
INERTIA RELIEF	LOAD[1]	LOAD[1]
NORMAL MODES	METHOD	METHOD
DIFFERENTIAL STIFFNESS	LOAD[1]	
BUCKLING	LOAD, METHOD	
DIRECT COMPLEX EIGENVALUE	CMETHOD	
DIRECT FREQUENCY RESPONSE	DLOAD, FREQ	DLOAD, FREQ
DIRECT TRANSIENT RESPONSE	DLOAD, TSTEP	DLOAD, TSTEP
MODAL COMPLEX EIGENVALUES	CMETHOD, METHOD	
MODAL FREQUENCY RESPONSE	METHOD, DLOAD, FREQ	USING MODES, DLOAD, FREQ
MODAL TRANSIENT RESPONSE	METHOD, DLOAD, TIME	USING MODES, DLOAD, TIME
DIFFERENTIAL STIFFNESS MODES	LOAD[1], DSCO, METHOD	
CYCLIC STATICS	HARMONICS, LOAD[1]	
CYCLIC MODES	HARMONICS, METHOD	
CYCLIC BUCKLING	HARMONICS, METHOD, LOAD[1]	
MATERIAL NONLINEAR	LOAD[1]	
GEOMETRIC NONLINEAR	LOAD[1], NLTYPE	
MIXED NONLINEAR	LOAD[1], NLTYPE	
SHOCK	SHOCK, SPC	
SENSITIVITY STATICS	LOAD[1], DESCON	
SENSITIVITY MODES	METHOD, DESCON	
MDO		OBJECTIVE
AERODYNAMICS		ARCHIVE AERO, SYMMETRY MACH, KFREQ
FLUTTER		USING AERO MODEL, USING MODES FMETHOD, DENS, SYMMETRY
MREDUCE		BOUNDARY, NETHOD
REDUCE		USING MODES, BOUNDARY, METHOD AUTOREDUCE
FLUID MODES[2]		METHOD
STRUCTURAL MODES[2]		METHOD
COUPLED MODES[2]		METHOD, USING FLUID MODES, USING STRUCTURAL MODES
1. LOAD indicates either the LOAD command or other equivalents:TEMP(LOAD), SPC or DEFORM		
2. These disciplines are used in performing fluid-structure interaction analyses using modal synthesis		

4.6 COMMONLY USED OPTIONS

There are a number of commonly used terms the appear in the Case Control command definitions. This section defines these so it is not necessary to repeat them throughout subsequent command descriptions.

4.6.1 SORT1 and SORT2

Most of the **UAI/*NASTRAN*** solution results may be printed in either of two collating orders, or sorts. These are called **SORT1** and **SORT2**. All output for your **UAI/*NASTRAN*** execution must be in one of these sort orders. You select the sort with the command:

$$\textbf{SORT} = \left\{ \begin{array}{l} \textbf{SORT1} \\ \text{SORT2} \end{array} \right\}$$

Note that **SORT1** is used if you do not select a sort option except in transient analysis where the only sort order available is **SORT2**. Table 4-14 illustrates the manner in which results are presented for the various Rigid Formats and **SORTi** options.

When using the standard Rigid Formats, any request for **SORT2** in Case Control will cause all output to be in that sort. When performing Analysis or Design with the **MULTI** Solution Sequence, the sorting of data depends on the discipline. For all disciplines except Statics, the sort request may vary from case to case. In the case of Statics analysis all consecutive cases are treated as a single group for data sorting purposes. As a result, any request for **SORT2** will cause the entire group to be processed in **SORT2**. Also, in **MULTI**, the Normal Modes results may also be in **SORT2**.

4.6.2 PRINT and NOPRINT (POST)

For efficiency, **UAI/*NASTRAN*** only recovers the solution results which you request in your input data stream. This means, for example, that element stresses are not computed unless specifically requested for output using the **STRESS** Case Control command. Each type of result may be controlled separately by a Case Control output request. When such a request specifies the **PRINT** option, then the results are generated and written on the Print File.

In some cases, you may wish to pass the results to another computer program for post-processing and not have them printed in the program execution. For example, if all element stress data were desired for post processing, but not for printing, either of the following commands would be used:

```
STRESS(NOPRINT) = ALL
STRESS(POST) = ALL
```

NOPRINT is a synonym for the **POST** and **PLOT** command options which were formerly available.

Table 4-14. OUTPUT SORT ORDER FOR RIGID FORMATS

RIGID FORMAT	S0RT1	S0RT2
STATICS (1,2)	SUBCASE 1 ALL GRID RESULTS ALL ELEMENT RESULTS SUBCASE 2 ALL GRID RESULTS ALL ELEMENT RESULTS ...	GRID 1 ALL SUBCASE RESULTS GRID 2 ALL SUBCASE RESULTS ... ELEMENT 1 ALL SUBCASE RESULTS ELEMENT 2 ALL SUBCASE RESULTS ...
NORMAL MODES (3)	MODE 1 ALL GRID RESULTS ALL ELEMENT RESULTS MODE 2 ALL GRID RESULTS ALL ELEMENT RESULTS ...	**S0RT2** output is not available.
FREQUENCY RESPONSE (8,11)	FREQUENCY 1 ALL GRID RESULTS ALL ELEMENT RESULTS FREQUENCY 2 ALL GRID RESULTS ALL ELEMENT RESULTS ...	GRID 1 ALL FREQUENCIES GRID 2 ALL FREQUENCIES ... ELEMENT 1 ALL FREQUENCIES ELEMENT 2 ALL FREQUENCIES ...
TRANSIENT RESPONSE (9,12)	**SORT1** output is not available.	GRID 1 ALL TIME POINTS GRID 2 ALL TIME POINTS ... ELEMENT 1 ALL TIME POINTS ELEMENT 2 ALL TIME POINTS ...

In static analysis, you may use the **REPCASE** command, followed by commands of the form:

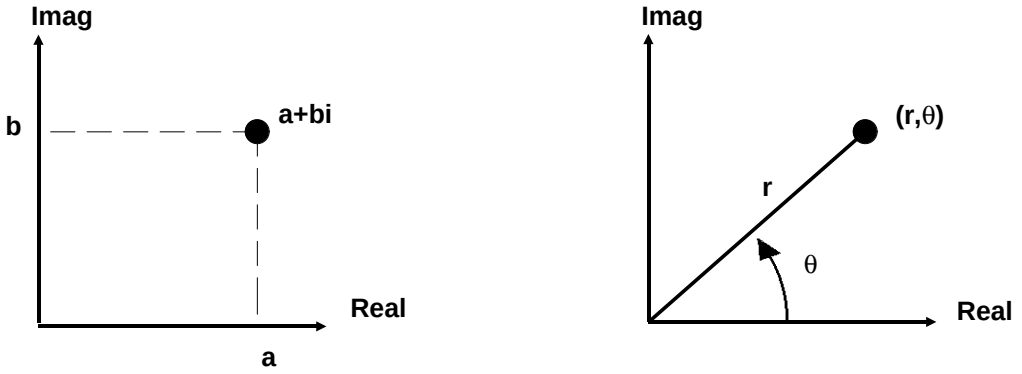
```
SET n = 1 THRU 100
STRESS(PRINT) = n
```

to generate printed output for a subset of data previously computed using the **NOPRINT** option. The use of the **SET** command is described later in this chapter.

4.6.3 RECTANGULAR and POLAR

Complex number results occur when performing frequency response or complex eigenvalue analyses. There are two available formats for recovering complex solution results. The first is called **RECTANGULAR** and the second **POLAR**. Figure 4-2 illustrates the manner in which complex values are represented in each.

Figure 4-2. COMPLEX OUTPUT REPRESENTATIONS



The **RECTANGULAR** option represents the value of a complex number, *P*, using the real and imaginary parts:

$$P = a + bi$$

while the **POLAR** format represents the value as a radius magnitude, *r*, and the angle, θ, as shown in the figure. Formerly, **RECTANGULAR** was called Real/Imaginary and **POLAR** was called Magnitude/Phase. These descriptors may still appear in the tabular output for various solution results.

4.6.4 Output Set Selection

Case Control commands for requesting output generally allow you to select from three options:

```
{ ALL  }
{ sid  }
{ NONE }
```

An output request for **ALL** may produce large amounts of Print. This may be avoided by specifying an *sid* which limits the output to a subset of points or elements. The **NONE** option is used to disable for a selected CASE an output request which you selected above the CASE level.

4.6.5 Configuration Parameters

As indicated in Chapter 1, **UAI/*NASTRAN*** is configured to perform optimally on a given computer. This is done by using a Configuration. Some of the items in the **[UAI/NASTRAN]** Component of your Configuration control parameters are found in Case Control commands. When this is the case, the item is noted for the specific command.

4.7 COMPATIBILITY WITH OTHER SYSTEMS

Compatibility with other NASTRAN variants is maintained in many ways. This section describes some of these compatibility features.

4.7.1 The AUTOSPC Feature

UAI/*NASTRAN* provides a Case Control command, **AUTOSPC**, to automatically remove singularities from your model. This feature may also be selected and controlled with the Bulk Data entry:

```
PARAM,AUTOSPC,YES
```

If this form is used instead of the Case Control command, then the **AUTOSPC** function options are **SPC** and **NOPRINT**.

Print control of the **AUTOSPC** function is also available using the Bulk Data entry:

```
PARAM,PRGPST,YES
or
PARAM,PRGPST,NO
```

which are equivalent to the **AUTOSPC** Case Control command options **PRINT** and **NOPRINT**.

4.7.2 The AUTOOMIT Feature

The **AUTOOMIT** function may also be invoked using the Bulk Data entry:

```
PARAM,ASING,-1
```

If this form is used instead of the Case Control command, then the **AUTOOMIT** function options are **PRINT** and **MASS=0.0**. Furthermore, the **AUTOOMIT** function with these options is automatically activated whenever a Givens or Modified Givens eigenextraction method is requested in the analysis, and the dynamic reduction procedure is not used.

4.8 CASE CONTROL COMMAND DESCRIPTIONS

Case Control commands are entered in free-format. In presenting the general formats for each command and its options, the following conventions are used:

- Many Case Control commands contain ***keywords***. All keywords are shown in capitalized, bold computer type such as: **AUTOSPC**. All such keywords must be entered exactly as they are specified, subject only to the exception that they may be abbreviated by their first four characters.
- Some commands contain ***parentheses***. These must be entered if an option requiring them is selected.
- Lower case italicized computer type, such as *sid*, indicates that you must provide a specific data value.
- Braces { } enclose a list of two or more options from which you may select one.
- Brackets [ ] indicate that the enclosed keywords and parameters may be omitted when you use the command.
- When a choice is to be made from a list, the default choice is presented in boldface type as in the following: { CHOICE1 / CHOICE2 / **DEFAULT** }
- A single command line may not exceed 72 characters. However, you may continue a command by ending the current record with a comma and continuing to the next record. You may not continue any commands which define titling information.

Additionally, command options have a valid data range and, in some cases, a default value. The following table defines the data range specifications found in this chapter.

DATA RANGE SPECIFIER	MEANING
Integer	The data must be an integer number in the range of indicated values.
Real	The data must be a real number in the range of indicated values.
Character	The data must be a string of characters beginning with a letter.
Keyword	The data must be a character string matching the first four or more unique characters of a keyword option.

this page is intentionally blank.

Case Control Command $ [Comment]

Defines the beginning of commentary text.

Command Syntax:

```
$ [Any character data]
```

Examples:

1. Enter a comment line in the Case Control packet:

```
$ THIS LOAD IS A 1-G PULLUP.
LOAD = 101
```

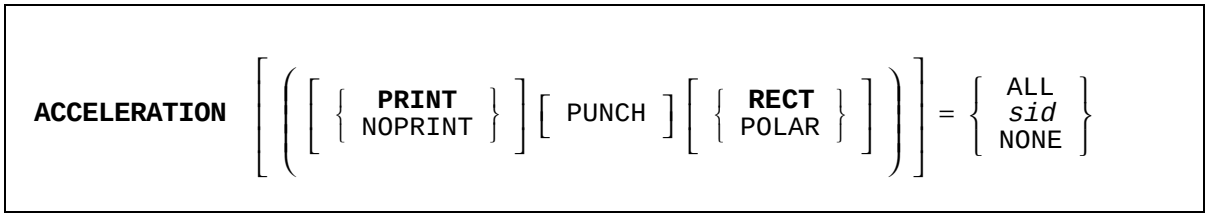
2. Place a comment after a Case Control command:

```
SPC = 10 $ SIMPLE FIXED-FIXED CONFIGURATION
```

Case Control Command ACCELERATION

Requests acceleration vector solution results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **RECT** / POLAR }	Selects output format for Frequency Response problems.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **ACCELERATION** output are only available for Transient and Frequency Response analyses.

Example:

1. Print the accelerations for all points in **SORT2** using **POLAR** format. Also write the results to the **PUNCH** file:

```
SORT = SORT2
...
ACCELERATION(POLAR,PUNCH) = ALL
```

Case Control Command ARCHIVE

Requests creation of an archive database with selected geometry and solution results.

Command Syntax:

ARCHIVE *option_list* [TO *logical_name* [:*path_name*]] [{ **APPEND** / OVERWRITE }]

option_list ⇒ *option_term*,*option_term*,...,*option_term*

option_term ⇒ { **GEOMETRY** / { ELEMSOL / GRIDSOL } or **SOLUTION** / RANDOMSOL / OPTIMIZATION / SENSITIVITY / MODES / AERODYNAMICS }

Option	*Meaning*	*Data Range*
{ **GEOMETRY** / **SOLUTION** / GRIDSOL / ELEMSOL }	Selects the type of analysis data that will be written to the archival database. Selections include: the **GEOMETRY** data; the analytical **SOLUTION** results; or the solution results for GRID points or elements, **GRIDSOL** and **ELEMSOL**, respectively. [1]	Keyword
{ OPTIMIZATION / SENSITIVITY }	Selects the type of Multidisciplinary Design Optimization solution data to be written to the archival database. [2]	Keyword
RANDOMSOL	Requests that Random Response solution data be written to the archival database.	Keyword
MODES	Requests that normal modes matrices be written to the archival database. [3]	Keyword
AERODYNAMICS	Requests that aerodynamic matrices (AIC's) be written to the archival database. [4]	Keyword
logical_name	Selects the logical name of an **ASSIGN**ed file [5].	Character
path_name	Specifies an optional path name giving an ***eBase*** directory when the selected data will be stored [6].	Character
{ **APPEND** / OVERWRITE }	Specifies whether the selected data will be **APPEND**ed to the existing ARCHIVE file, or will **OVERWRITE** existing data. [7]	Keyword

4 CASE

Remarks:

1. If you specify **GEOMETRY**, then all element connection data, GRID point data, element property data, and material property data will be stored in relations on the database. If you specify **SOLUTION**, then the GRID point and element response quantities are placed on the database. The specific quantities depend on the analysis discipline. You may use the **GRIDSOL** and **ELEMSOL** options to limit the responses either to those at GRID points or those of elements.

   The **ARCHIVE** command may appear above the SUBCASE level or within a SUBCASE like other Case Control commands. It behaves in the usual manner except when you select **GEOMETRY**. No matter how many **GEOMETRY** requests are present, the data are only written once to each unique database location, i.e. combination of *logical_name* and *path_name*.

   Note that the default options for **ARCHIVE** are **GEOMETRY** and **SOLUTION**. You may enter the **ARCHIVE** options on a single command, or you may use multiple commands:

   ```
   ARCHIVE GEOMETRY,SENSITIVITY,MODES
          or
   ARCHIVE GEOMETRY
   ARCHIVE SENSITIVITY
   ARCHIVE MODES
   ```

2. The **OPTIMIZATION** and **SENSITIVITY** options are available only when performing Multidisciplinary Design Optimization. When you select **OPTIMIZATION**, then the mathematical design variables for all iterations and the optimization history are written to the ARCHIVE database. If **SENSITIVITY** is specified, then all constraint values and their sensitivities are archived. Only one of each command may appear in your Case Control packet.

3. The **MODES** option may only be used when performing a **CASE MODES** in **MULTI**. When archiving the **MODES**, the eigenvalue and eigenvector data are identified by the **CASE** identification number, *case_id*:

   CASE *case_id* **MODES**

   When these modes are used later to perform analyses by the modal method, then you must reference them by the same **CASE** identification number.

   USING MODES *case_id* **FROM** *logical_name*

   Further, if *case_id* already exists on the archive database, the old results will be overwritten.

4. The **AERODYNAMICS** option may only be used when performing a **CASE AERO** in the **MULTI** solution. To use the aerodynamic loads in a subsequent flutter analysis, the **USING** command is applied:

   ```
   CASE 101 AERO
      USING AERO FROM logical_file
   ```

5. A default logical file, **ARCHIVE**, is provided. You may use the **ASSIGN** Executive Control command to define another logical file with **USE=ARCHIVE**, and you may **ASSIGN** as many ARCHIVE files as you wish. Specific **ARCHIVE** requests may then direct output to any or all of the logical files as shown in Example 2 below.

6. The ***eBase*** *path_name* descriptors have the form:

   logical_name:*/dir1/dir2/.../dirn*

   where *diri* are directory names on the database. If any or all paths in the *path_name* do not exist, then they are created.

7. The **OVERWRITE** option results in the deletion and re-creation of all database entities generated during the current analysis. For example, if an execution creates and saves an ARCHIVE for three SUBCASEs of Static analyses:

```
ARCHIVE
...
SUBCASE 1
   ...
SUBCASE 2
   ...
SUBCASE 3
...
```

Then the resulting **SOLUTION** relations will contain records for three **SUBCASE**s. If this were followed by the subsequent execution:

```
ARCHIVE OVERWRITE
...
SUBCASE 4
...
```

which **ASSIGN**ed the same ARCHIVE file, then the **SOLUTION** relations after this job will only contain the results for **SUBCASE** 4.

Examples:

1. Write all geometry and solution results for each subcase on the logical file **YOURFILE** in a directory named **myeb/model**:

```
...
ASSIGN YOURFILE,NEW,USE=ARCHIVE
...
ARCHIVE GEOMETRY,SOLUTION TO YOURFILE:/myeb/model
...
SUBCASE 1
   ...
SUBCASE 2
   ...
```

2. Write the geometry results to logical file **GEOMFILE** and the solution results from SUBCASEs **101** and **201** to logical files **SOL1FILE** and **SOL2FILE**, respectively:

```
...
ASSIGN GEOMFILE,NEW,USE=ARCHIVE
ASSIGN SOL1FILE,NEW,USE=ARCHIVE
ASSIGN SOL2FILE,NEW,USE=ARCHIVE
...
ARCHIVE GEOMETRY TO GEOMFILE
...
SUBCASE 101
   ARCHIVE SOLUTION TO SOL1FILE
   ...
SUBCASE 201
   ARCHIVE SOLUTION TO SOL2FILE
   ...
```

3. Save the normal modes generated in a job to logical file **SAVEMODE** so they may be used in a subsequent modal Frequency Response analysis:

```
ASSIGN SAVEMODE,NEW,USE=ARCHIVE
...
CASE 100 MODES
   ARCHIVE MODES TO SAVEMODE
   ...
```

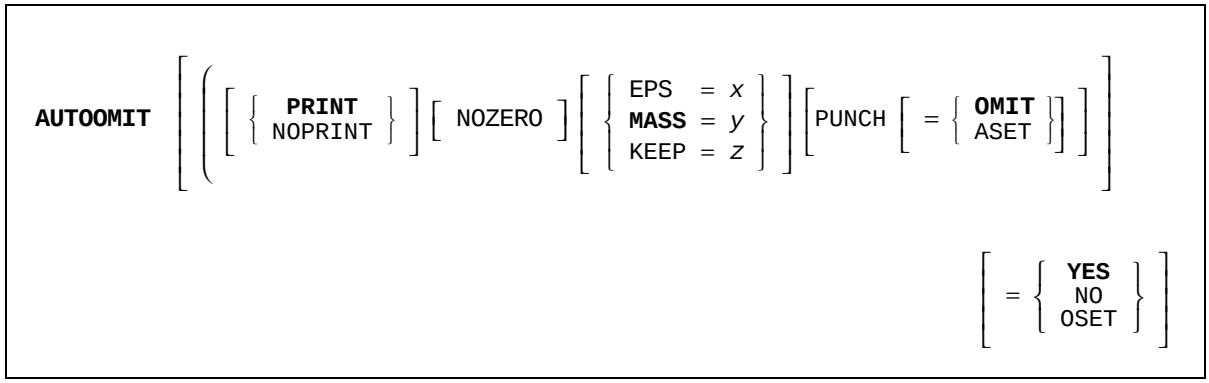
4. Save the results of an Aerodynamic analysis to logical file **MYAERO** for a later Flutter analysis:

```
ASSIGN MYAERO,NEW,USE=ARCHIVE
...
CASE 1 AERO
   ...
   ARCHIVE AERO TO MYAERO
   ...
```

Case Control Command AUTOOMIT

Requests that certain degrees of freedom be OMITted, or reduced, from the solution.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Selects or deselects the printing of the summary table of degrees of freedom that were automatically omitted. [1]	Keyword
NOZERO	Deselects the printing of degrees of freedom whose mass is zero from the summary. [1]	Keyword
x	Specifies the value of mass-to-stiffness ratio below which degrees of freedom will be omitted.	Real≥0.0 Default=0.0
y	Specifies value of mass, in model units, below which degrees of freedom will be omitted. [2]	Real≥0.0 Default=0.0
z	Specifies the decimal percentage of all degrees of freedom, ranked in descending order of mass-to-stiffness ratio, to be retained in the *a-set*. [2]	0.0≤Real≤100.0
PUNCH	Requests that **OMIT** or **ASET** Bulk Data entries be written to the **BULK** file for degrees of freedom that have been omitted, or retained in the *a-set*, respectively. [3]	Keyword
{ **YES** / NO }	Enables or disables the **AUTOOMIT** option. When enabled, the operation is performed on the a-set degrees of freedom.[4]	Keyword
OSET	Specifies that the **AUTOOMIT** feature operates on the o-set degrees of freedom. [5]	Keyword

Remarks:

1. Selecting the **PRINT** option can show a modeling error such as the omission of a lumped mass. However, for large models, the **NOZERO** print option is recommended to avoid a summary of uninteresting eliminations such as null rotational mass components.
2. The use of the **MASS** or **KEEP** option is suggested only for models that have very strong, well-defined mass characteristics. Either option may be used to perform a Guyan Reduction without specifying the *a-set* explicitly, in which case only degrees of freedom with mass will be retained.
3. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=BULK** if you use the **PUNCH** option.
4. The **AUTOOMIT** feature is automatically enabled if a **GIV** or **MGIV** eigenvalue extraction is requested and a dynamic reduction is not requested. In all other cases, the **AUTOOMIT** feature is disabled. The following all have the same meaning and represent the default condition when a **GIV** or **MGIV** eigenvalue extraction procedure is requested:

   ```
   AUTOOMIT
   AUTOOMIT(PRINT,MASS=0.0)
   AUTOOMIT=YES
   ```

   For these cases, all massless degrees of freedom will be assigned to the *o-set* unless otherwise constrained.
5. The **AUTOOMIT** operation can be used even if you have either **ASET** or **OMIT** Bulk Data entries in your input data stream. The results depend on the options selected. For example, if you select the option **KEEP=10.0**, then the results is desribed in the following table:

If the AUTOOMIT Option is:	Then, the number of degrees of freedom (dof) in the final *a-set* is:
YES	The number of dof in the *a-set* resulting from the **ASET** or **OMIT** Bulk Data entries **PLUS** 10% of the number of dof specified in the *a-set*.
OSET	The number of dof in the *a-set* resulting from the **ASET** or **OMIT** Bulk Data entries **PLUS** 10% of the number of dof specified in the *o-set*.

   The **OSET** option is useful when you want to protect specific degrees of freedom during the **AUTTOOMIT** processing, i.e. none of these degrees of freedom will be moved to the *o-set*.
6. The **AUTOOMIT** command must be used above the SUBCASE level.
7. The Case Control command **BOUNDARY** may also be used to define a set of boundary degrees of freedom that will be retained in the a-set during the **AUTOOMIT** processing.

Examples:

1. Perform an **AUTOOMIT** retaining 10 percent of the degrees of freedom having the greatest mass to stiffness ratio and print the results:

   ```
   AUTOOMIT(PRINT,KEEP=10.0)
   ```
2. Perform an **AUTOOMIT** disabling the print of massless degrees of freedom. Also write the results to the **BULK** file:

   ```
   AUTOOMIT(NOZERO,PUNCH) = YES
   ```

Case Control Command AUTOREDUCE

Requests that an automatic static condensation, or Guyan reduction, be performed on a model to produce a new, reduced model. This command is available only in a **CASE REDUCE** when using **SOL MULTI**.

Command Syntax:

```
AUTOREDUCE [ ( [ { PRINT } ] [ { KEEP = x , INCREMENT = Δx } ]
             [ { NOPRINT } ]   { EPS = y , INCREMENT = Δy }
             [ { DETAIL  } ]   { MASS = z , INCREMENT = Δz }

             [ TOLER [ = { tid   } ] ] [ { PERCENT  } ]
                         { toler }     { ABSOLUTE } ]

             [ LFREQ = f1 ] [ HFREQ = f2 ] [ MAXITER = niter ] ) ]
```

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT / DETAIL }	Selects (**PRINT**) or deselects (**NOPRINT**) printing a summary of model frequencies for each iteration. The **DETAIL** option includes a complete table of the degrees of freedom that are omitted from the model. **WARNING**: The **DETAIL** option prints each omitted degree of freedom for each iteration. For large models, this results in very large quantities of output. [1]	Keyword
$x, \Delta x$	Specifies the decimal percentage of all degrees of freedom to be *retained* in the *a-set* during the first iteration, and the increment in the percentage to be used during subsequent iterations. [2]	Real≥0.0 Default=0.0
$y, \Delta y$	Specifies the value of mass-to-stiffness ratio below which degrees of freedom will be omitted during the first iteration, and the increment in the mass-to-stiffness ratio to be used during subsequent iterations. [2]	Real≥0.0 Default=0.0
$z, \Delta z$	Specifies value of mass, in model units, below which degrees of freedom will be omitted during the first iteration, and the increment in the mass to be used during iteration. [2]	0.0≤Real≤100.0
tid	Identification number of a **TABLED1** Bulk Data entry that defines the convergence criteria for the natural frequencies. [3]	Integer>0
toler	Specifies the convergence criteria of all the frequencies of the current iteration with respect to the initial frequencies of the baseline model. [3]	Real≥0.0

New: V20.1

{ **PERCENT** } { ABSOLUTE }	Defines the interpretation of *toler*, or the data in table *tid*, as either a **PERCENT**, or an **ABSOLUTE** numeric value. [4]	Keyword
f1	Specifies the lowest frequency to be used for model convergence tests. [4]	Real≥0.0 Default =–∞
f2	Specifies the highest frequency to be used for model convergence tests. [4]	Real≥0.0 Default =+∞
niter	Specifies the maximum number of iterations to perform.	Integer>0 Default=5

Remarks:

1. The iterative procedure is used only when performing the **AUTOREDUCE** operation in a **CASE REDUCE** discipline.
2. You must select one of the options **KEEP**, **EPS** or **MASS**, and in all cases, you must specify an **INCREMENT** which must be a positive value. For **EPS** and **MASS**, the **INCREMENT** value is used to **REDUCE** the control value for successive iterations. The **CASE** must also include a **BOUNDARY** command, a **METHOD** command, and a **USING** command. Also, the **CASE REDUCE** must have the same **SPC** set as the **CASE** referenced on the **USING** command.
3. Convergence criteria may be frequency dependent by using a **TABLED1** Bulk data entry to define the criteria for a given frequency. For example:

   ```
   TABLED1,100,,,,,,,,+T1
   +T1,0.0,0.01,500.0,0.10
   ```

   defines the convergence criteria to be 1% (i.e. `1.0` if the **PERCENT** option is used, or `0.01` Hz if the **ABSOLUTE** option is used) at `0.0` Hz and increase linearly to `10.0`%, or `0.10` Hz, at `500.0` Hz. Intermediate criteria values are interpolated from this linear relationship.
4. The extracted modal frequencies, as specified by the **METHOD** command, are used for defining the convergence criteria. You may limit the convergence checks on the modes by specifiying a lower bound (**LFREQ**), an upper bound (**HFREQ**), or both.

   If the **SPC** set that you select does not remove all rigid body modes, you will obtain modes with frequencies near `0.0`. In this case, the **PERCENT** option will fail. If you have this situation, then you may change to the **ABSOLUTE** method, or you may specify a lower bound frequency that is greater than `0.0`, i.e. *f1*>`0.0`.
5. **PERCENT** and **ABSOLUTE** are used to determine when successive sets of modal frequencies have been computed with sufficiently small differences so that convergence is achieved, and the current a-set is accepted.
6. You must use the **EXPORT REDUCED MODEL** Case Control command to output the reduced matrices.

New: V20.1

Examples:

1. Perform an **AUTOREDUCE** using static condensation so that the natural frequencies of the reduced model are within 5% of the full model up to 100 Hz. Retain 10% of the model degrees of freedom in the first iteration, and increase the number by 2% in subsequent iterations:

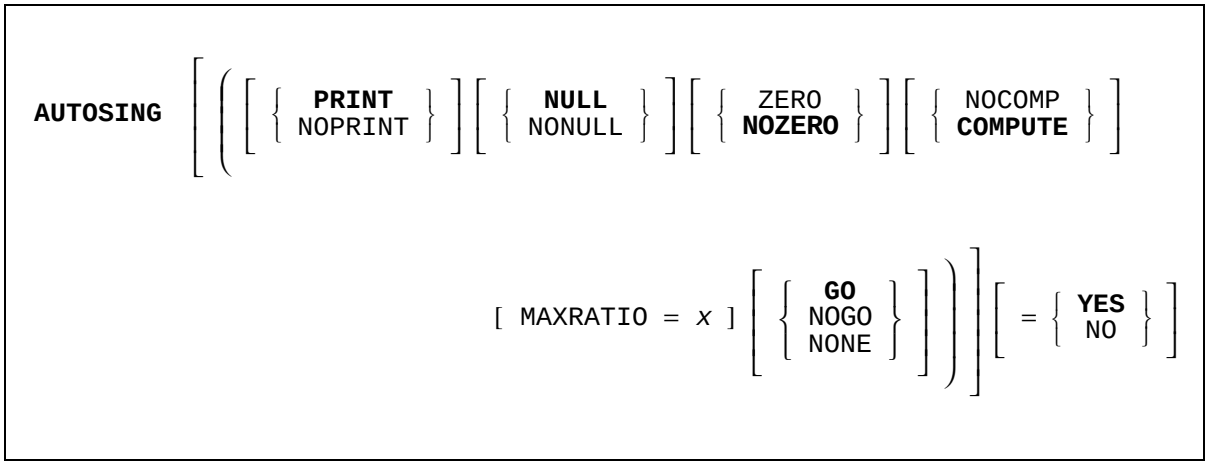
```
CASE 1 MODES
   METHOD = 100
   SPC = 1
   ...
CASE 2 REDUCE
   AUTOREDUCE (KEEP=10.0,INCREMENT=2.0,TOLER=5.0,PERCENT,HFREQ=100.0)
   USING MODES 1
   METHOD = 100
   SPC = 1
   BOUNDARY = 1
   EXPORT REDUCED MODEL ...
```


New: V20.1

Case Control Command AUTOSING

Requests singular matrix decomposition options for the **symmetric** solver.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Enables and disables the printing of a table of detected singularities.	Keyword
{ **NULL** / NONULL }	Enables and disables the placing of the value 1.0 on the diagonal if a **NULL** column is encountered to allow decomposition to continue.	Keyword
{ ZERO / **NOZERO** }	Enables and disables **ZERO** diagonal term checking, if enabled, decomposition continues unless the leading minor is zero. If disabled with **NOZERO**, the decomposition will terminate when a zero diagonal is encountered.	Keyword
{ NOCOMP / **COMPUTE** }	Disables and enables replacing any diagonal term which is **COMPUTE**d as zero with a small number which allows decomposition to continue.	Keyword
x	Specifies a threshold value for the ratio of terms on the diagonal of the stiffness matrix to those corresponding terms on the diagonal of the decomposition factor. All terms whose ratio exceeds *x* are identified.	Real>0.0 Default = 10^7
{ **GO** / NOGO / NONE }	Requests that execution be continued or terminated if any values of **MAXRATIO** exceed *x*. **NONE** is used to deselect this check.	Keyword

Remarks:

1. The **AUTOSING** command must be used above the **SUBCASE** level.
2. **AUTOSING** will be used only with real symmetric matrix decomposition.
3. Extreme caution should be exercised when using the **AUTOSING** command. It is possible to generate erroneous answers if singularities are present in important parts of a model and processing is allowed to continue.

4. The use of **AUTOSPC** and **AUTOOMIT** is recommended for removing singularities prior to decomposition.

5. When the **AUTOSING** command is not used, the **PRINT**, **NONULL**, **NOZERO**, **COMPUTE**, and **GO** options are automatically in effect.

6. You may provide default values for some of the data specified on this command using your Configuration. Defaults are found in the **[eBase:matlib]** Component. You can print the contents of your Configuration with the command:

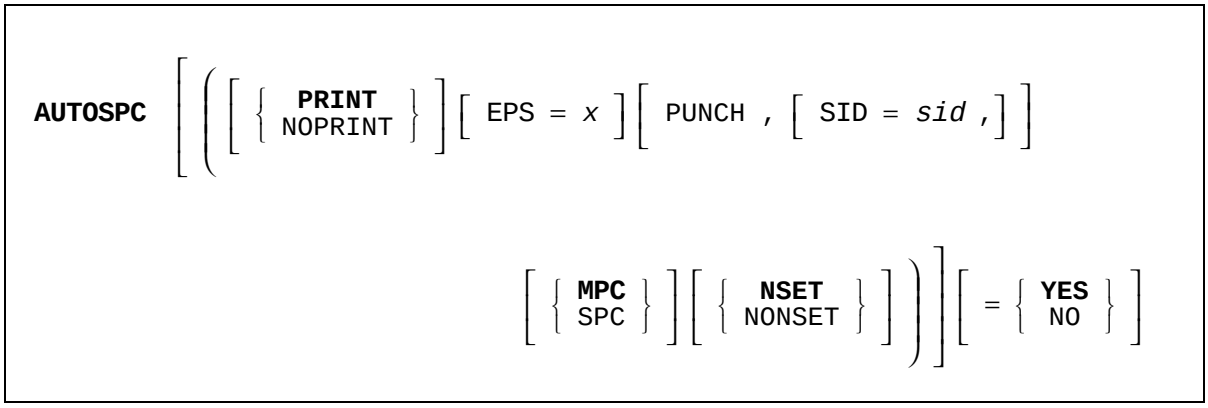
   ```
   PRINT CONFIGURATION
   ```

   Your System Support Specialist can provide you with more information about this advanced feature.

Case Control Command AUTOSPC

Requests that stiffness singularities and near singularities be automatically moved to the *m-set* or *s-set*.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Enables and disables the printing of a summary table of degrees of freedom that were found to be singular and were automatically constrained. The default value of **PRINT** may be modified by your System Administrator.	Keyword
x	Specifies the stiffness ratio *x* below which degrees of freedom will be defined as singular.	Real Default=10^{-8}
PUNCH, *sid*	Requests that **SPC** or **MPC** Bulk Data be written to the **BULK** file using *sid* as the set identification number. [1]	Keyword and Integer>0 Default *sid* **=999**
{ **MPC** / SPC }	Selects the **MPC** option in which dependency equations are generated to constrain the singularity, or the **SPC** option which causes the weakest direction of singularity to be constrained as a boundary with **SPC**s. [2]	Keyword
{ NSET / **NONSET** }	Requests that, for nonlinear analyses, the **AUTOSPC** operation be performed only on the *g-set* matrices, **NONSET**, or on both the *g-set* and *n-set* matrices, **NSET**. [3]	Keyword
{ **YES** / NO }	Enables or disables the **AUTOSPC** feature. [4]	Keyword

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=BULK** if you use the **PUNCH** option. Note that you must select **YES** to **PUNCH** these data.
2. The **MPC** option may be somewhat more expensive than the **SPC** option. However, it is exact. The **SPC** option may generate unwanted springs-to-ground.

Rev: V20.1

3. In linear analyses, the **AUTOSPC** operations are always performed on both *g-set* and *n-set* degrees of freedom.

4. You may provide default values for some of the data specified on this command using your Configuration. Defaults are found in the **<Matrix Conditioning>** section of the **[UAI/NASTRAN]** Component of your Configuration. You can print the contents of your Configuration with the command:

   ```
   PRINT CONFIGURATION
   ```

   Your System Support Specialist can provide you with more information about this advanced feature.

5. The **AUTOSPC** command may be used within the SUBCASE level.

6. The **AUTOSPC** feature is very convenient for removing the intrinsic singularities in your model (e.g. normal rotations of plate elements). You should check very carefully that the degrees of freedom automatically constrained are not causing unwanted boundary conditions.

Examples:

1. Remove stiffness singularities for all degrees of freedom with stiffness ratios less than 10^{-6}:

   ```
   AUTOSPC(EPS=1.0-6)
   ```

2. Remove singularities, disable printing, and punch the resulting **MPC** Bulk Data with a **SET ID** of 100:

   ```
   AUTOSPC(NOPRINT,PUNCH,SID=100)=YES
   ```

Case Control Command AXISYMMETRIC

Selects boundary conditions for problems using two-dimensional harmonic elements or specifies the existence of fluid harmonics for hydroelastic problems.

Command Syntax:

AXISYMMETRIC = { SINE / COSINE / FLUID }

Option	*Meaning*	*Data Range*
{ SINE / COSINE / FLUID }	Selects **SINE** boundary conditions, **COSINE** boundary conditions, or indicates the existence of **FLUID** harmonics. [1,2]	Keyword

Remarks:

1. The **SINE** boundary condition will constrain components 1,3 and 5 at every ring for the zero harmonic whereas the **COSINE** boundary condition will constrain components 2,4 and 6.
2. This command is required for problems containing **CONEAX**, **TRAPAX** or **TRIAAX** elements.
3. If this command is used for hydroelastic problems, at least one harmonic must be specified using an **AXIF** Bulk Data entry.
4. **SPCAX** and **MPCAX** Case Control commands may also be used to define additional constraints.

Case Control Command B2GG

Selects one or more direct input symmetric damping matrices which are *g-set* size.

Command Syntax:

```
B2GG = { name      }
       { sid       }
       { ( name_list ) }
```

Option	*Meaning*	*Data Range*
name	Selects a single B2GG damping matrix which is defined with **DMIG** or **DMIAX** Bulk Data entries that reference *name*. [1,2,4]	Name
sid	Selects a set that contains a list of one or more B2GG damping matrix names. [3]	Integer
name_list	Explicit list of B2GG damping matrix names. [3]	Name

Remarks:

1. **DMIG** and **DMIAX** matrices are not used for damping unless selected with the **B2GG** or **B2PP** commands.
2. The matrix *name* must be symmetric in form.
3. All selected matrices are added to form the final B2GG matrix.
4. **B2GG** matrix input is available in Complex Eigenvalue, Transient Response and Frequency Response analyses.

Examples:

1. Select a damping matrix named **MYDAMP**:

```
B2GG = MYDAMP
```

2. Select three damping matrices named **DAMP1**, **DAMP2**, and **DAMP3**:

```
SET 1 = DAMP1,DAMP2,DAMP3
...
B2GG = 1
```

3. Define the three damping matrices in Example 2 using an explicit list:

```
B2GG = (DAMP1,DAMP2,DAMP3)
```

4 CASE

Case Control Command B2PP

Selects one or more direct input damping matrices which are *p-set* size.

Command Syntax:

```
B2PP = { name      }
       { sid       }
       { ( name_list ) }
```

Option	*Meaning*	*Data Range*
name	Selects a single B2PP damping matrix which is defined with **DMIG** or **DMIAX** Bulk Data entries that reference *name*. [1,3]	Name
sid	Selects a set that contains a list of one or more B2PP damping matrix names. [2]	Integer
name_list	Explicit list of B2PP damping matrix names. [2]	Name

Remarks:

1. **DMIG** and **DMIAX** matrices are not used for damping unless selected with the **B2PP** or **B2GG** commands.
2. All selected matrices are added to form the final B2PP matrix.
3. **B2PP** matrix input is available in Complex Eigenvalue, Transient Response and Frequency Response analyses.

Examples:

1. Select a damping matrix named **MYDAMP**:

```
B2PP = MYDAMP
```

2. Select three damping matrices named **DAMP1P**, **DAMP2P**, and **DAMP3P**:

```
SET 1 = DAMP1P,DAMP2P,DAMP3P
...
B2PP = 1
```

3. Define the three damping matrices in Example 2 using an explicit list:

```
B2PP = (DAMP1P,DAMP2P,DAMP3P)
```

Case Control Command BMFORCE

Requests that equivalent beam forces (moments, shears, axial loads, and torques) be computed for pseudo-elements, composed of collections of solid elements, defined by **BMFORCE** and **BMFORC1** Bulk Data entries.

Command Syntax:

```
BMFORCE [ ( [ { PRINT } , ] [ PUNCH ] ) ] = { ALL }
        [     { NOPRINT }                   ]   { sid }
                                                { NONE }
```

Option	*Meaning*	*Data Range*
{ **PRINT** } { NOPRINT }	Specifies that results be generated and written to the **PRINT** file, or that they be generated for later post-processing.	Keyword Default=**PRINT**
PUNCH	Requests that results be written to the **PUNCH** file. [1]	Keyword
{ **ALL** } { *sid* } { NONE }	Specifies that the equivalent beam forces are to be computed for **ALL** equivalent beams defined with **BMFORCE** and **BMFORC1** Bulk Data entries, or for those specified in the integer **SET** with identification number *sid*. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. The **BMFORCE** and **BMFORC1** Bulk Data entries define pseudo-elements, each of which has its own identification number. To select specific **BMFORCE** elements, you must define an integer **SET** in the Case Control packet which gives their identification numbers.
3. **BMFORCE** computations may be performed only in Statics and Normal Modes analyses.

New: V20.1

Case Control Command BOUNDARY

Defines the degrees of freedom in the boundary of a PHASE 1 Basic Substructure, or selects a set of degrees of freedom to be retained in the *a-set* during automatic reduction operations.

Command Syntax:

BOUNDARY *sid*

Option	*Meaning*	*Data Range*
sid	Selects a boundary set identification number. [1,2,3]	Integer>0

Remarks:

1. The identification number *sid* refers to **BDYS** or **BDYS1** Bulk Data entries. These data will not be used unless selected with this command.
2. The **BOUNDARY** Command is used when executing PHASE 1 for a Basic Substructure to define degrees of freedom that will be retained in the *a-set*. This is important if you are using the Dynamic Reduction, **AUTOOMIT** or **AUTOSPC** features. When performing Dynamic Reduction, any dof in the boundary are retained along with the generalized coordinates. For **AUTOOMIT** and **AUTOSPC**, the dof are ignored during the Guyan reduction and constraint elimination procedures, respectively.
3. The **BOUNDARY** command may also be used in any job to insure that specified degrees of freedom will be retained in the *a-set*. By doing so, these degrees of freedom will be uneffected by Dynamic Reduction or the **AUTOOMIT**, **AUTOSPC**, **AUTOREDUCE** and **NLREDUCE** operations.

Case Control Command CASE

Delimits and identifies an analysis CASE when using the Multidisciplinary Solution Sequence, **SOL MULTI**.

Command Syntax:

```
                                  ┌ ┌ STATICS                  ┐ ┐
                                  │ │                          │ │
                                  │ │ ┌ STRUCTURAL ┐           │ │
                                  │ │ │ FLUID      │ MODES     │ │
                                  │ │ └ COUPLED    ┘           │ │
                                  │ │                          │ │
CASE  caseid                      │ { REDUCE                   } │
                                  │ │ MREDUCE                  │ │
                                  │ │ FREQUENCY                │ │
                                  │ │ MFREQUENCY               │ │
                                  │ │ TRANSIENT                │ │
                                  │ │ MTRANSIENT               │ │
                                  │ │ AERO                     │ │
                                  │ └ FLUTTER                  ┘ │
                                  └                              ┘
```

Option	*Meaning*	*Data Range*
caseid	Defines the CASE identification number. [1]	Integer>0
{ STATICS, MODES, FREQUENCY, MFREQUENCY, TRANSIENT, MTRANSIENT, AERO, FLUTTER }	Selects the analysis discipline from **STATICS**, Normal **MODES** analysis, Direct **FREQUENCY** Response analysis, Modal Frequency Response analysis (**MFREQUENCY**), **TRANSIENT** Response analysis, Modal Transient Response analysis (**MTRANSIENT**), **AERO**dynamic analysis, or **FLUTTER** analysis. [2,3,4]	Character
{ REDUCE, MREDUCE }	Requests either the Guyan reduction or modal reduction procedure. [5,6]	Character
{ **STRUCTURAL**, FLUID, COUPLED }	Type of modal extraction being performed in a hydroelastic or acoustic analysis for subsequent use as generalized coordinates for a modal response analysis **CASE**. [7]	Character

Remarks:

1. Each **CASE** identification number, *caseid*, must be greater than all previous **CASE** identification numbers. Note that **SUBCASE** may also be used for compatibility with other versions of NASTRAN.
2. For Modal Frequency Response and Modal Transient Response analyses in **SOL MULTI**, you must first define a **CASE MODES** that has ***the same boundary conditions*** as will be used by the dynamics response **CASE**. Then, in the dynamics response **CASE** you must use the **USING** command. See Example 1 below.

Rev: V20.1

3. When performing Frequency Response and Transient Response analyses, you may also use the command forms **MFREQUENCY** and **MTRANSIENT** to indicate that the modal approach is being used. These analyses require a previous **CASE MODES** to extract the mode shapes that will be used as generalized coordinates, and a **USING** command to select them. If you select the commands **FREQUENCY** or **TRANSIENT**, **AND** you have a **USING** command, then the modal approach will automatically be used. (See Example 1 below.)
4. See Chapter 28 of the ***User's Guide*** for a description of the use of **CASE**s in Aerodynamic and Flutter analyses.
5. When performing **REDUCE**, you must have a previous **CASE MODES** which defines the modes of the baseline model. The **USING** Case Control command must then reference these modes.
6. See Chapter 6 of the ***User's Guide*** for a description of the use of **CASE**s in performing model reductions.
7. See Chapter 23 of the ***User's Guide*** for a description of the use of **CASE**s in Hydroelastic and Acoustic analyses.

Examples:

1. Use the normal modes computed in Case **1001** as generalized degrees of freedom for a Modal Frequency Response case, **2001**:

```
CASE 1001 MODES
   MPC = 5
   SPC = 10
   ...
CASE 2001 FREQUENCY ( or you may use MFREQUENCY)
   MPC = 5
   SPC = 10
   USING MODES 1001
```

2. Use the normal modes computed in Case **11** as generalized degrees of freedom for a Flutter analysis Case, **22**, which uses Aerodynamic loads stored on an ***eBase*** database with the logical name my_aero:

```
CASE 11 MODES
   MPC = 5
   SPC = 10
   ...
CASE 22 FLUTTER
   FMETHOD = PKITER
   FLSOLVE = 1
   DENS    = 100
   VLIST = 1000
   USING MODES 11
   USING AERO FROM my_aero
```

Case Control Command CMETHOD

Selects a complex eigenvalue extraction method and control parameters.

Command Syntax:

```
CMETHOD = sid
```

Option	*Meaning*	*Data Range*
sid	Selects an **EIGC** Bulk Data entry identification number.	Integer>0

Remarks:

1. The **CMETHOD** command must be used to extract complex eigenvalues.

Case Control Command DEFORM

Selects an Element Deformation Set to be applied to the structural model.

Command Syntax:

```
DEFORM = defid
```

Option	*Meaning*	*Data Range*
defid	Selects a **DEFORM** Bulk Data entry identification number. [1]	Integer>0

Remarks:

1. The **DEFORM** Bulk Data will not be used unless selected with this command.
2. **DEFORM** is only available in Static, Nonlinear Static, Inertia Relief, Differential Stiffness, and Buckling analyses.

Case Control Command DENS

Defines the atmosphere or fluid density values where a Flutter solution will be computed.

Command Syntax:

```
DENS = { aid          }
       { SET sid      }
       { ( dens_list ) }
```

Option	*Meaning*	*Data Range*
aid	Selects an **ATMOS** Bulk Data entry defining Density-Altitude values. [1,2]	Integer>0
sid	Specifies the the identification number of a Real **SET** command defining the values.	Integer>0
dens_list	Gives an explicit list of air density values.	Real>0

Remarks

1. The **ATMOS** Bulk Data entry allows the use of equivalent airspeed input and output with a variety of altitude-density points.
2. If you wish to analyze *match point* speed conditions only, then the **DENS** command must reference an **ATMOS** Bulk Data entry. Off *match point* conditions can also be analyzed by using the **VLIST** Case Control command with the **DENS** command. **DENS** may also select a set of fluid density points.
3. The **DENS** command is **required** if you have specified a **CASE FLUTTER**.

Examples:

1. Define a set with identification number **100** having three densities, **0.1**, **0.2** and **0.3**. Select this set as the densities used in a Flutter analysis.

```
SET 100 = 0.1,0.2,0.3
DENS = 10
```

2. Define the same three densities explicitly:

```
DENS = ( 0.1,0.2,0.3 )
```

Case Control Command DESCON

Selects design constraints.

Command Syntax:

```
DESCON = { ALL   }
         { rcsid }
```

Option	*Meaning*	*Data Range*
{ **ALL** } { *rcsid* }	Selects **ALL** response constraint Bulk Data entries or a set of response constraints. [1,2]	Integer>0

Remarks:

1. The set identification number *rcsid* references a Case Control set which in turn lists the constraint set identification numbers of the desired **DCELEM**, **DCGRID**, **DCFREQ**, and **DCMODE** Bulk Data entries.
2. If you select **ALL**, then all of the constraint Bulk Data entries in the Bulk Data packet are used.
3. The **DESCON** command is used only when performing Sensitivity analyses.

Case Control Command DISPLACEMENT

Requests displacement vector results.

Command Syntax:

```
DISPLACEMENT [ ( [ { PRINT } , ] [ PUNCH , ] [ TIC = sid , ]
               [ { NOPRINT }   ]                [ { RECT } ] ) ] [ = { ALL } ]
                                                [ { POLAR } ]       [ { sid } ]
                                                                    [ { NONE } ]
```

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
TIC = *n*	Requests that results be written to the **BULK** file as transient initial condition (**TIC**) Bulk Data entries. The value of integer **n** will be used for the initial condition set id. [2]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response problems.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [3]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=BULK** if you use the **TIC** option.
3. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.

Examples:

1. Print the displacements for all points using Rectangular format.

   ```
   DISPLACEMENT(RECT) = ALL
   ```

   or, equivalently since the selected option is the default:

   ```
   DISPLACEMENT = ALL
   ```

2. Print the displacements defined by **SET 100** in **SORT2** order using Polar format. Also, write the results on the **PUNCH** file:

   ```
   SORT = SORT2
   ...
   DISPLACEMENT(POLAR,PUNCH)=100
   ```

Case Control Command DLOAD

Selects dynamic loads and enforced motions to be applied in a Transient or Frequency Response analysis.

Command Syntax:

DLOAD = *lid*

Option	*Meaning*	*Data Range*
lid	Selects a dynamic load set identification number. [1]	Integer>0

Remarks:

1. The identification number *lid* references any dynamic loads or enforced motions defined with **DLOAD**, **DLOAD1**, **RLOAD1**, **RLOAD2**, **TLOAD1**, **TLOAD2**, and **ACSRCE** Bulk Data entries. These data will not be used unless selected with the **DLOAD** command.
2. The **RLOAD1**, **RLOAD2** , and **ACSRCE** Bulk Data entries are used only in Frequency Response analysis while **TLOAD1** and **TLOAD2** Bulk Data entries are used only in Transient Response analysis.
3. The **DLOAD**, **DLOAD1**, and **ACSRCE** Bulk Data entries may be used to combine dynamics loads defined separately using **RLOAD1, RLOAD2, TLOAD1**, **TLOAD2** , and **ACSRCE** Bulk Data entries.

Case Control Command DSCOEFFICIENT

Selects the coefficient set for a Differential Stiffness problem.

Command Syntax:

DSCOEFFICIENT = { DEFAULT / *sid* }

Option	*Meaning*	*Data Range*
{ DEFAULT / *sid* }	Selects a **DSFACT** Bulk Data entry identification number or a single **DEFAULT** coefficient whose value is 1.0. [1]	Keyword or Integer>0

Remarks:

1. The **DSFACT** Bulk Data will not be used unless selected with this command.
2. The **DSCOEFFICIENT** command must appear in the second SUBCASE of a Differential Stiffness analysis.

Examples:

1. Select a coefficient set whose **DSFACT** set identification number is 3:

```
DSCOEFFICIENT = 3
```

2. Select the default coefficient:

```
DSCO = DEFAULT
```

Case Control Command DSYM

Specifies that a model has either one or two planes of overall symmetry when using the **DIHEDRAL** Cyclic Symmetry feature.

Command Syntax:

DSYM = { S, A, SS, SA, AS, AA }

Option	*Meaning*	*Data Range*
{ S, A }	Specifies that the model is **S**ymmetrical or **A**ntisymmetrical with respect to Side 1.	Keyword
{ SS, SA, AS, AA }	Specifies that the model has two planes of symmetry. The first symbol indicates that the model is **S**ymmetrical or **A**ntisymmetrical with respect to Side 1. The second indicates that the model is also **S**ymmetrical or **A**ntisymmetrical with respect to a plane perpendicular to Side 1.	Keyword

Examples:

1. Specify that your dihedral symmetry model is antisymmetrical about Side 1:

   DSYM = A

2. Indicate that your model has two planes of symmetry; antisymmetric about Side 1 and symmetric with respect to a plane perpendicular to Side 1:

   DSYM = AS

Case Control Command DYNRED

Selects the Dynamic Reduction feature.

Command Syntax:

```
DYNRED = sid
```

Option	*Meaning*	*Data Range*
sid	Selects a **DYNRED** Bulk Data entry identification number.	Integer>0

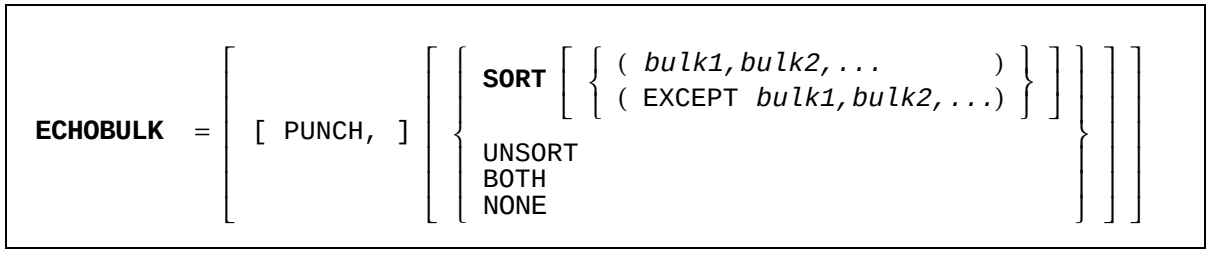
Remarks:

1. Dynamic Reduction is available in Normal Modes, Complex Eigenvalue, and dynamic response analyses as an alternative to Guyan Reduction for reducing the size of the eigenvalue problem.
2. The *a-set* will contain physical degrees of freedom specified with **ASETi** and **SUPORT** Bulk Data entries plus SCALAR point degrees of freedom associated with a set of generalized coordinates. All required SCALAR points are generated automatically.

Case Control Command ECHOBULK

Requests a complete echo of the Bulk Data packet.

Command Syntax:



Option	*Meaning*	*Data Range*
PUNCH	Requests that the sorted Bulk Data packet be written on the **BULK** file. [1]	Keyword
{ **SORT** / UNSORT / BOTH / NONE }	Selects the Bulk Data echo format that will appear in the print file. The options are an alphanumeric **SORT**ed echo, an **UNSORT**ed echo, or **BOTH** echoes. **NONE** disables printing and punching of the Bulk Data echo. [2,3]	Keyword
bulki	Selects which Bulk Data entries are to be echoed, or if the keyword **EXCEPT** is used, which Bulk Data entries are not to be echoed in the print file. [3]	Any legal Bulk Data entry names

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=BULK** file if you use the **PUNCH** option.
2. If no **ECHO** command appears a **SORT**ed echo will be written to the print file. This default behavior may be overridden by an entry in your Configuration. Defaults are found in the **<Print File Controls>** section of the **[UAI/NASTRAN]** Configuration Component. You can print the contents of your Configuration with the command:

   PRINT CONFIGURATION

   Your System Support Specialist can provide you with more information about this advanced feature.
3. If you use the (*bulk1, bulk2, ...*) or (**EXCEPT** *bulk1, bulk2,...*) options to the **SORT** argument, these must follow any use of the **PUNCH** arguement.
4. The Bulk Data entries **ECHOOFF** and **ECHOON** may be used throughout the Bulk Data packet to selectively control Bulk Data echo if the **UNSORT**ed echo request is made in the Case Control packet.

Case Control Commands ECHOOFF and ECHOON

Disables or enables the normal echo of Case Control commands.

Command Syntax:

```
ECHOOFF
ECHOON
```

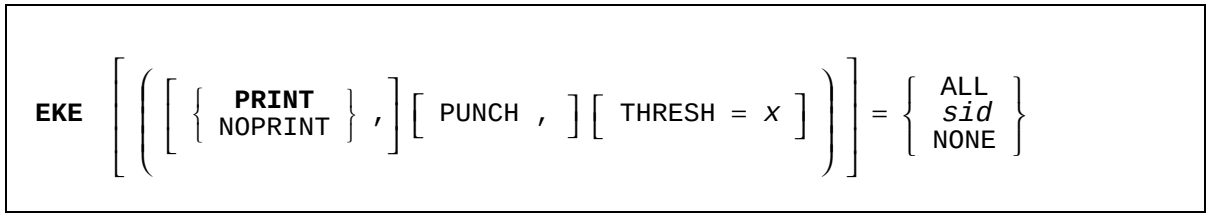
Remarks:

1. The echo of consecutive Case Control commands may be removed from the print file by insertion of the **ECHOOFF** command into the Case Control packet. Echo may be reactivated by use of the **ECHOON** command.
2. As many pairs of **ECHOOFF**, **ECHOON** commands may be used as desired.
3. **ECHOON** is automatically invoked for the Bulk Data packet at the completion of processing the Case Control packet. **ECHO** of the Bulk Data packet is then controlled by Bulk Data entries **ECHOOFF** and **ECHOON**.

Case Control Command EKE

Requests element kinetic energy results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written to the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0
x	Requests that only elements whose kinetic energy is at least *x* percent of the total kinetic energy be selected.	0.0≤Real≤100.0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. You may provide a default value for *x* in your Configuration. The default is found in the **<Data Checking>** section of the **[UAI/NASTRAN]** Configuration Component. You can print the contents of the Configuration with the command:

   PRINT CONFIGURATION

   Your System Support Specialist can provide you with more information about this advanced feature.
4. Results are only available in **SORT1** order.
5. Element kinetic energies are available for Normal Modes (including Design Sensitivity and MDO) and Frequency Response analyses.

Examples:

1. Print the kinetic energies for all elements and write the results to the **PUNCH** file:

```
EKE(PUNCH)=ALL
```

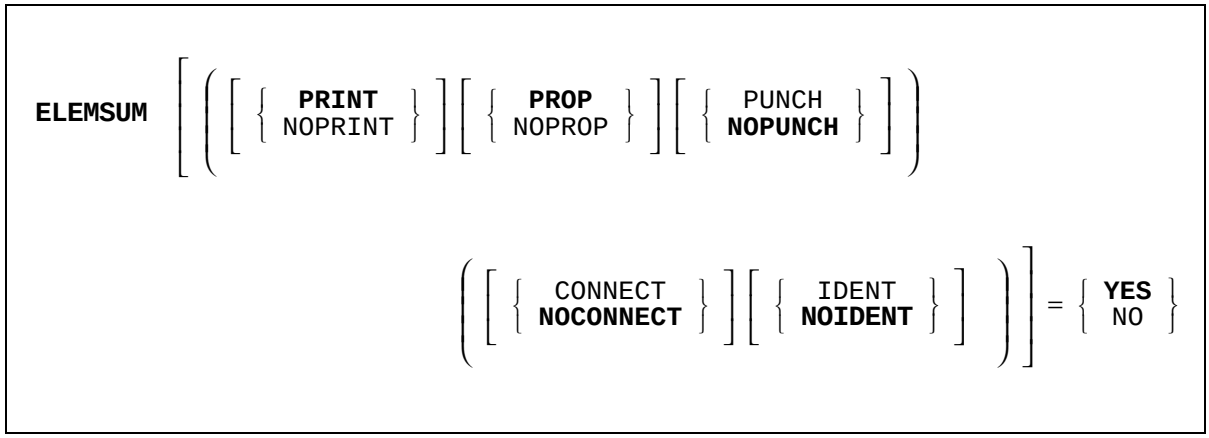
2. Print the kinetic energy for all elements which contribute 2% or more to the total kinetic energy of the model.

```
EKE(PRINT,THRESH=2.0) = ALL
```

Case Control Command ELEMSUM

Requests a summary of finite element characteristics.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that the element summary be generated and then written to the print file.	Keyword
{ **PROP** / NOPROP }	Selects or deselects a summary of the element physical properties in the model, sorted by element identification number. This includes element length, area, volume, weight and mass, and the total of each. Additionally, each of the quantities is subtotaled for each element type.	Keyword
{ PUNCH / **NOPUNCH** }	Requests that **PROP** results be written on the **PUNCH** file. [1]	Keyword
{ CONNECT / **NOCONNECT** }	Selects or deselects the element connection summary table which gives, for each GRID point, a list of element identification numbers, and their element types, that are connected to it.	Keyword
{ IDENT / **NOIDENT** }	Selects or deselects a report giving the element identification number and element type for all elements in the model, sorted by element identification number.	Keyword
{ **YES** / NO }	Enables or disables the **ELEMSUM** options.	Keyword

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option. Note that only the **GEOM** table may be written to the **PUNCH** file.

Example:

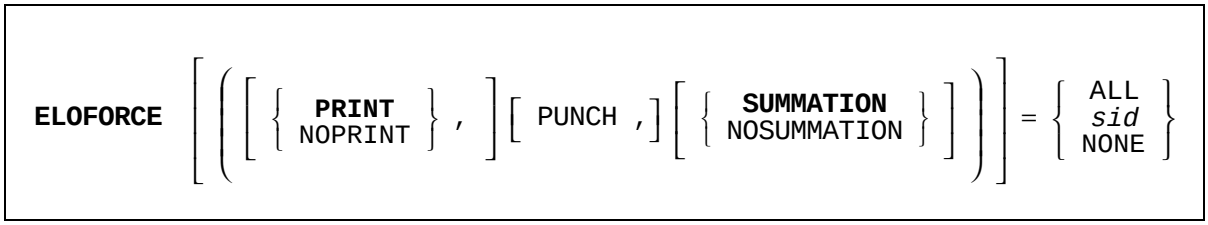
1. Print and punch the element summary and print the connection tables:

```
ELEMSUM(PRINT,PUNCH,CONNECT) = YES
```

Case Control Command ELOFORCE

Requests element force results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written to the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **SUMMATION** / NOSUMMATION }	Requests that results be generated for the **SUMMATION** of element-oriented forces along adjacent edge directions, or that **NOSUMMATION** will be performed.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** GRID points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the GRID points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. Element-oriented forces are available for Static, Normal Modes and Frequency Response analyses.
4. Results are only available in **SORT1** order.
5. The element-oriented forces are computed automatically from the GRID point force balance results. You do not need to use the **GPFORCE** command, but if you wish to have a summary of these forces you may use the command at no additional computational cost.

Example:

1. Print the element-oriented forces, but not the summary, for the points defined in **SET 5**:

```
ELOFORCE(NOSUMMARY) = 5
```

2. Compute, but do not print, the element-oriented forces for all points in **SET 32** and write the results to the **PUNCH** file:

```
ELOFORCE(NOPRINT,PUNCH) = 32
```

Case Control Command EQUILIBRIUM

Requests the computation of force equilibrium checks about the origin of the Basic Coordinate System or about a selected GRID point.

Command Syntax:

```
EQUILIBRIUM [ = gid ]
```

Option	*Meaning*	*Data Range*
gid	Selects a GRID point identification number about which the equilibrium is computed. If absent, the origin of the Basic Coordinate System is used.	Integer>0

Remarks:

1. The forces on GRID points are computed for the directly applied loads, single-point constraint forces, and the multi-point constraint forces.
2. Force equilibrium may be obtained only for linear Static Analysis and linear Static Analysis with Inertia Relief.

Case Control Command ESE

Requests element strain energy results.

Command Syntax:

```
ESE [ ( ( [ { PRINT } , ] [PUNCH,] [THRESH = x ] ) ) ] = { ALL }
    [                 ]                                  { sid }
    [ { NOPRINT }     ]                                  { NONE }
```

Option	*Meaning*	*Data Range*
{ **PRINT** } { NOPRINT }	Specifies that results be generated and then written to the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL } { *sid* } { NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0
x	Requests that only elements whose strain energy is at least *x* percent of the total strain energy be selected. [3]	0.0≤Real≤100.0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. You may provide a default value for *x* in your Configuration. The default is found in the **<Data Checking>** section of the **[UAI/NASTRAN]** Configuration Component. You can print the contents of your configuration with the command:

   PRINT CONFIGURATION

   Your System Support Specialist can provide you with more information about this advanced feature.
4. Results are only available in **SORT1** order.
5. Element strain energies are available for Static, Normal Modes, and Frequency Response analyses.

Examples:

1. Print the strain energies for all elements and write the results to the **PUNCH** file:

```
ESE(PUNCH)=ALL
```

2. Print the strain energy for all elements which contribute 2% or more to the total strain energy of the model.

```
ESE(PRINT,THRESH=2.0) = ALL
```

Case Control Command EXPORT REDUCED MODEL

Requests that reduced model matrices, and the associated GRID Points and SCALAR Points, be written to the Bulk Data file as **DMIG** Bulk Data entries.

Command Syntax:

```
EXPORT REDUCED MODEL [TO logical_name] [STIFFNESS = stiff_name ] [MASS = mass_name ]
                     [GRID [ ( { OFFSET } ) = { n } ] ] [SCALAR [ ( { OFFSET } ) = { n } ] ]
                                { START }       { NO }              { START }       { NO }
```

Option	*Meaning*	*Data Range*
logical_name	Selects the logical name of an **ASSIGN**ed file where the Bulk Data entries will be written. [1]	Character
stiff_name	Requests that the reduced stiffness matrix be written to **DMIG** Bulk Data entries and assigned the name *stiff_name*.	Character
mass_name	Requests that the reduced mass matrix be written to **DMIG** Bulk Data entries and assigned the name *mass_name*.	Character
GRID	Requests the generation of **GRID** point Bulk Data entries corresponding to the degrees of freedom in the reduced matrices. [2]	Keyword
SCALAR	Requests the generation of scalar point, **SPOINT**, Bulk Data entries corresponding to the degrees of freedom in the reduced matrices. [2]	Keyword
{ OFFSET } { START }	Identification numbers of GRID points or **SPOINT**s may be selected by adding an **OFFSET** to last Grid point or **SPOINT** in the baseline model, or by **START**ing the numbers from a new value. The new GRID points or **SPOINT**s are automatically output unless disabled with the option described next. [2]	Keyword
{ *n* } { NO }	Value of the **OFFSET** *n* to to last Grid point or **SPOINT** in the baseline model, or the **START**ing value *n*, depending on the option selected above. **NO** will disable output of the selected item. [2]	Integer>0

New: V20.1

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=BULK** when using this command.

2. When **EXPORT**ing a **STATICS** model, the output **GRID** point Bulk Data will define the degrees of freedom retained during the reduction plus any GRID points that were defined to be in the **BOUNDARY** set. Similarly, **SPOINT** Bulk Data will be created for any SCALAR points that were placed in the **BOUNDARY** set or retained during the reduction. In all cases, these new data will use the numbering rule defined by the **GRID** and **SCALAR** options.

   For a **MODAL** model, the resulting data will be **SPOINT** Bulk Data for the generalized degrees of freedom, and **GRID** Bulk Data for the degrees of freedom retained during the reduction plus any points that were defined to be in the **BOUNDARY** set. Similarly, **SPOINT** Bulk Data will be created for any scalar points that were placed in the **BOUNDARY** set. In each case, these new data will use the numbering rule defined by the **GRID** and **SCALAR** options.

5. The **EXPORT** command may only be used with **SOL MULTI**, and it must appear in a **CASE REDUCE STATICS** or **CASE REDUCE MODAL**.

Example:

1. Write the reduced stiffness and mass matrices to the Bulk file.

```
CASE 101 REDUCE STATICS
...
   EXPORT REDUCED MODEL STIFFNESS=KMTX,MASS=MMTX,GRID(OFFSET)=10000
...
```


New: V20.1

Case Control Command FLPRINT

Requests summary or detailed print when performing Flutter analysis.

Command Syntax:

```
FLPRINT = { DETAIL  }
          { SUMMARY }
          { NONE    }
```

Option	*Meaning*	*Data Range*
{ DETAIL / SUMMARY / **NONE** }	Requests printing of Flutter analysis solution results. [1,2]	Keyword

Remarks:

1. The following table describes the output printed for each option:

OPTION	Results Printed
DETAIL	Prints a history of of PK Flutter iterations, v-g curve data, generalized aerodynamic curve fit error checks, and the flutter speed crossing summary.
SUMMARY	Prints v-g curve data, generalized aerodynamic curve fit error checks, and flutter speed crossing summary.
NONE	Prints only the generalized aerodynamic curve fit error checks and flutter speed crossing summary.

2. **FLPRINT** is only used if you have specified a **CASE FLUTTER**.
3. Example output is found in Chapter 28 of the ***User's Guide***.

Case Control Command FLSOLVE

Selects solution options for Flutter analysis.

Command Syntax:

```
FLSOLVE = fid
```

Option	*Meaning*	*Data Range*
fid	Selects an **FLSOLVE** Bulk Data entry identification number. [1]	Integer>0

Remarks:

1. **FLSOLVE** is only used if you have specified a **CASE FLUTTER**.

Case Control Command FLUX

Requests flux and gradient results for elements in Heat Transfer analysis.

Command Syntax:

FLUX ([{ **PRINT** / NOPRINT }] [,PUNCH]) = { ALL / *sid* / NONE }

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** elements, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the elements at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **FLUX** output are available only when performing Heat Transfer analyses.

Example:

1. Print the flux for all elements in **SET 1000** and write the results to the **PUNCH** file:

```
FLUX(PUNCH)=1000
```

Case Control Command FMETHOD

Selects a Flutter analysis method.

Command Syntax:

```
FMETHOD = { K      }
          { PKITER }
          { PKSWEEP }
```

Option	*Meaning*	*Data Range*
{ K / PKITER / PKSWEEP }	Selects the Flutter method. [1,2,3,4,5]	Keyword

Remarks:

1. All methods require the following Case Control commands:

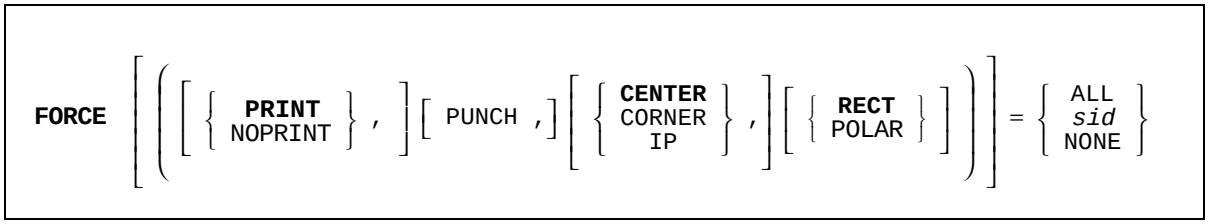
   SYMMETRY
   DENS
   USING MODES

2. The **K** option selects the *American K Method*. This method generally requires the Case Control command **KLIST**.
3. The **PKITER** option selects the *Iterative PK Method*. No additional Case Control commands are required.
4. The **PKSWEEP** option selects the *Frequency Sweep PK Method*. This method generally requires the Case Control command **KLIST**.
5. The **FMETHOD** command is **required** if you have specified a **CASE FLUTTER**.

Case Control Command FORCE

Requests element force results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written to the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **CENTER** / CORNER / IP }	Selects the location of force computation within two- and three-dimensional elements. [2,3]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response problems.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** elements, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the elements at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. The selection of force locations depends on the type of elements in your model. The following table describes these.

CENTER	Element forces for all two- and three-dimensional elements are printed at the element center. In addition, the TRIA6, QUAD8, PENTA, HEXA and TETRA elements print the forces at the corner points of each element. This is the default value.
CORNER	Elements forces for any TRIA3, TRIAR, QUAD4, and QUADR elements are computed for, and printed at, the corner GRID points and center of each element. This is also true for the TRIA6, QUAD8, PENTA, HEXA and TETRA elements as above.
IP	Element forces are computed and printed at the integration points and center for all plate and solid elements. In the output, these are labelled by the GRID point closest to the integration point.

Rev: V20.1

3. If the **IP** option is selected by **any FORCE**, **STRAIN**, or **STRESS** Case Control command, then the **IP** option will be used for **all** of these output requests.
4. The **FORCE** output are available in all analysis disciplines.

Example:

1. Print the element forces for all elements in **SORT2** using Polar format. Also write the results to the **PUNCH** file:

```
SORT = SORT2
...
FORCE(PRINT,POLAR,PUNCH) = ALL
```

2. Request that the element forces in **SET 10** be generated for subsequent post-processing:

```
FORCE(NOPRINT)=10
```

Case Control Command FREQUENCY

Selects the set of excitation frequencies in Frequency Response problems.

Command Syntax:

FREQUENCY = *freqid*

Option	*Meaning*	*Data Range*
freqid	Selects a frequency set identification number. [1]	Integer>0

Remarks:

1. The identification number *freqid* references **FREQ**, **FREQ1, FREQ2** and **FREQ3** Bulk Data entries. These data will not be used unless selected with the **FREQUENCY** command.
2. Any number of **FREQ**, **FREQ1, FREQ2** or **FREQ3** Bulk Data entries may have the same set identification in which case the frequency set is the union of all frequencies thus created.
3. The **FREQUENCY** command is available only for Frequency Response analyses.

Case Control Command GPFORCE

Requests GRID point force balance for applied loads, single-point constraints, and element constraints.

Command Syntax:

```
GPFORCE [ ( [ { PRINT } , ] [ PUNCH ] ) ] = { ALL }
        [     { NOPRINT }                   ]   { sid }
                                                { NONE }
```

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The GRID point Force Balance is available for linear Static, Normal Modes and Frequency Response analyses.
4. Results are only available in **SORT1** order.

Examples:

1. Print the GRID point force balance for the points defined in **SET 5**:

```
GPFORCE = 5
```

2. Compute, but do not print, the GRID point force balance for all points in **SET 32** and write the results to the **PUNCH** file:

```
GPFO(NOPRINT,PUNCH) = 32
```

4 CASE

Case Control Command GPKE

Requests GRID point kinetic energy output for normal modes analysis.

Command Syntax:

```
GPKE [ ( [ { PRINT } , ] [ PUNCH ] ) ] = { ALL }
     [     { NOPRINT }                   ]   { sid }
                                           { NONE }
```

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The GRID point Kinetic Energy is available for Normal Modes analyses.

Examples:

1. Print the GRID point kinetic energy for all points in the model:

```
GPKE = ALL
```

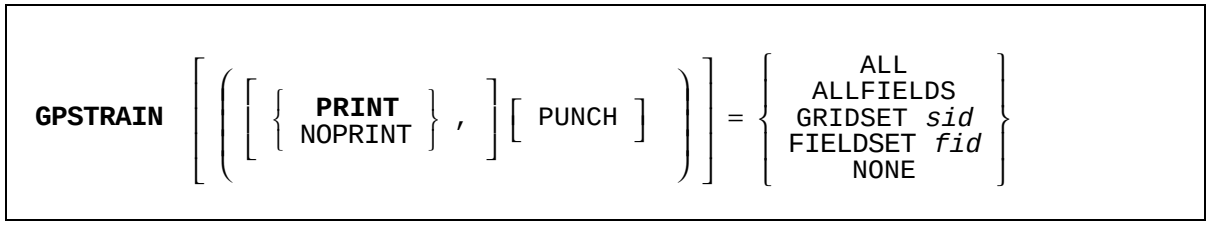
2. Compute, but do not print, the GRID point kinetic energies for all points in **SET 102** and write the results to the **PUNCH** file:

```
GPKE(NOPRINT,PUNCH) = 102
```

Case Control Command GPSTRAIN

Requests GRID point strain output.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
ALL	Selects data for **ALL** GRID points. [2]	Keyword
ALLFIELDS	Requests that GRID point strains be recovered for all GRID points in stress/strain fields defined on **GPFIELD** Bulk Data entries. [3]	Keyword
GRIDSET *gsid*	Requests that GRID point strains be recovered at all eligible GRID points defined in the Case Control Integer **SET** *gsid*. [3]	Keyword and Integer>0
FIELDSET *fsid*	Requests that GRID point strains be recovered at all eligible GRID points in those stress/strain fields defined with **GPFIELD** Bulk Data having the set identification number *fsid*. [3]	Keyword and Integer>0
NONE	Disables the option.	Keyword

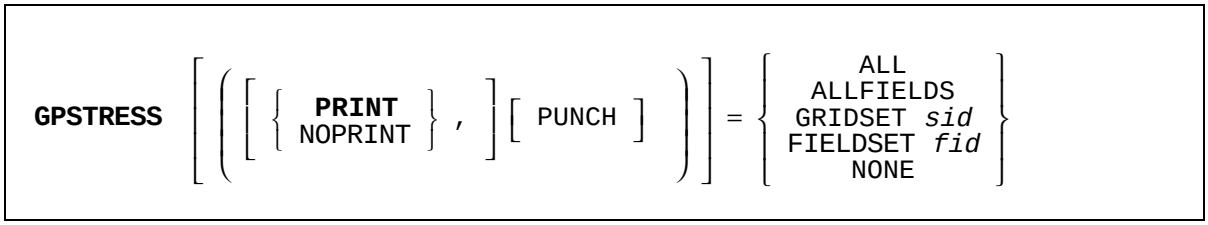
Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. If a request for element strain output does not exist for any of the elements involved in the GRID point strain recovery, a **STRAIN(NOPRINT)** request will automatically be generated for that element.
3. Stress/strain fields are described in the ***User's Guide***.
4. The GRID point strain output are available in Static, Normal Modes, Buckling, and Differential Stiffness analysis. It is also available for linear elements in Nonlinear Static analysis.

Case Control Command GPSTRESS

Requests GRID point stress output.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
ALL	Selects data for **ALL** GRID points. [2]	Keyword
ALLFIELDS	Requests that GRID point stresses be recovered for all GRID points in stress/strain fields defined on **GPFIELD** Bulk Data entries. [3]	Keyword
GRIDSET *gsid*	Requests that GRID point stresses be recovered at all eligible GRID points defined in the Case Control Integer **SET** *gsid*. [3]	Keyword and Integer>0
FIELDSET *fsid*	Requests that GRID point stresses be recovered at all eligible GRID points in those stress/strain fields defined with **GPFIELD** Bulk Data having the set identification number *fsid*. [3]	Keyword and Integer>0
NONE	Disables the option.	Keyword

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. If a request for element stress output does not exist for any of the elements involved in the GRID point stress recovery, a **STRESS(NOPRINT)** request will automatically be generated for that element.
3. Stress/strain fields are described in the ***User's Guide***.
4. The GRID point stress output are available in Static, Normal Modes, Buckling, and Differential Stiffness analysis. It is also available for linear elements in Nonlinear Static analysis.

Case Control Command **HARMONICS** (Axisymmetric Harmonic Modeling)

Controls the number of harmonics output for Axisymmetric Harmonic elements.

Command Syntax:

HARMONICS = { ALL / *nharm* / NONE }

Option	*Meaning*	*Data Range*
{ ALL / *nharm* / NONE }	For Axisymmetric Harmonic analysis, enables results generation for **ALL** points or specifies the number of harmonics, *nharm*, for which output will be generated. **NONE** disables output requests. [1,2]	Keyword or Integer≥0

Remarks:

1. Requested output will be prepared for all harmonics up to and including *nharm*.
2. If you do not use the **HARMONICS** command in Axisymmetric Harmonics analysis, output will be generated only for the zero harmonic.

Case Control Command **HARMONICS** (Cyclic Symmetry Modeling)

Selects which harmonics are used in Cyclic Symmetry analyses.

Command Syntax:

HARMONICS = { ALL / *sid* }

Option	*Meaning*	*Data Range*
{ ALL / *sid* }	For Cyclic Symmetry analysis, selects the use of **ALL** or a specific set of harmonics, defined by the set *sid*, which will be used in the solution. [1]	Keyword or Integer>0

Remarks:

1. The **HARMONICS** command is required when you perform Cyclic Symmetry analyses.

Case Control Command HOUTPUT

Requests harmonic output in Cyclic Symmetry analyses.

Command Syntax:

$$\textbf{HOUTPUT}\left(\left\{\begin{matrix}\text{C}\\ \text{S}\\ \text{C*}\\ \text{S*}\end{matrix}\right\}\right)=\left\{\begin{matrix}\text{ALL}\\ \textit{sid}\end{matrix}\right\}$$

Option	*Meaning*	*Data Range*
$\left\{\begin{matrix}\text{C}\\ \text{S}\\ \text{C*}\\ \text{S*}\end{matrix}\right\}$	Selects the harmonic coefficient.	Keyword
$\left\{\begin{matrix}\text{ALL}\\ \textit{sid}\end{matrix}\right\}$	Enables results generation and printing for **ALL** harmonics, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the harmonics at which results will be generated and printed. [1]	Keyword or Integer>0

Remarks:

1. The **ALL** option requests output for all harmonics specified on the **HARMONICS** Case Control command.
2. Either this command, or the **NOUTPUT** command, must be present to recover Cyclic Symmetry solution results.

Examples:

1. Request output for all **C*** harmonics:

```
HOUTPUT(C*)=ALL
```

2. Request output for all **S** harmonic coefficients for the harmonics specified in **SET 100**:

```
HOUTPUT(S)=100
```

Case Control Command IC

Selects non zero initial conditions for Direct and Modal Transient Response analyses.

Command Syntax:

```
IC [ ( [ { DSET } , ] [ EQUIL [ , NRB = n ] ] ) ] [ = sid ]
   [ { HSET }       ]
```

Option	*Meaning*	*Data Range*
{ **DSET** } { HSET }	Specifies the displacement set to which the initial conditions belong. [1]	Keyword
EQUIL	Requests the automatic calculation of initial conditions for Direct Transient analysis which will satisfy equilibrium conditions at t=0.0, given non-zero loads existing at t=0.0. [2]	Keyword
n	Requests the automatic calculation of initial conditions for Modal Transient analysis which will satisfy equilibrium conditions at t=0.0, given non-zero loads existing at t=0.0. [3,4]	Integer≥-1
sid	Selects **TIC**, **TICTV**, **TICRV** or **TICS** Bulk Data entries to define initial condition data. This value should be omitted from the command if you are using the **EQUIL** option. [5,6,7]	Integer>0

Remarks:

1. Physical initial conditions belong to the *d-set*. If you are performing either a Direct or Modal Transient response analysis, you may enter physical initial conditions. For these typical situations, you would use the default value of **DSET**.

   However, if you have computed modal coordinate initial conditions with an external procedure, and you are performing a Modal Transient response analysis, you may enter those modal coordinate initial conditions. In this case you must select the value of **HSET**. In this case, the GRID point identification numbers entered on **TIC** Bulk Data entries refer to mode numbers. Use a component code of 0 or 1 for these data.

   A displacement set should not be selected if you have specified the **EQUIL** option.

2. If you have non-zero loads existing at t = 0.0, you are performing a Direct Transient analysis, and you want the model to be placed in static equilibrium at t=0.0, this keyword will request the internal, automatic calculation of the initial conditions. If the structure has free body motions, you must also use a **SUPORT** Bulk Data entry to define these free body motions. In this case, initial velocities will be set to 0.0, and initial rigid body accelerations will be automatically computed.

Mod: V20.1

3. Initial conditions are computed automatically for Modal Transient Response analyses if the initial forces are non-zero and the value $n \geq 0$ denotes the number of rigid body modes. You may also use the Bulk Data entry:

   PARAM,NRB,*n*

   where the value $n \geq 0$ denotes the number of rigid body modes. The use of the **PARAM** Bulk Data entry is provided for compatibility with previous versions of **UAI*/NASTRAN***.

4. Initial conditions for cases with non-zero loads at **t=0.0** may also be computed and saved when performing Static analyses. Refer to the **DISP** Case Control command for additional information.

5. The **TIC**, **TICTV**, **TICRV** and **TICS** Bulk Data are not used unless selected with the **IC** command. **TICS** data are for use with substructuring analysis.

6. In Transient Heat Transfer analysis, non-zero initial temperatures are defined on **TEMP** or **TEMPD** Bulk Data entries, and these entries must be selected with this command.

Examples:

1. You are performing a Direct Transient Response analysis with initial deflections specified using **TIC** Bulk Data entries with an identification number of **101**. Specify these data as initial conditions:

   ```
   IC = 101
   ```

2. You are performing a Direct Transient Response analysis and want the initial conditions to be computed automatically:

   ```
   IC(EQUIL)
   ```

   If the model has free body motion, a **SUPORT** Bulk Data entry is required.

3. You are performing a Modal Transient Response analysis and want the initial conditions to be computed automatically. Additionally, six free body modes exist for the model:

   ```
   IC(EQUIL,NRB=6)
   ```

   This case also requires a **SUPORT** Bulk Data entry to constrain the rigid body modes.

Case Control Command INCLUDE

Merges the contents of another file into the **UAI*/NASTRAN*** input file.

Command Syntax:

```
INCLUDE = ['] phys_name ['][,params]
```

Option	*Meaning*	*Data Range*
phys_name	Specifies the host computer dependent name of the physical file containing data that you wish to have included in your input data stream. [1,2]	[1]
params	Selects host computer dependent parameters.	[3]

Remarks:

1. Physical file name vary depending on your host computer. See Chapter 1 for details.
2. The character string that you specify for *phys_name* is always converted to upper case characters by **UAI*/NASTRAN***. If your host computer has case-sensitive file names, and if you wish to use lower case characters, then you must enclose *phys_name* in single quotation marks, sometimes called tics, as shown in Example 2 below.
3. Other host computer dependent parameters may be available. See Chapter 1.
4. An **INCLUDE** file may not contain any other **INCLUDE** commands.
5. Any number of **INCLUDE** commands may appear anywhere in the Case Control packet.

Examples:

1. Include the file **CASELIB** in your Case Control packet:

```
INCLUDE CASELIB
```

2. Include the contents of the file **/cases/my_case.d** in you Case Control packet:

```
INCLUDE '/cases/my_case.d'
```

Case Control Command K2GG

Selects one or more direct input symmetric stiffness matrices which are *g-set* size.

Command Syntax:

```
K2GG = { name     }
       { sid      }
       { (name_list) }
```

Option	*Meaning*	*Data Range*
name	Selects a single K2GG damping matrix which is defined with **DMIG** or **DMIAX** Bulk Data entries that reference *name*. [1,2,4]	Name
sid	Selects a set that contains a list of one or more K2GG damping matrix names. [3]	Integer
name_list	Explicit list of K2GG stiffness matrix names. [3]	Name

Remarks:

1. **DMIG** and **DMIAX** matrices are not used for stiffness unless selected with the **K2GG** or **K2PP** commands.
2. The matrix *name* must be symmetric in form.
3. All selected matrices are added to form the final K2GG matrix.
4. **K2GG** is available in all Rigid Formats.

Examples:

1. Select a direct input stiffness matrix named **MYK2GG**:

```
K2GG = MYK2GG
```

2. Select three stiffness matrices named **STIF1**, **STIF2**, and **STIF3**:

```
SET 1 = STIF1,STIF2,STIF3
...
K2GG = 1
```

3. Define the three stiffness matrices in Example 2 using an explicit list:

```
K2GG = (STIF1,STIF2,STIF3)
```

4 CASE

Case Control Command K2PP

Selects one or more direct input stiffness matrices which are *p-set* size.

Command Syntax:

```
K2PP = { name      }
       { sid       }
       { ( name_list ) }
```

Option	*Meaning*	*Data Range*
name	Selects a single K2PP damping matrix which is defined with **DMIG** or **DMIAX** Bulk Data entries that reference *name*. [1,2,4]	Name
sid	Selects a set that contains a list of one or more K2PP damping matrix names. [3]	Integer
name_list	Explicit list of K2PP stiffness matrix names. [3]	Name

Remarks:

1. **DMIG** and **DMIAX** matrices are not used for stiffness unless selected with the **K2PP** or **K2GG** commands.
2. The matrix *name* must be symmetric in form.
3. All selected matrices are added to form the final K2PP matrix.
4. **K2PP** matrices may be used in Complex Eigenvalue, Transient Response and Frequency Response analyses.

Examples:

1. Select a direct input stiffness matrix named **MYK2PP**:

```
K2PP = MYK2PP
```

2. Select three stiffness matrices named **STIF1P**, **STIF2P**, and **STIF3P**:

```
SET 1 = STIF1P,STIF2P,STIF3P
...
K2PP = 1
```

3. Define the three stiffness matrices in Example 2 using an explicit list:

```
K2PP = (STIF1P,STIF2P,STIF3P)
```

Case Control Command KFREQGEN

Defines a set of reduced frequencies for which unsteady aerodynamic matrices will be generated.

Command Syntax:

$$\text{KFREQGEN} = \left\{ \begin{matrix} (\ K1,\ K2,\ldots,Ki\ ) \\ kid \end{matrix} \right\}$$

Option	*Meaning*	*Data Range*
K1,K2,...K3	Gives an explicit list of reduced frequencies for which Aerodynamic matrix data is calculated.	Real≥0.0
kid	Selects the identification number of a real **SET** defining the values. [1,2,3]	Integer>0

Remarks

1. To define all Mach number and reduced frequencies you must also use the **MACHGEN** Case Control command.
2. All combinations of Mach number and reduced frequency, *k*, where:

   $$k = \frac{\omega \bar{c}}{2V}$$

   are used to generate Aerodynamic Influence Coefficient matrices for the selected symmetry condition. Multiple **AERO** CASEs are used to specify various symmetry, Mach and reduced frequency combinations.
3. The **KFREQGEN** command is only used if you have specified a **CASE AERO**.

Examples:

1. Define a set with identification number **10** having three reduced frequencies, **0.1**, **0.2** and **0.3**. Select this set as the reduced frequencies used in a Flutter analysis.

```
SET 10 = 0.1,0.2,0.3
KFREQGEN = 10
```

2. Define the same three reduced frequencies explicitly:

```
KFREQGEN = ( 0.1,0.2,0.3 )
```

4 CASE

Case Control Command KLIST

Defines the list of *soft point* reduced frequencies used for performing Flutter analysis with the **K** or **PKSWEEP** methods.

Command Syntax:

```
KLIST = { ( K1,K2,...,K1 ) }
        {      kid         }
```

Option	*Meaning*	*Data Range*
K1,K2,...,Ki	Explicit list of *soft point* reduced frequencies. [1,2]	Real>0.0
kid	Identification number of a real **SET** command containing the list of *soft point* reduced frequencies. [1,2]	Integer>0

Remarks

1. To define all speed conditions and reduced frequencies, you must use the **KLIST** Case Control command to define the 'soft point' reduced frequencies. The aerodynamic matrix data at the *hard points* will be automatically interpolated to the *soft point* frequencies. The interpolation method may be selected with the **FLSOLVE** Bulk Data entry.
2. The **MSELECT** and **KSELECT** Case Control commands are used to specify a subset of the available *hard point* data in the current aerodynamic model for the purpose of limiting the scope of the current analysis or for convergence studies. If these commands are not used, then the flutter analysis is performed using all available reduced frequency data at each available Mach numbers.
3. The **KLIST** command is only used if you have specified a **CASE FLUTTER**. The command is required if **FMETHOD** is **K** or **PKSWEEP**.

Examples:

1. Define a set with identification number **111** having four soft point reduced frequencies, **0.15**, **0.20**, **0.25** and **0.5**. Select this set as the reduced frequencies used in a Flutter analysis.

```
SET 111 = 0.15,0.20,0.25,0.5
KLIST = 111
```

2. Define the same four reduced frequencies explicitly:

```
KLIST = ( 0.15,0.20,0.25,0.5 )
```

Case Control Command KSELECT

Defines a set of available *hard point* reduced frequencies for Flutter analysis.

Command Syntax:

```
KSELECT = { ALL          }
          { ( K1, K2,...,Ki ) }
          { kid          }
```

Option	*Meaning*	*Data Range*
ALL	Selects all available *hard point* reduced frequencies	Keyword
K1,K2,...,Ki	Gives an explicit list of *hard point* reduced frequencies for which Aerodynamic matrix data is calculated. [1]	Real≥0.0
kid	Selects the identification number of a real **SET** command defining the values. [1]	Integer>0

Remarks

1. The **MSELECT** and **KSELECT** Case Control commands may be used, individually or together, to optionally specify a subset of the available *hard point* data in the current aerodynamic model for the purpose of limiting the scope of the current analysis or for convergence studies. If these commands are not used, then the flutter analysis is performed using all available reduced frequency data at each of the available Mach numbers.
2. The **KSELECT** command is only used if you have specified a **CASE FLUTTER**.

Examples:

1. Define a set with identification number **99** having three hard point reduced frequencies, **0.15**, **0.25** and **0.5**. Select this set as the reduced frequencies used in a Flutter analysis.

```
SET 99 = 0.15,0.25,0.5
KSELECT = 99
```

2. Define the same three reduced frequencies explicitly:

```
KSELECT = ( 0.15,0.25,0.5 )
```

4 CASE

Case Control Command LABEL

Defines titling which will appear on the third line of each page of printed output.

Command Syntax:

```
LABEL = any_character_string
```

Remarks:

1. A **LABEL** command appearing at the SUBCASE level will label output for that SUBCASE only.
2. A **LABEL** command which you place before all SUBCASEs will label any outputs which are not SUBCASE dependent.
3. If the **LABEL** command is not used, the label line will be blank.
4. The **LABEL** information is also placed on plots, see Chapters 4 and 5.

Case Control Command LOAD

Selects the external static load set to be applied to the structural model.

Command Syntax:

```
LOAD = lid
```

Option	*Meaning*	*Data Range*
lid	Specifies an external static load Bulk Data entry identification number.	Integer>0

Remarks:

1. Static load data will not be used unless selected with the **LOAD** command.
2. The **GRAV** Bulk Data cannot have the same load set identification number as any of the other load types. If it is desired to apply a gravity load in addition to other static loads, the **LOAD** Bulk Data entry must be used to combine loads. For example the following data combine gravity load set 2 with mechanical force load set 1 to define loading condition set 10:

   Case Control Command:

   ```
   LOAD = 10
   ```

   Bulk Data Entries:

   ```
   LOAD,10,1.0,1.0,1,1.0,2
   GRAV,1,0,386.,0.,0.,-1.
   FORCE,2,12,0,1000.,1.,1.,1.
   ```
3. The **LOAD** command may only be used in Linear and Nonlinear Static, Inertia Relief, Differential Stiffness and Buckling problems.
4. The total load applied will be the sum of external (**LOAD**), thermal (**TEMP(LOAD)**), element deformation (**DEFORM**) and constrained displacement (**SPCD** or **SPCDS**) loads.
5. Static, thermal and element deformation loads must have unique set identification numbers.

Case Control Command LOADCOMBINE

Combines static loads for Basic Substructures during PHASE 2 solution.

Command Syntax:

$$\textbf{LOADCOMBINE} = \left\{ \begin{array}{c} lid \\ \text{ALL} \\ \text{ALLAPP} \end{array} \right\}$$

Option	*Meaning*	*Data Range*
lid	Specifies an external static load Bulk Data entry identification number of the PHASE 1 Basic Substructures. [1]	Integer>0
ALL	Selects all of the PHASE 1 loads and creates PHASE 2 SUBCASE commands automatically. [2]	Keyword
ALLAPP	Selects only the last set of appended loads for the solution.	Keyword

Remarks:

1. This command is equivalent to combining all PHASE 1 loads with the same identification number using **LOADC** Bulk Data entries, but it requires less work. For example, the PHASE 2 Case Control:

```
SUBCASE 1
   LOADCOMB = 10
SUBCASE 2
   LOADCOMB = 20
```

   Results in the same load sets as the input:

```
SUBCASE 1
   LOAD = 10
SUBCASE 2
   LOAD = 20
BEGIN BULK
LOADC,10,1.0,SUBNAM1,10,1.0,SUBNAM2,10,1.0
LOADC,20,1.0,SUBNAM1,20,1.0,SUBNAM2,20,1.0
```

   This removes the need for the Bulk Data.

2. The **ALL** option results in the combination of all sets of loads with the same set identification number. A PHASE 2 SUBCASE is generated for each unique *lid*.

3. A **LOADCOMBINE** may only be used in a Substructuring PHASE 2 run.

Case Control Command M2GG

Selects one or more direct input symmetric mass matrices which are *g-set* size.

Command Syntax:

```
M2GG = { name      }
       { sid       }
       { ( name_list ) }
```

Option	*Meaning*	*Data Range*
name	Selects a single M2GG damping matrix which is defined with **DMIG** or **DMIAX** Bulk Data entries that reference *name*. [1,2,4]	Name
sid	Selects a set that contains a list of one or more M2GG damping matrix names. [3]	Integer
name_list	Explicit list of M2GG mass matrix names. [3]	Name

Remarks:

1. The **DMIG** and **DMIAX** matrices are not used for mass unless selected with the **M2GG** or **M2PP** commands.
2. The matrix *name* must be symmetric in form.
3. All selected matrices are added to form the final M2GG matrix.
4. The **M2GG** command is available in all Rigid Formats.

Examples:

1. Select a direct input mass matrix named **MYM2GG**:

```
M2GG = MYM2GG
```

2. Select three mass matrices named **MASS1**, **MASS2**, and **MASS3**:

```
SET 1 = MASS1,MASS2,MASS3
...
M2GG = 1
```

3. Define the three mass matrices in Example 2 using an explicit list:

```
M2GG = (MASS1,MASS2,MASS3)
```

4 CASE

Case Control Command M2PP

Selects one or more direct input mass matrices which are *p-set* size.

Command Syntax:

```
M2PP = { name      }
       { sid       }
       { ( name_list ) }
```

Option	*Meaning*	*Data Range*
name	Selects a single M2PP damping matrix which is defined with **DMIG** or **DMIAX** Bulk Data entries that reference *name*. [1,2,4]	Name
sid	Selects a set that contains a list of one or more M2PP damping matrix names. [3]	Integer
name_list	Explicit list of M2PP mass matrix names. [3]	Name

4 CASE

Remarks:

1. The **DMIG** and **DMIAX** matrices are not used for mass unless selected with the **M2PP** or **M2GG** commands.
2. The matrix *name* must be symmetric in form.
3. All selected matrices are added to form the final M2PP matrix.
4. The **M2PP** matrices may be used in Complex Eigenvalue, Transient Response and Frequency Response analyses.

Examples:

1. Select a direct input mass matrix named **MYM2PP**:

```
M2PP = MYM2PP
```

2. Select three mass matrices named **MASS1P**, **MASS2P**, and **MASS3P**:

```
SET 1 = MASS1P,MASS2P,MASS3P
...
M2PP = 1
```

3. Define the three mass matrices in Example 2 using an explicit list:

```
M2PP = (MASS1P,MASS2P,MASS3P)
```

Rev: V20.1

Case Control Command M2PP

Selects a direct input mass matrix which is *p-set* size.

Command Syntax:

M2PP = *name*

Option	*Meaning*	*Data Range*
name	Selects an M2PP mass matrix which is defined with **DMIG** or **DMIAX** Bulk Data entries that reference *name*. [1]	Character

Remarks:

1. The **DMIG** and **DMIAX** matrices are not used for mass unless selected with the **M2PP** or **M2GG** commands.
2. The **M2PP** matrices may be used in Complex Eigenvalue, Transient Response and Frequency Response analyses.

Example:

1. Select a direct input mass matrix named **MYM2PP**:

```
M2PP = MYM2PP
```

Case Control Command MAXRETAIN

Specifies the maximum number of design constraints to be retained during Multidisciplinary Design Optimization.

Command Syntax:

```
MAXRETAIN = numg
```

Option	*Meaning*	*Data Range*
numg	Specifies the maximum number of design constraints to be retained during the optimization procedure. [1]	Integer>0

Remarks:

1. The default value is **100**.
2. The **MAXRETAIN** command may only be used when performing Multidisciplinary Design Optimization.

Case Control Command MACHGEN

Defines a set of Mach numbers for which unsteady aerodynamic matrices will be generated.

Command Syntax:

```
MACHGEN = { ( M1, M2,...,Mi ) }
          {       mid       }
```

Option	*Meaning*	*Data Range*
M1,M2,...,Mi	Gives an explicit list of Mach numbers for which Aerodynamic matrix data is calculated. [1,2]	Real≥0.0
mid	Specifies an explicit list of Mach numbers for which Aerodynamic matrix data is calculated, or references a real set, *mid*, defining the values. [1,2]	Integer>0

Remarks

1. To define all Mach number and reduced frequencies you must also use the **KFREQGEN** Case Control command.
2. All combinations of Mach number and reduced frequency, *k*, where:

   $$k = \frac{\omega \bar{c}}{2V}$$

   will be used to generate Aerodynamic Influence Coefficient matrices for the selected symmetry condition. Multiple **AERO** CASEs are used to specify various symmetry, Mach and reduced frequency combinations.
3. The **MACHGEN** command is only used if you have specified a **CASE AERO**.

Examples:

1. Define a set with identification number **10** having two Mach numbers, **0.2** and **0.6**. Select this set as the Mach numbers used in a Flutter analysis.

```
SET 10 = 0.2, 0.6
MACHGEN = 10
```

2. Define the same two Mach numbers explicitly:

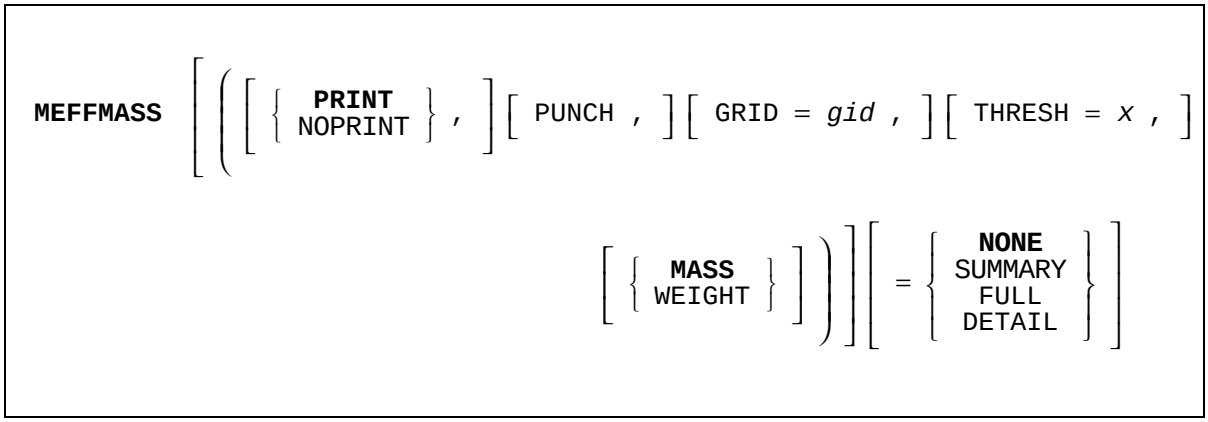
```
MACHGEN = ( 0.2,0.6 )
```

4 CASE

Case Control Command MEFFMASS

Requests calculation of modal effective mass.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
gid	Specifies a GRID point identification number that defines the rigid body mode origin.	Integer>0 Default=0.0
x	Only modes with contributions greater than *x* will be printed.	Real>0.0
{ **MASS** / WEIGHT }	Selects either **MASS** or **WEIGHT** output units. [2]	Keyword
{ **NONE** / SUMMARY / FULL / DETAIL }	Requests printing of Modal Effective Mass outpur reports. **NONE** disables printing. [3]	Keyword

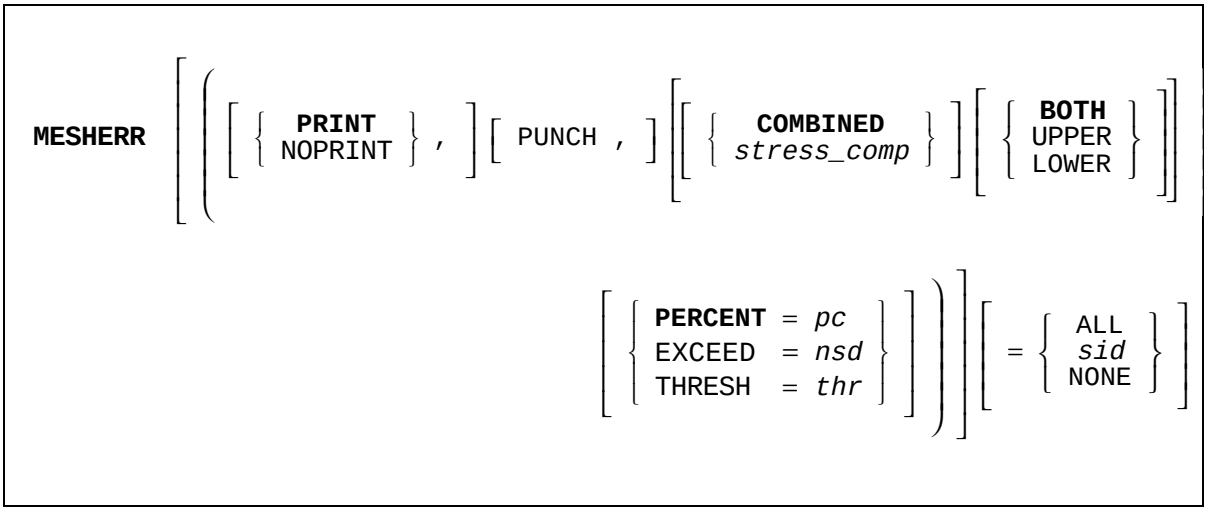
Remarks

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. If **MASS** is selected, then the Bulk Data entry **PARAM,WTMASS** must appear in the Bulk Data packet.
3. See Chapter 11 of the ***User's Guide*** for detailed descriptions of these print options and examples of their results.

Case Control Command MESHERR

Requests calculation of mesh error estimates for solid or plate and shell elements.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ COMBINED / *stress_comp* }	Selects the stress component to be used for computing the mesh error. Slected from the **COMBINED** rule or a single *stress_comp*. [2].	Keyword
{ **BOTH** / UPPER / LOWER }	Selects either the **UPPER** or **LOWER** fiber stress components, or **BOTH** components, for plate elements. [2]	Keyword
pc	Specifies that the output will consist of elements with estimated error percentages in the *pc*th percentile of all estimated error percentage values. [3]	0.0<Real<100.0
nsd	Specifies that the output will consist of elements with estimated error percentages exceeding *nsd* multiples of the standard deviation of the estimated errors. [3]	Real>0.0
thr	Specifies that the output will consist of elements with estimated error percentages above the *thr* value. [3]	Real>0.0
{ ALL / *sid* / NONE }	Enables results generation for **ALL** or a set of eligible elements, or disables the request. [4]	Keyword

4 CASE

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. The **COMBINED** rule combines all of the stress components from all points in an element as described in the ***User's Guide***. You may also select individual stress components and, for plate elements, the upper or lower fiber. The allowable *stress_comp* values are:

FOR ELEMENTS:	ALLOWABLE *stress_comp* ARE:
QUAD4, QUAD8, QUADR, TRIA3, TRIA6and TRIAR	SIGX,SIGY,TAUXY,VONM
HEXA, PENTA, and TETRA	SIGX,SIGY,SIGZ TAUXY,TAUYZ,TAUZX VONM

   You must take care in interpreting mesh error results when using only single stress components.
3. The **PERCENT**, **EXCEED**, and **THRESH** options may not be used with **SORT2** output requests.
4. The eligible elements consist of solid elements HEXA, PENTA, and TETRA, and plate and shell elements QUAD4, QUAD8, QUADR, TRIA3, and TRIA6, and TRIAR.
5. Mesh error estimates are available only when performing static or normal modes analysis, including the use of these disciplines as part of Multidisciplinary Design Optimization.

Examples:

1. Print the mesh error estimates in the 90th percentile:

```
MESHERR (PERCENTILE=90.0) = ALL
```

2. Punch the mesh error estimates which exceed two standard deviations of all the estimated errors:

```
MESHERR (NOPRINT,PUNCH,EXCEED=2.0) = ALL
```

3. Request the mesh error computations using only the normal-y stresses in the upper fiber of your model:

```
MESHERR(PRINT,SIGY,UPPER) = ALL
```

Case Control Command METHOD

Selects real eigenvalue extraction method and control parameters.

Command Syntax:

```
METHOD = sid
```

Option	*Meaning*	*Data Range*
sid	Selects the set identification number of eigenvalue extraction data. [1]	Integer>0

Remarks:

1. The **METHOD** command must be used with Rigid Formats which extract real eigenvalues. These include Normal Modes analysis, Dynamic Response analyses which use the modal approach, and Buckling analysis . The set identification number *sid* refers to an **EIGR** Bulk Data entry in the first two cases, and to an **EIGB** Bulk Data entry when performing Buckling analysis.

Case Control Command MODESEXCLUDE

Excludes specific modes from the set of generalized coordinates used when performing dynamic and flutter analyses using the modal method.

Command Syntax:

```
MODESEXCLUDE = mlid
```

Option	*Meaning*	*Data Range*
mlid	Selects the identification number of an integer **SET** command containing the list of mode identification numbers to be excluded. [1]	Integer>0

Remarks:

1. If you do not use this command, then all modes are used as generalized coordinates.
2. The **MODESEXCLUDE** command is only used if you have specified a **CASE FLUTTER**.

Examples:

1. Perform a modal frequency response analysis using the modes created in a previous case. Exclude modes 1,3,9,14 and 15.

```
CASE 101 MODES
   METHOD = 100
   ...
CASE 201 FLUTTER
   SET 101 = 1,3,9,14,15
   ...
   USING MODES 101
   ...
   MODESEXCLUDE 101
   ...
```

Case Control Command MPC

Selects the multipoint constraint set to be applied to the structural model.

Command Syntax:

```
MPC = sid
```

Option	*Meaning*	*Data Range*
sid	Selects a multipoint constraint set identification number. [1]	Integer>0

Remarks:

1. The identification number *sid* refers to **MPC, MPCS** and **MPCADD** Bulk Data entries. These data will not be used unless selected with this command.
2. Multipoint constraints are created internally with the rigid elements **RBAR**, **RBE1**, **RBE2**, **RBE3**, **RROD**, **RSPLINE** and **RTRPLT**. An **MPC** request is not necessary to activate rigid elements.

Case Control Command MPCFORCES

Requests multipoint forces of constraint results.

Command Syntax:

MPCFORCES ([{ **PRINT** / NOPRINT } ,] [PUNCH]) = { ALL / *sid* / NONE }

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. Multipoint forces of constraint include the forces associated with rigid elements.
4. The **MPCFORCES** output is available for Statics, Statics with Inertia Relief, and Normal Modes analyses.

Examples:

1. Print the multipoint constraint forces for the points defined in **SET 10**:

```
MPCFORCES = 10
```

2. Print the multipoint constraint forces for all points in **SET 101** and write the results to the **PUNCH** file:

```
MPCF(PUNCH) = 101
```

Case Control Command MPFACTOR

Requests the output of modal participation factors corresponding to shock application points, when performing Shock Spectra Response analyses.

Command Syntax:

```
MPFACTOR ( [ { PRINT } , ] [ ,PUNCH ] ) = { ALL }
         ( [ { NOPRINT } ]                { sid }
                                          { NONE }
```

Option	*Meaning*	*Data Range*
{ **PRINT** } { NOPRINT }	Specifies that results be generated and then written to the print file, or retained for post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL } { NONE }	Specifies that **ALL** modal participation factors will be output, or overrides a previous request appearing above the SUBCASE level.	Keyword

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. The **MPFACTOR** command may only be used when you are performing a Shock Spectra Response analysis using Rigid Format 17.

Case Control Command MSELECT

Defines a set of *hard point* Mach numbers for Flutter analysis.

Command Syntax:

```
MSELECT = { ALL          }
          { ( M1, M2,...,Mi ) }
          { kid          }
```

Option	*Meaning*	*Data Range*
ALL	Selects all available *hard point* Mach numbers.	Keyword
M1,M2,...,Mi	Gives an explicit list of *hard point* Mach numbers for which Aerodynamic matrix data are calculated. [1]	Real≥0.0
mid	Selects the identification number of a real **SET** command defining the values. [1]	Integer>0

4 CASE

Remarks

1. The **MSELECT** and **KSELECT** commands may be used, individually or together, to specify a subset of the available *hard point* data in the current aerodynamic model, for the purpose of limiting the scope of the current analysis or for convergence studies. The default action is to perform the flutter analysis using all available reduced frequency data at each available Mach number.
2. The **MSELECT** command is only used if you have specified a **CASE FLUTTER**.

Examples:

1. Define a set with identification number **75** having three hard point Mach numbers, **0.45**, **0.55** and **0.65**. Select this set as the Mach numbers used in a Flutter analysis.

```
SET 75 = 0.45,0.55,0.65
MSELECT = 75
```

2. Define the same three reduced frequencies explicitly:

```
MSELECT = ( 0.45,0.55,0.65 )
```

Case Control Command NLFORCE

Requests nonlinear element force results.

Command Syntax:

```
NLFORCE [ ( [ { PRINT } , ] [ PUNCH ] ) ] = { ALL }
        [     { NOPRINT }               ]   { sid }
                                            { NONE }
```

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** nonlinear elements, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the elements at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **NLFORCE** output are only available in Nonlinear Static analysis, i.e.

```
APP NONLIN
SOL 1
```

Examples:

1. Print the forces for the nonlinear elements defined in **SET 101**:

```
NLFORCE = 101
```

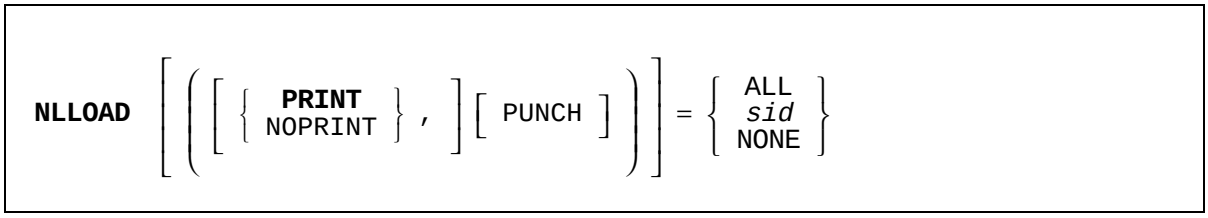
2. Print the forces for nonlinear elements in **SET 2** in **SORT2** format and write the results to the **PUNCH** file:

```
SORT = SORT2
...
NLFORCE(PRINT,PUNCH) = 2
```

Case Control Command NLLOAD

Requests output of applied nonlinear loads for Transient Response analysis.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2,3,4]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. Results are only available in **SORT2** order.
3. The **NLLOAD** output are available for Transient Response analyses. The output are available only in the solution set (*d-set* or *h-set*).
4. The nonlinear loads are generated by using **NOLINi** Bulk Data entries and the Case Control command **NONLINEAR**.

Examples:

1. Print the nonlinear loads for all of the points in the model:

```
NLLOAD = ALL
```

2. Print the nonlinear loads defined by **SET 27** and also write them on the **PUNCH** file:

```
NLLOAD(PRINT,PUNCH) = 27
```

Case Control Command NLPRINT

Requests summary or detailed print of the nonlinear solution procedure in Nonlinear Static analysis.

Command Syntax

```
NLPRINT = { DETAIL }
          { SUMMARY }
          { NONE    }
```

Option	*Meaning*	*Data Range*
{ **DETAIL** / SUMMARY / NONE }	Requests printing of either a **DETAIL**ed history of unbalanced force iteration data and stiffness matrix update conditions for each SUBCASE, STEP and load INCREMENT, or a brief **SUMMARY** of the nonlinear solution control parameters and a description of the conditions of solution convergence or divergence. **NONE** disables printing. [1]	Keyword

Remarks:

1. If this command is not used, the **DETAIL**ed report is printed.
2. This command may be used either above or below the STEP level.
3. The **NLPRINT** command may only be used in Nonlinear Static analysis, i.e.

```
APP NONLIN
SOL 1
```

Case Control Command NLREDUCE

Requests the automatic selection of degrees of freedom for the nonlinear set.

Command Syntax:

```
NLREDUCE [ = { YES } ]
         [   { NO  } ]
```

Option	*Meaning*	*Data Range*
{ YES } { NO }	Selects or deselects the **NLREDUCE** option	Keyword

Remarks:

1. The **NLREDUCE** command may be used either above the SUBCASE level or may be changed, one SUBCASE to the next. It may not be used at the STEP level.

4 CASE

New: V20.1

Case Control Command NLRESTART

Initiates a RESTART execution for a Nonlinear Static analysis at a specified point in the solution procedure.

Command Syntax:

NLRESTART [SUBCASE *i* [, STEP *j* [, INCREMENT *k*]]]

Option	*Meaning*	*Data Range*
i	Specifies the identification number of a previously executed SUBCASE.	0<Integer<99 Default=1
j	Specifies the identification number of a previously executed STEP.	0<Integer<99 Default=1
k	Specifies the identification number of a previously executed INCREMENT.	0<Integer<99 Default=1

Remarks:

1. The **NLRESTART** command must appear before any **SUBCASE** command.
2. If not specified, a restart begins from the last INCREMENT of the previously executed analysis.
3. Your Case Control packet must contain all of the commands used in the original execution up to the point where you request the restart to begin.
4. All data contained on the NLDB database for the RESTART point and all following steps will be deleted when the RESTART begins.
5. The **NLRESTART** command may only be used in Nonlinear Static analysis, i.e.

```
APP NONLIN
SOL 1
```

Case Control Command NLSAVE

Requests that nonlinear output be saved on the NLDB and be printed at the load INCREMENT level. Without this command, output is saved and printed at each load STEP level. Also requests that data be saved for a subsequent material or geometric nonlinear analysis.

Command Syntax:

NLSAVE $\left[\left(\text{[MAT,][GEOM]}\right)\right]$ EVERY *n* INCREMENT

Option	*Meaning*	*Data Range*
MAT	Requests saving data for a material nonlinear restart.	Keyword
GEOM	Requests saving data for a geometric nonlinear restart.	Keyword
n	Requests that every n^{th} load INCREMENT be saved and output generated.	Integer>0

Remarks:

1. This command must be used above the STEP level.
2. The **NLSAVE** command may only be used in Nonlinear Static analysis, i.e.

```
APP NONLIN
SOL 1
```

Case Control Command NLSOLVE

Selects the nonlinear solution control data.

Command Syntax:

```
NLSOLVE = nlsid
```

Option	*Meaning*	*Data Range*
nlsid	Selects an **NLSOLVE** Bulk Data entry identification number. [1]	Integer>0

Remarks:

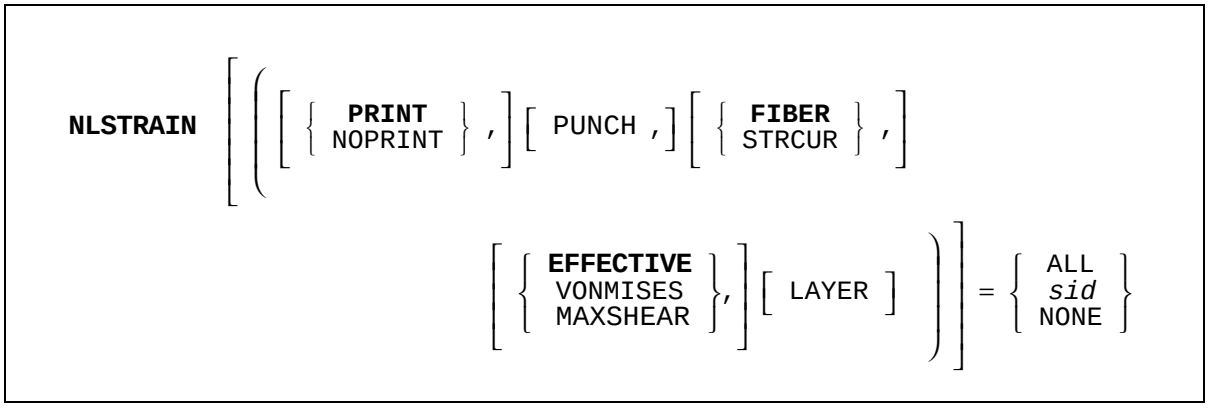
1. The **NLSOLVE** request may be made above or below the SUBCASE and STEP levels in the Case Control command packet.
2. The **NLSOLVE** command may only be used in Nonlinear Static analysis, i.e.

```
APP NONLIN
SOL 1
```

Case Control Command NLSTRAIN

Requests nonlinear element strain results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **FIBER** / STRCUR }	Requests strains to be calculated at the extreme (top and bottom) fibers of a plate element or as strains and curvatures at the midsurface of the element.	Keyword
{ **EFFECTIVE** / VONMISES / MAXSHEAR }	Requests the calculation of effective strain, Hencky-von Mises equivalent strains, or maximum shear strains for plate elements or octahedral shear strains for solid elements.	Keyword
LAYER	Requests that for composite elements individual layer stresses and failure indices be calculated. [2]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request.	Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. **LAYER** stresses are only available in Geometric Nonlinear Analyses, **NLTYPE=GEOM**.
3. The **NLSTRAIN** output are only available in Nonlinear Static analysis, i.e.

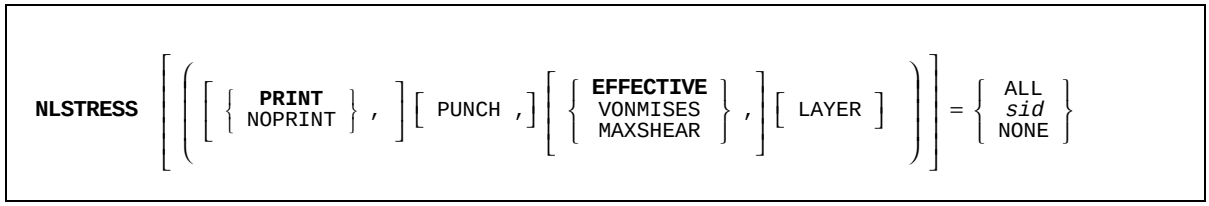
```
APP NONLIN
SOL 1
```

Rev: V20.1

Case Control Command NLSTRESS

Requests nonlinear element stress results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **EFFECTIVE** / VONMISES / MAXSHEAR }	Requests the calculation of effective stress, Hencky-von Mises equivalent stress, or maximum shear stress for plate elements or octahedral shear stress for solid elements.	Keyword
LAYER	Requests that for composite elements individual layer stresses and failure indices be calculated. [2]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request.	Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. **LAYER** stresses are only available in Geometric Nonlinear Analyses, **NLTYPE=GEOM**.
3. The **NLSTRESS** output are only available in Nonlinear Static analysis, i.e.

```
APP NONLIN
SOL 1
```

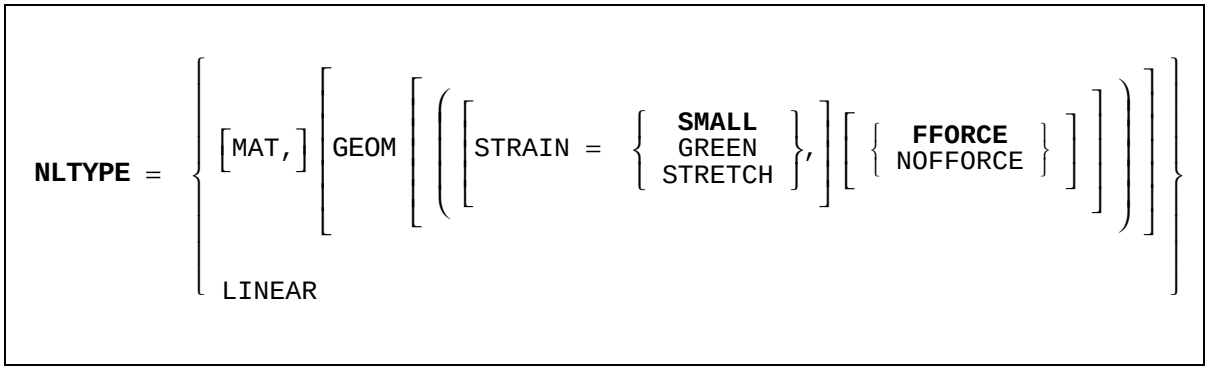
4 CASE

Rev: V20.1

Case Control Command NLTYPE

Specifies the type of nonlinear analysis to be performed.

Command Syntax:



Option	*Meaning*	*Data Range*
MAT	Selects a material nonlinear analysis. [1]	Keyword
GEOM	Selects a geometric nonlinear analysis.	Keyword
LINEAR	Selects a linear analysis. [2]	Keyword
{ **SMALL** GREEN STRETCH }	Selects a strain formulation for geometric nonlinear analysis. [3]	Keyword
{ **FFORCE** NOFFORCE }	Selects the computation of follower forces and load stiffnesses for geometric nonlinear analysis. [4]	Keyword

Remarks:

1. The **NLTYPE** command is only used when performing nonlinear static analyses, i.e.

   ```
   APP NONLIN
   SOL STATICS
   ```

   If the **NLTYPE** command is not used, then **NLTYPE=MAT** by default. **NLTYPE** is placed within, and may vary by, SUBCASE, but the command must appear above the STEP level.

2. Using **NLTYPE=LINEAR** requests that a linear analysis be performed for all SUBCASEs and STEPs in the Case Control packet. If this option is selected, **NLTYPE** must appear above all **SUBCASE** commands.

3. In geometric nonlinear analysis, all strains are measured with respect to the displaced coordinate system. **SMALL** strain specifies that while rotations and displacements may have large absolute values, the rotation and stretch are small relative to the displaced coordinate system. **GREEN** strain allows the rotations to be large in the displaced system, but the stretching must be small. Finally, **STRETCH** allows the stretching to be large but the rotations must be small.

4. If you select the Follower Force option, **FFORCE**, then all loads defined by **FORCE1**, **FORCE2**, **MOMENT1**, **MOMENT2**, **PLOAD1** and **PLOAD4** Bulk Data entries will be treated as follower forces. Load stiffness is available only with **PLOAD4** entries.

Examples:

1. Request a geometric nonlinear analysis with small strain, follower forces and load stiffness:

```
NLTYPE = GEOM
```

2. Request a combined material and geometric nonlinear analysis with Green strains and no follower forces:

```
NLTYPE = MAT,GEOM(STRAIN=GREEN,NOFFORCE)
```

3. Request a material nonlinear analysis in **SUBCASE 1** and a geometric nonlinear analysis in **SUBCASE 2**:

```
SUBCASE 1
   NLTYPE = MAT
   STEP 1
      ...
      ...
   STEP 2
      ...
      ...
SUBCASE 2
   NLTYPE = GEOM
   STEP 1
      ...
      ...
BEGIN BULK
...
```

Case Control Command NONLINEAR

Selects nonlinear loads for Transient Response analysis.

Command Syntax:

```
NONLINEAR = nllid
```

Option	*Meaning*	*Data Range*
nllid	Selects a of nonlinear load set identification number. [1]	Integer>0

Remarks:

1. The identification number *nllid* references **NOLINi** Bulk Data. These data will not be used unless selected with this command.

Case Control Command NOUTPUT

Requests solution results for segments in Cyclic Symmetry analysis.

Command Syntax:

$$\textbf{NOUTPUT}\left[\left(\left\{\begin{matrix}\text{R}\\\text{L}\end{matrix}\right\}\right)\right]=\left\{\begin{matrix}\text{ALL}\\\textit{sid}\end{matrix}\right\}$$

Option	*Meaning*	*Data Range*
$\left\{\begin{matrix}\text{R}\\\text{L}\end{matrix}\right\}$	Restricts output to either right or left half-segments. [1]	Keyword
$\left\{\begin{matrix}\text{ALL}\\\textit{sid}\end{matrix}\right\}$	Enables results generation for **ALL** segments, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the segments at which output will be generated.	Integer>0

Remarks:

1. If neither **R** nor **L** are specified, then output for both half-segments is given.

Examples:

1. Request output for all right and left half-segments:

```
NOUTPUT = ALL
```

2. Request output for the right-half segments specified in **SET 101**:

```
NOUTPUT(R)=101
```

Case Control Command NROOT

Selects the number of roots to be computed in a Flutter analysis using the **PKITER** or **PKSWEEP** methods, or the number of roots to output when using the **K** method.

Command Syntax:

$$\textbf{NROOT} = \left\{ \begin{matrix} \textbf{ALL} \\ n \end{matrix} \right\}$$

Option	*Meaning*	*Data Range*
ALL	Specifies that **ALL** roots will be found when performing Flutter analysis using either the **PKITER** or **PKSWEEP** method, or that **ALL** roots will be output when using the **K** method. [1,2]	Keyword
n	Specifies that the first *n* roots will be found when performing Flutter analysis using either the **PKITER** or **PKSWEEP** method, or that *n* roots will be output when using the **K** method. [1,2]	Integer>0

Remarks:

1. The Flutter analysis method is selected with the **FMETHOD** Case Control command.
2. The **NROOT** command may request a limited set of output from the **K** method when only a subset of the results are desired. For the PK methods, the number of calculations will be reduced since these methods will only iterate to convergence for the first *n* modes.
3. The **NROOT** command is only used if you have specified a **CASE FLUTTER**.

Case Control Command OBJECTIVE

Defines the objective function for Multidisciplinary Design Optimization and selects optimization parameters.

Command Syntax:

```
OBJECTIVE { MINIMIZE } { WEIGHT }
          { MAXIMIZE } {  MASS  }
                       { VOLUME }
                       {  conid }

      MAXITER     = niter
      MINACTIVE   = gactive
      MAXFEASIBLE = gfeas
      CONVERGE    = eps
```

Option	*Meaning*	*Data Range*
{ MINIMIZE } { MAXIMIZE }	Specifies whether the objective function will be **MINIMIZE**d or **MAXIMIZE**d.	Keyword
{ WEIGHT } { VOLUME }	Selects the structural **WEIGHT** or **VOLUME** as the objective function of the optimization.	Keyword
conid	Specifies the identification number of a single response constraint to be used as the objective function	Integer>0
niter	Maximum number of design iterations to be performed. [3]	Integer>0 Default=10
gactive	Value which defines an active constraint. [4]	Real Default=-0.001
gfeas	Value which defines a violated constraint. [5]	Real Default=0.001
eps	Specifies the maximum per cent change in the design which indicates convergence. [6]	Real>0.0 Default=1.0

Remarks:

1. For constraint satisfaction problems, you omit the **OBJECTIVE** command from the Case Control packet.
2. The **OBJECTIVE** Command may only be used when performing Multidisciplinary Design Optimization, i.e.

   ```
   APP DISP
   SOL MULTI
   ```

3. For most MDO problems the default number of iterations is sufficient, but for flat design spaces and large numbers of design variables (i.e. more than 100) more iterations may be required.
4. The **MINACTIVE** option defines the maximum value of a constraint that is placed in the inactive set $G_{inactive}$ defined by:

   $$G_{inactive} = \{ g \in G \mid g < gactive \}$$

Constraints that are inactive are not considered during the solution of the approximate optimization problem.

5. The **MAXFEASIBLE** option, along with **MINACTIVE** option, defines the sets of violated, $G_{violated}$ and active, G_{active} constraints. These sets are defined by:

$$G_{violated} = \left\{ g \in G \mid g \geq \textbf{MAXFEASIBLE} \right\}$$

$$G_{active} = \left\{ g \in G \mid \textbf{MAXFEASIBLE} > g \geq \textbf{MINACTIVE} \right\}$$

Both violated and active constraints are retained during the approximate optimization process. For constrained MDO problems, there are no violated constraints at the optimum, and at least one constraint is active.

6. **CONVERGE** specifies the convergence criteria for the ***Approximate Optimization Problem***. Two criteria are used to determine if convergence has been achieved between the current approximation and the full optimization problem. Both criteria attempt to determine if a design point was found that is different from the one about which the Approximate Problem was formed. The first is a limit on the percent change in the objective function, and the second is a limit on the length of the $\delta \boldsymbol{X}$ vector as a percentage of the length of $\boldsymbol{X}$., which is the design vector. These are each described below.

Relative Change in the Objective. The first check determines if the final prediction of the *approximate* objective function value is unchanged relative to the initial (*exact*) function value:

$$\delta \boldsymbol{X} = \left| \bar{F}(\boldsymbol{X}^n) - F(\boldsymbol{X}^{n-1}) \right| \leq \left| \varepsilon_O \cdot F(\boldsymbol{X}^{n-1}) \right|$$

Relative Design Variable Move. The second check determines the change in the final design point relative to the point about which the approximate problem was built. This is computed from the Euclidean Norm of the *change in the design point* and is normalized by the length of the design vector $\boldsymbol{X_0}$ to account for the scaling of the mathematical design variables. The ratio of the two Euclidean Norms must satisfy:

$$\frac{|| \delta \boldsymbol{X} ||}{|| \boldsymbol{X} ||} \leq \varepsilon_X$$

Both of these conditions must be met before the Approximate Problem is considered to have converged. The values ε_O and ε_X are computed from the value given by the *eps* value of the **CONVERGE** subcommand as:

$$\varepsilon_O = \frac{eps}{100.0}$$

$$\varepsilon_X = \frac{\mathbf{3} \cdot eps}{100.0}$$

7. See Chapter 25 of the ***User's Guide*** for a more detailed description of MDO capability.

Case Control Command OFREQUENCY

Requests Frequency Response analysis results at a specified set of frequencies.

Command Syntax:

OFREQUENCY = { ALL / *sid* }

Option	*Meaning*	*Data Range*
{ ALL / *sid* }	Requests output for **ALL** frequencies or provides the identification number *sid* of a previously appearing Real **SET** Case Control command. Output for frequencies closest to those defined in this set will be output.	Integer>0

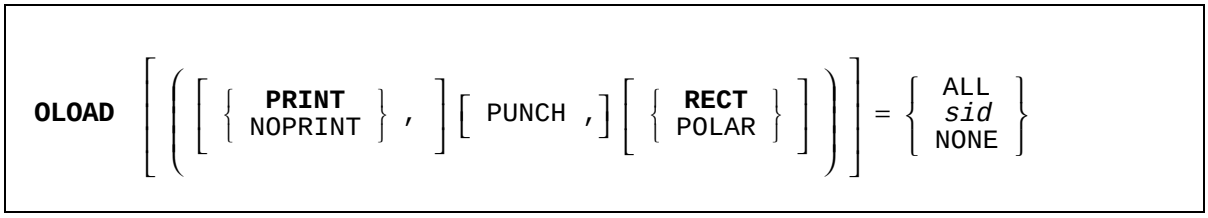
Remarks:

1. If the **OFREQUENCY** command is not used, output is generated for all frequencies.

Case Control Command OLOAD

Requests applied load vector output.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response analyses.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **OLOAD** output are available for all analyses except eigenvalue analyses.

Example:

1. Print the applied load vector for the points specified in **SET 101**:

```
OLOAD = 101
```

Case Control Command OMODES

Requests eigenvector results for specified modes.

Command Syntax:

OMODES = { ALL / *sid* }

Option	*Meaning*	*Data Range*
{ ALL / *sid* }	Requests output for **ALL** eigenvectors or provides the identification number *sid* of a previously appearing Integer **SET** Case Control command.	Keyword or Integer>0

Remarks:

1. If the **OMODES** command is not used, output is printed for all computed eigenvectors.
2. The **OMODES** command does not effect the number of modes that are computed, but rather simply those computed eigenvectors that will be output. The actual number of modes computed is controlled by the **EIGR** Bulk Data entry.

Case Control Command **OTIME**

Requests Transient Response analysis results at a specified set of times.

Command Syntax:

OTIME = { ALL / *sid* }

Option	*Meaning*	*Data Range*
{ ALL / *sid* }	Requests output for **ALL** time steps or provides the identification number *sid* of a previously appearing Real **SET** Case Control command. Output for times closest to those defined in this set will be output.	Keyword or Integer>0

Remarks:

1. If the **OTIME** command is not used, output is printed for all times.
2. The **OTIME** command is particularly useful with restarts to request a subset of the output such as stresses at only peak times.

Case Control Command OUTPUT

Delimits the Structural Plotter and the X-Y Plotter Subpackets.

Command Syntax:

OUTPUT ({ PLOT / XYPLOT })

Option	*Meaning*	*Data Range*
{ PLOT / XYPLOT }	Defines the beginning of the structural **PLOT** subpacket or the **XYPLOT** subpacket. These commands must precede all plotter control commands of the specified type.	Keyword

Remarks:

1. The structural plotter subpacket and the X-Y plotter subpacket must be at the end of the Case Control command packet. See Chapters 5 and 6 of this manual for more information.

Case Control Command PLOTID

Defines titling information which will appear on the first frame of any Structural Plotter output.

Command Syntax:

PLOTID = *any_character_string*

Remarks:

1. The **PLOTID** command must appear before the command:

   OUTPUT(PLOT)

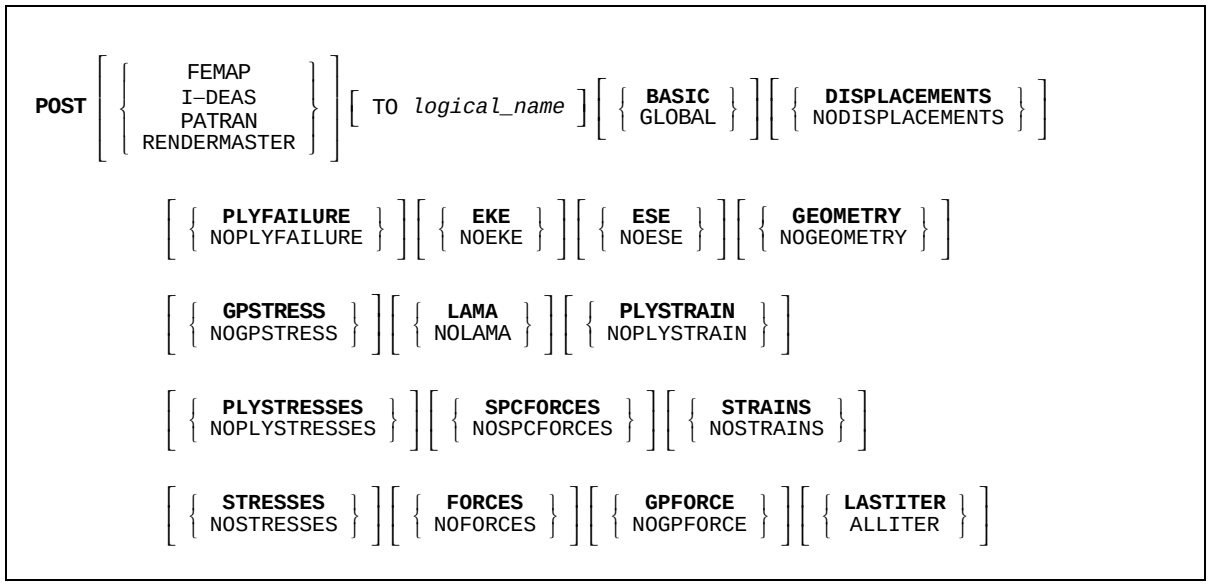
2. When you use the **PLOTID** command, a header frame containing the specified character data is plotted. No header is plotted otherwise.

3. The **PLOTID** header frame is not generated for the X-Y Plotter.

Case Control Command POST

Controls selection of data for commercial post-processors including ESP's FEMAP, MSC/Patran, SDRC's I-DEAS, and **UAI**/***RenderMaster***.

Command Syntax:



Option	*Meaning*	*Data Range*
{ FEMAP / I–DEAS / PATRAN / RENDERMASTER }	Selects the commercial post-processor product.	Keyword
logical_name	Defines a logical file name. [1,2,3]	Character
{ **BASIC** / GLOBAL }	Request that selected output quantities be in the **BASIC** or **GLOBAL** coordinate system.	Keyword
{ **EKE** / NOEKE }	Selects or deselects specific output quantities. [4]	Keyword
{ **LASTITER** / ALLITER }	Requests post-processing output for the **LAST ITER**ation of Multidisciplinary Design Optimization (MDO), or output for **ALL ITER**ations.	Keyword

Remarks:

1. Unless a default is provided in your Configuration, you must use the **ASSIGN** Executive Control command to define a logical file with **USE=OUTPUT2** and a logical file name of *logical_name*.

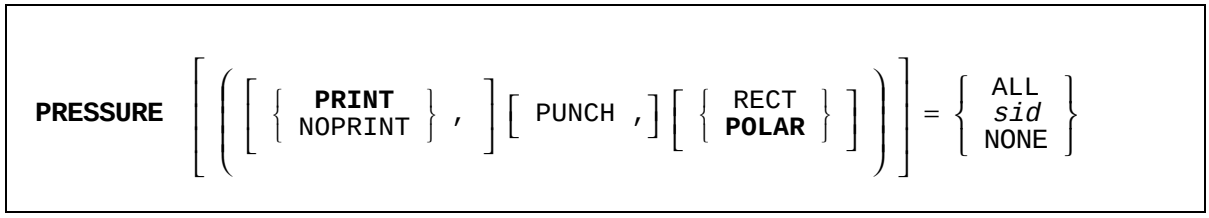
2. The default logical file name is **OUTPUT2**. This can be changed in your Configuration.
3. When you use the Differential Stiffness or Buckling disciplines, the **POST** command must appear at the subcase level. If **POST** is requested in both subcases, the data should be directed to two different logical files.
4. The following table describes the output selection options:

OPTION	DESCRIPTION
DISPLACEMENT	GRID point displacement.
EKE	Element kinetic energy.
ESE	Element strain energy.
FORCES	Element forces.
GEOMETRY	Model geometric data.
GPFORCE	GRID point forces.
GPSTRAIN	GRID point strains.
GPSTRESS	GRID point stresses.
LAMA	Eigenvalue data.
PLYSTRAIN	Composite lamina strains.
PLYSTRESS	Composite lamina stresses.
SPCFORCE	Forces of single-point constraint.
STRAIN	Elements strains.
STRESS	Element stresses.

Case Control Command PRESSURE

Requests pressure results for Hydroelastic, Acoustic, or Aerodynamic analyses.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing. [1,2]	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1,2,3]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response analyses. [1,2]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [4,5]	Keyword or Integer>0

Remarks:

1. These options are not available when performing Aerodynamic analyses.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
4. When performing Hydroelastic or Acoutic analyses, **PRESSURE** output is only available for Normal Modes, Transient Response and Frequency Response analyses.
5. You may not specify an *sid* when performing Aerodynamic analyses. The **ALL** option prints the pressures for all aerodynamic elements at all reduced frequencies and Mach numbers.

Example:

1. Print the pressure, in Polar format, for the points specified in **SET 12**:

```
PRESS(POLAR) = 12
```

2. Print the pressure in **SET 100** in **SORT2** order using **RECT**angular format and also write the results to the **PUNCH** file:

```
SORT = SORT2
...
PRESS(RECT,PUNCH) = 100
```

4 CASE

Case Control Command RANDOM

Selects power spectral density and autocorrelation functions for Random analysis.

Command Syntax:

```
RANDOM = sid
```

Option	*Meaning*	*Data Range*
sid	Selects a power spectral density and autocorrelation function set identification number. [1]	Integer>0

Remarks:

1. The identification number *sid* references power spectral density and autocorrelation time lag constants which are defined with **RANDPS** and **RANDT1** Bulk Data, respectively. These data will not be used unless selected with the **RANDOM** command.
2. The **RANDOM** command must be selected in the first SUBCASE of a group representing a new boundary condition, excitation frequencies, or direct input matrices selected with **K2PP**, **M2PP**, and **B2PP** Case Control commands.

Case Control Command REPCASE

Delimits and identifies a repeated SUBCASE.

Command Syntax:

```
REPCASE subid
```

Option	*Meaning*	*Data Range*
subid	Specifies a SUBCASE identification number.	Integer>1

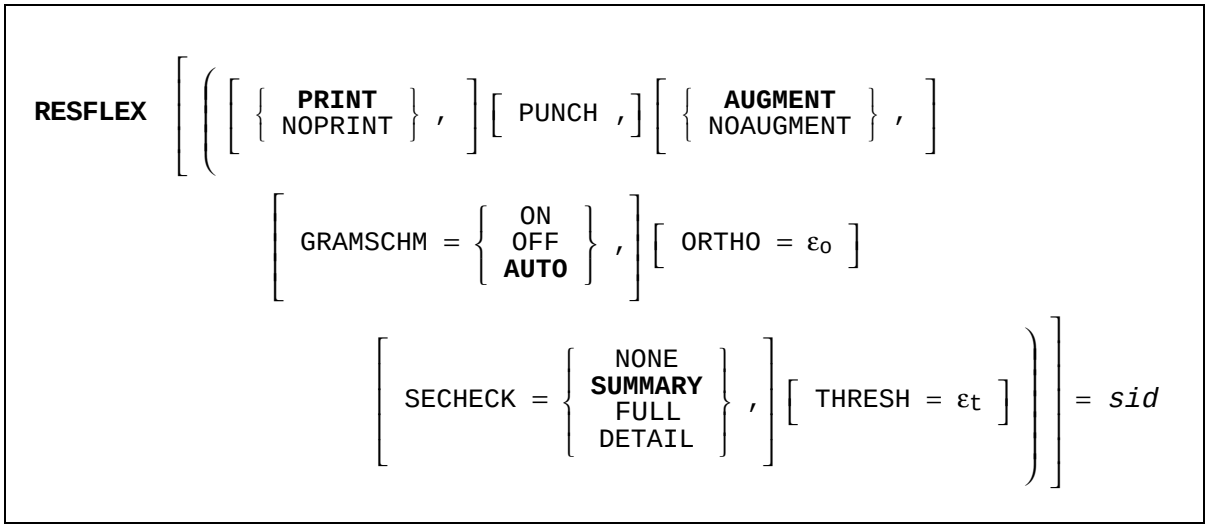
Remarks:

1. The SUBCASE identification numbers *subid* must be greater than all previous SUBCASE identification numbers.
2. Only output from the last SUBCASE appearing in the Case Control packet may be selected. This allows additional set specification within the REPCASE.
3. The **REPCASE** command is only available in Static or Inertia Relief analyses.
4. One or more REPCASEs must immediately follow the SUBCASE which they reference.

Case Control Command RESFLEX

Selects the residual flexibility option to augment the generalized degrees of freedom in modal frequency analysis with additional vectors that are solutions to referenced Statics Analyses. This can improve the accuracy of modal solution results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that Strain Energy Checks be written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results of Strain Energy Checks be written on the **PUNCH** file. [1]	Keyword
{ **AUGMENT** / NOAUGMENT }	Specifies that the associated static solution vectors be used to **AUGMENT** the generalized degrees of freedom, or only for **SECHECK** computations.	Keyword
{ ON / OFF / **AUTO** }	Specifies that Gram-Schmidt orthogonality checks be performed (**ON**), not performed (**OFF**), or performed **AUTO**matically if the **ORTHO**gonality check fails. [2]	Keyword
ε_o	Specifies the **ORTHO**gonality check criterion.	Real≥0.0 Default=0.01
{ NONE / **SUMMARY** / FULL / DETAIL }	Selects the type of report to be printed from the Strain Energy Checks. [3]	Keyword
ε_t	Specifies a threshhold value for printing modal participation factors when **SECHECK=FULL** is selected. Values larger than ε_t are printed. [3]	Real≥0.0 Default=0.0
sid	Provides the identification number *sid* of a Case Control Integer **SET** command which lists **CASE** identification numbers of Statics cases whose solution vectors will be used to augment the modal degrees of freedom. [4]	Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.

2. As a check to ensure the linear independence of the augmented generalized degrees of freedom, a Gram-Schmidt Orthogonalization procedure may be applied to the matrix of normal modes augmented with the residual vectors. Residual vectors that are linear combinations of normal modes or other residual vectors will then be omitted from the augmented degrees of freedom.

   Since this check can be very expensive, three levels of control are provided. When **ON**, the check is performed — typically this would be done during an initial phase when the residual vectors are being selected. When **OFF**, the check is not performed — typically this would be selected once a good set of residual vectors had been identified and qualified using this check. When **AUTO**, the check is performed if the augmented eigenanalysis encounters a singularity or the initial augmented generalized mass matrix fails the orthogonality criterion — this is the default and avoids the computational cost if the vectors are linearly independent, but omits the dependent vectors if the resultant dynamical system is not diagonalized. This is the best overall choice for computational cost and safety. The augmentation process and the orthogonality check is discussed in Chapter 11 of the User Guide.

3. The Strain Energy Check option, **SECHECK**, computes the modal participation factors in the strain energy associated with each residual vector. A **SUMMARY** level shows the total fraction of the each residual vector's strain energy that can be represented by the participating modes. A **FULL** print also shows the fraction of the strain energy that is represented by each participating normal mode and a **DETAIL** print further includes the modal participation factors. The ε_t value controls the **FULL** and **DETAIL** print option to limit the amount of output.

4. This command only applies when performing Modal Frequency Response analyses using the Multidisciplinary Solution Sequence **MULTI**. It refers to

   CASE *id* **STATICS**

   cases whose solution vectors will augment the generalized degrees of freedom.

Example:

1. Perform a Modal Frequency Response solution using the modes computed in a modal analysis and select the solution vbectors from two statics cases as additional degrees of freedom:

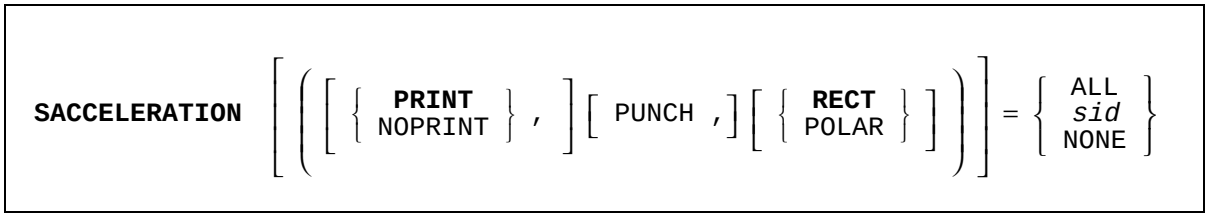
```
APP DISP
SOL MULTI
$
CASE 10 STATICS          First Static Load Case
   LOAD=1000
CASE 20 STATICS          Second Static Load Case
   LOAD=2000

CASE 100 MODES           Normal Modes Case
   METHOD = 1
$
CASE 200 MFREQ
   USE MODES = 100       Requests modes from Case 100
   SET 20100 = 10,20     Defines a set of statics cases 10 and 20
   RESFLEX = 20100       Augments modes with statics results
...
```

Case Control Command SACCELERATION

Requests solution set acceleration results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response analyses.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **SACCELERATION** output are only available for Transient and Frequency Response analyses.

Example:

1. Print the solution set acceleration, in Polar format, for the points specified in **SET 12**:

```
SACCE(POLAR) = 12
```

Case Control Command SDAMPING

Selects a modal damping function for modal Dynamic Response analyses.

Command Syntax:

```
SDAMPING = tid
```

Option	*Meaning*	*Data Range*
tid	Selects a **TABDMP1** Bulk Data entry identification number. [1]	Integer>0

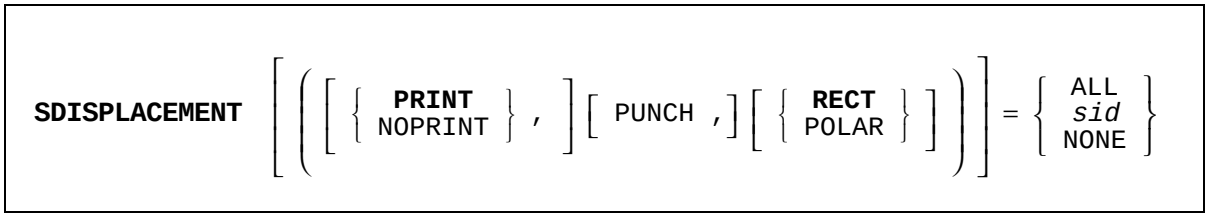
Remarks:

1. The **TABDMP1** data are not used unless selected with this command.
2. In Frequency Response analyses, multiple **SDAMPING** requests are allowed.
3. In Modal Transient analyses, the **SDAMPING** command must appear above the SUBCASE level.

Case Control Command SDISPLACEMENT

Requests solution set displacement results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response analyses.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **SDISP** output are only available for Transient and Frequency Response analyses.

Example:

1. Punch the solution set displacements, in Rectangular format, for the points specified in **SET 12**:

```
SDISP(PUNCH,RECT) = 12
```

Case Control Command SENSITIVITY

Requests design constraint and constraint sensitivity results for DMAP sensitivity analyses.

Command Syntax:

```
SENSITIVITY ( { DISP   } , [ { PRINT   } , ] [ PUNCH , ] [FILE] ) = { ALL   }
              { FREQ   }     { NOPRINT }                             { sid   }
              { STRESS }                                             { NONE  }
              { STRAIN }
              { FORCE  }
              { VECTOR }
```

Option	*Meaning*	*Data Range*
{ DISP, STRESS, STRAIN, FORCE, VECTOR }	Selects the type of constraint sensitivities to output from the GRID point **DISP**lacement, natural **FREQ**uency, element **STRESS**, **STRAIN** or **FORCE**, or eigen**VECTOR** sensitivity.	Keyword
{ **PRINT**, NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
FILE	Specifies that the output sensitivities will be written on logical file **DESSEN**. [2]	Keyword
{ ALL, *sid*, NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [3]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. Unless a default is provided in your Configuration, you must use the **ASSIGN** Executive Control command to define the logical file **DESSEN** with **USE=OUTPUT4** if you use the **FILE** option.
3. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
4. The **SENSITIVITY** output are only available when performing Sensitivity analyses using Rigid Formats 51 and 52. If this command is used in Multidiscipinary Design Optimization, it is ignored. In this case, you must use the **ARCHIVE** command to recover optimization and sensitivity solution results.

Examples:

1. Print the displacement sensitivities for the points specified in **SET 101** and the stress sensitivities for the element in **SET 201**:

```
SENS(DISP)=101
SENS(STRESS)=201
```

2. Write the sensitivities for all eigenvectors on the **DESSEN** file:

```
SENS (VECTOR,NOPRINT,FILE) = ALL
```

Case Control Command SET – Integer List

Defines a list of integer identification numbers.

Command Syntax:

```
SET  sid  = { term1 [,term2,... ] }
            { ALL                 }

   termi  ⇒  idnum1 [THRU idnum2 [EXCEPT idlist]]
```

Option	*Meaning*	*Data Range*
sid	Defines the set identification number. [1]	Integer>0
termi	Specifies one or more identification numbers. Three forms may be used: a single integer identifier; a range specification which specifies an inclusive range of identification numbers; or an exception modifier for a range specification which removes the specified identification numbers from the range.	
idnumi	Defines a single identification number.	Integer>0 *idnum1*<*idnum2*
idlist	Specifies a list of one or more identification numbers, separated by commas, that will be excluded from the preceding range. [2]	Integer>0
ALL	Defines the set to contain all identification numbers. The actual elements in the set are determined by the operation that you perform using the set.	Keyword

Remarks:

1. Set identification numbers must be unique.
2. The identification numbers to be **EXCEPT**ed must be in ascending order.
3. Sets defined within a SUBCASE may only be used in that SUBCASE.

Examples:

1. Define **SET 5** which contains elements **1**, **8**, **54**, **101-201**, **306** and **515**:

```
SET 5 = 1,8,54,101 THRU 201,306,515
```

2. Define the same set, but assume that you do not want to include identification numbers **118**, **154** and **192**:

```
SET 5 = 1,8,54,101 THRU 201 EXCEPT 118,154,192,306,515
```

4 CASE

Case Control Command SET – Real List

Defines an explicit list of real values.

Command Syntax:

SET *sid* = *rval1* [,*rval2*,*rval3*,...]

Option	*Meaning*	*Data Range*
sid	Defines the set identification number. [1,2]	Integer>0
rvali	Specifies a real value. [3]	Real>0.0

Remarks:

1. Set identification numbers must be unique.
2. Sets defined within a SUBCASE may only be used in that SUBCASE.
3. The most common use of real sets is to specify times or frequencies at which output will generated. In this case, the solution frequencies or times nearest those in the set will be output.
4. Sets of real values are only used in Transient and Frequency Response analyses.

Example:

1. Request output at times of **5.0**, **20.0**, **50.0** and **100.0**:

```
SET 1 = 5.0,20.0,50.0,100.0
```

Case Control Command SET – Real Generated List

Defines a computed list of real values.

Command Syntax:

```
SET ( { FREQ } ) sid = range1 [,range2 [,range3 ] ]
    ( { TIME } )
        rangei ⇒ rstart,rstep,rend
```

Option	*Meaning*	*Data Range*
sid	Defines the set identification number. [1,2]	Integer>0
rangei	Defines one, two, or three ranges of real values. [3]	Real
rstart *rstep* *rend*	Specifies the *start*ing value, the *end*ing value and the increment, or *step* size, used to generate intermediate values for each range.	Real>0.0

Remarks:

1. Set identification numbers must be unique.
2. Sets defined within a SUBCASE may only be used in that SUBCASE.
3. The most common use of real sets is to specify times or frequencies at which output will generated. In this case, the solution frequencies or times nearest those in the set will be output.
4. Sets of real values are only used in Transient and Frequency Response analyses.

Example:

1. Generate a list of time values, defined as **SET 100**, which begin with **0.0** and end with **100.0** such that the steps are **0.5** until time **50.0** and then the steps are **1.0**:

```
SET(TIME)100 = 0.0,0.5,50.,50.,1.0,100.
```

2. Generate a list of frequency values which go from **10.0** to **100.0** in steps of **10.0**:

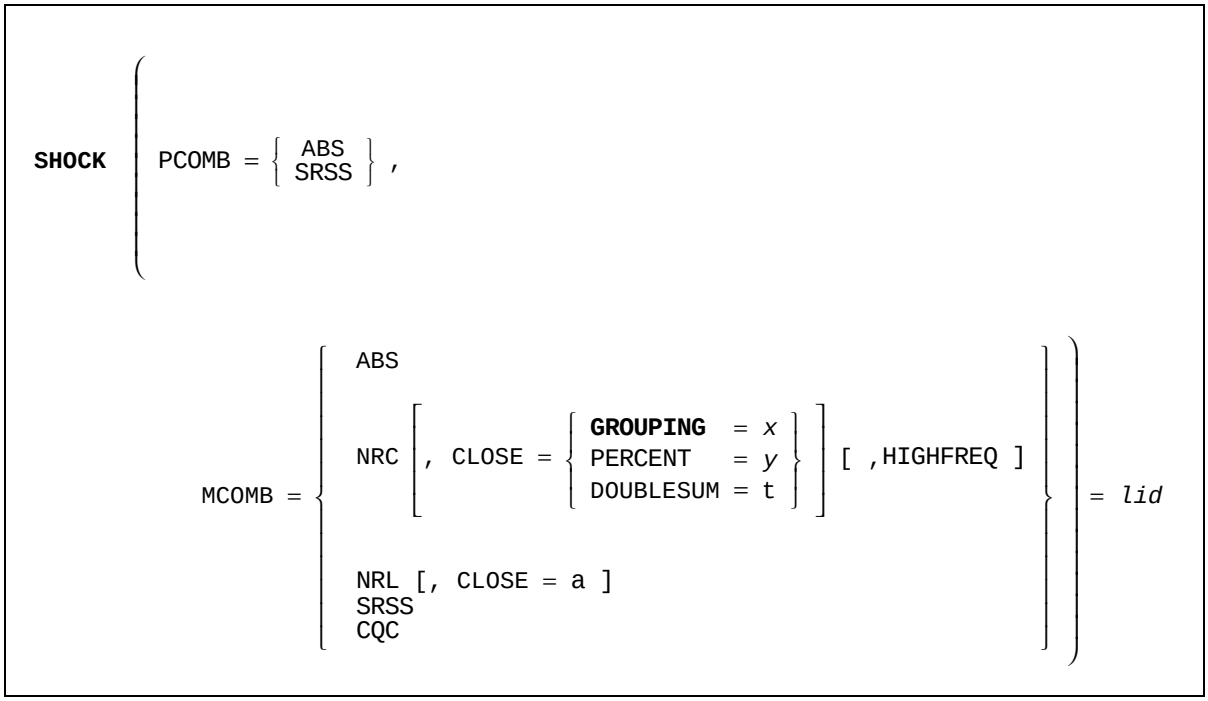
```
SET(FREQ)200 = 10.0,10.0,100.0
```

4 CASE

Case Control Command SHOCK

Selects the Shock Response Spectra loading condition for the structural model.

Command Syntax:



Option	*Meaning*	*Data Range*
lid	Specifies a shock sprectra load identification number. [1]	Integer>0
{ ABS / SRSS }	Selects the shock point combination rule from sum of the absolute values, **ABS**, or square root of the sum of the squares, **SRSS**. [2]	Keyword
{ ABS / NRC / NRL / SRSS / CQC }	Selects the peak modal combination rule from sum of the absolute values, **ABS**, Nuclear Regulatory Commision, **NRC**, Naval Research Laboratories, **NRL**, square root of the sum of the squares, **SRSS**, or complete quadratic, **CQC**. [2]	Keyword
{ **GROUPING** = *x* / PERCENT = *y* / DOUBLESUM = *t* }	Selects the **NRC** closely spaced mode treatment rule. [2]	Keyword
HIGHFREQ	Selects the **NRC** high frequency combination rule. [2]	Keyword
CLOSE = *a*	Selects the **NRL** closely spaced mode treatment rule. [2]	Real

Remarks:

1. One or more **SHOCK** Bulk Data entries with the same load identification numbers may be referenced on a **SHOCK** command.

2. Details describing mode combination and close mode rules are given in Chapter 15 of the ***User's Guide***.

3. The **SHOCK** command may only be used when you are performing a Shock Spectra Response analysis using Rigid Format 17.

Example:

1. Select a shock loading with an identification number of **100**. Use the **SRSS** shock point combination rule, and the **NRC** peak modal combination rule with the **GROUPING** rule for the treatment of closely spaced modes:

```
SHOCK(PCOMB=SRSS,MCOMB=NRC,CLOSE=GROUPING) = 100
```

Case Control Commands SKIPON and SKIPOFF

Disables and enables the processing of Case Control commands.

Command Syntax:

```
SKIPON
SKIPOFF
```

Remarks:

1. The processing of Case Control commands is suspended when the **SKIPON** command is encountered in the Case Control packet. All following commands are echoed but not processed until a **SKIPOFF** command is encountered. Echoing of skipped commands may be controlled by the **ECHOOFF** and **ECHOON** commands.
2. As many pairs of **SKIPON, SKIPOFF** commands may be used as desired.
3. The **SKIPOFF** option is automatically invoked for the Bulk Data packet at the completion of processing the Case Control packet. Skipping of data in the Bulk Data packet is then controlled by the Bulk Data entries **SKIPON** and **SKIPOFF**.

Case Control Command SORT

Selects the sort order of output solution results.

Command Syntax:

$$\textbf{SORT} = \left\{ \begin{array}{l} \textbf{SORT1} \\ \text{SORT2} \end{array} \right\}$$

Option	*Meaning*	*Data Range*
$\left\{ \begin{array}{l} \textbf{SORT1} \\ \text{SORT2} \end{array} \right\}$	Selects the output solution results sort order. [1]	Keyword

Remarks:

1. **SORT1** output is available in all Rigid Formats except those for Transient Response.
2. **SORT2** may be requested only in Static, Transient, and Frequency Response analyses.
3. For Transient Response analyses, output is always given in **SORT2**.

Case Control Command SPC

Selects the single-point constraint set to be applied to the model.

Command Syntax:

```
SPC = sid
```

Option	*Meaning*	*Data Range*
sid	Selects single-point constraint set identification number. [1]	Integer>0

Remarks:

1. The identification number *sid* references **SPC**, **SPC1, SPCADD, SPCS** and **SPCS1** Bulk Data entries. These data are not used unless selected with this command.

Case Control Command SPCFORCES

Requests single-point forces of constraint results.

Command Syntax:

SPCFORCES $\left[\left(\left[\left\{ \begin{matrix} \textbf{PRINT} \\ \text{NOPRINT} \end{matrix} \right\}, \right] \left[\text{PUNCH}, \right] \left[\left\{ \begin{matrix} \textbf{RECT} \\ \text{POLAR} \end{matrix} \right\}, \right] \left[\text{SPLIT} \right] \right) \right] = \left\{ \begin{matrix} \text{ALL} \\ \textit{sid} \\ \text{NONE} \end{matrix} \right\}$

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response analyses.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0
SPLIT	Requests that the **PRINT** or **PUNCH** results be split into two parts: forces due to user-specified SPC's; and forces due to AUTOSPC constraints.	Keyword

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **SPCFORCE** output are available for all analyses.

Example:

1. Print the **SPC** forces for all points in **SORT2** order using **POLAR** format. Also write the results on the **PUNCH** file:

```
SORT = SORT2
...
SPCFORCES(PUNCH,PRINT,POLAR) = ALL
```

Case Control Command STEP

Delimits and identifies a nonlinear analysis STEP.

Command Syntax

```
STEP stepid
```

Option	*Meaning*	*Data Range*
stepid	Defines the STEP identification number.	Integer>0

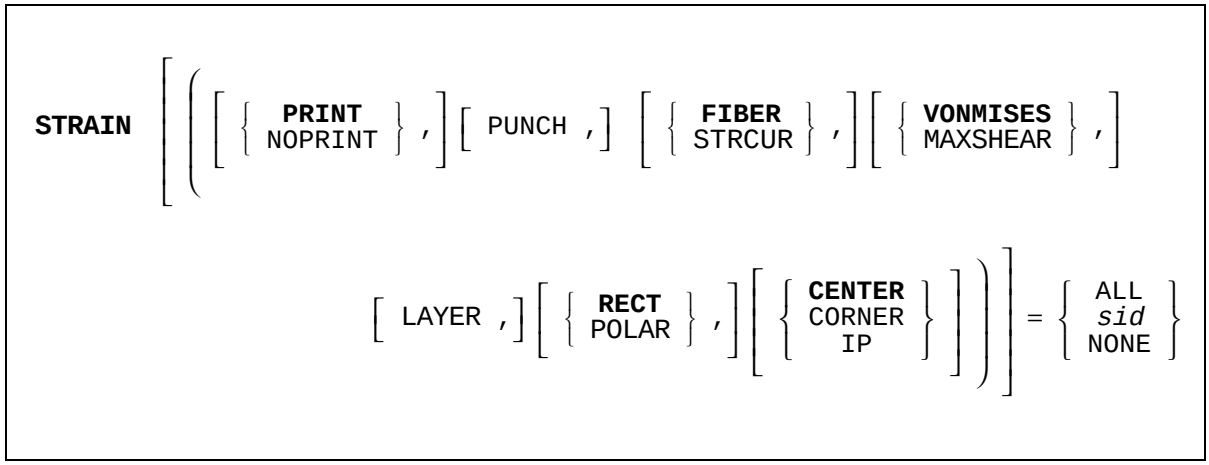
Remarks:

1. The **STEP** numbers must begin with 1 for each new SUBCASE and must be incremented by one only.
2. The **STEP** command may only be used in Nonlinear Static analyses.

Case Control Command STRAIN

Requests element strain results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **FIBER** / STRCUR }	Requests strains to be calculated at the extreme (top and bottom) fibers of a plate element or as strains and curvatures at the midsurface of the element.	Keyword
{ **VONMISES** / MAXSHEAR }	Requests either the calculation of Hencky-von Mises equivalent strain or maximum shear strain for plate elements or octahedral shear strain for solid elements.	Keyword
LAYER	Requests that for composite elements individual layer strains and failure indices be calculated.	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response problems.	Keyword
{ **CENTER** / CORNER / IP }	Selects the location of strain computation within two- and three-dimensional elements. [2,3]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** elements, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the elements at which output will be generated. **NONE** disables the request. [4]	Keyword or Integer>0

4 CASE

Rev: V20.1

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. The selection of strain locations depends on the type of elements in your model. The following table describes these.

CENTER	Element strains for all two- and three-dimensional elements are printed at the element center. In addition, the TRIA6, QUAD8, PENTA, HEXA and TETRA elements print the strains at the corner points of each element. This is the default value.
CORNER	Elements strains for any TRIA3, TRIAR, QUAD4, and QUADR elements are computed for, and printed at, the corner GRID points and center of each element. This is also true for the TRIA6, QUAD8, PENTA, HEXA and TETRA elements as above.
IP	Element strains are computed and printed at the integration points and center for all plate and solid elements. In the output, these are labelled by the GRID point closest to the integration point.

3. If the **IP** option is selected by **any FORCE**, **STRAIN**, or **STRESS** Case Control command, then the IP option will be used for **all** of these output requests.
4. The **STRAIN** is computed for the BAR, BEAM, ROD, TUBE, PILE, SHEAR, TWIST, QUAD4, TRIA3, QUAD8, TRIA6, QUADR, TRIAR, HEXA, PENTA and TETRA elements.
5. The **STRAIN** output are available for all analysis disciplines.

Example:

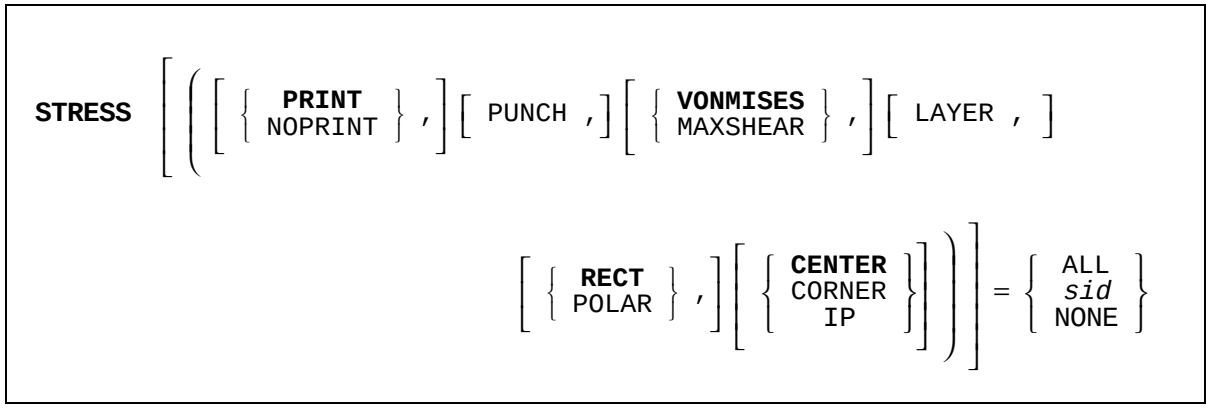
1. Print the midsurface strains and curvature for all plate elements and Von Mises strains for all elements. Also write the results to the **PUNCH** file:

```
STRAIN(STRCUR,VONMISES,PUNCH) = ALL
```

Case Control Command STRESS

Requests element stress results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **VONMISES** / MAXSHEAR }	Requests either the calculation of Hencky-von Mises equivalent stresses or maximum shear stress for plate elements or octahedral shear stress for solid elements.	Keyword
LAYER	Requests that for composite elements individual layer stresses and failure indices be calculated.	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response problems.	Keyword
{ **CENTER** / CORNER / IP }	Selects the location of stress computation within two- and three-dimensional elements. [2,3]	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** elements, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the elements at which output will be generated. **NONE** disables the request. [4]	Integer>0

4 CASE

Rev: V20.1

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.

2. The selection of stress locations depends on the type of elements in your model. The following table describes these.

CENTER	Element stresses for all two- and three-dimensional elements are printed at the element center. In addition, the TRIA6, QUAD8, PENTA, HEXA and TETRA elements print the stresses at the corner points of each element. This is the default value.
CORNER	Elements stresses for any TRIA3, TRIAR, QUAD4, and QUADR elements are computed for, and printed at, the corner GRID points and center of each element. This is also true for the TRIA6, QUAD8, PENTA, HEXA and TETRA elements as above.
IP	Element stresses are computed and printed at the integration points and center for all plate and solid elements. In the output, these are labelled by the GRID point closest to the integration point.

3. If the **IP** option is selected by **any FORCE**, **STRAIN**, or **STRESS** Case Control command, then the IP option will be used for **all** of these output requests.

4. The **STRESS** output are available for all analyses.

Example:

1. Print the Hencky-von Mises stress resultants for elements in **SET 10** and request that stresses at the GRID points be printed for any QUAD4 and TRIA3 elements in the model:

```
STRESS(PRINT,VONMISES,CORNER) = 10
```

Case Control Command SUBCASE

Delimits and identifies a SUBCASE.

Command Syntax:

```
SUBCASE subid
```

Option	*Meaning*	*Data Range*
subid	Defines a SUBCASE identification number. [1]	Integer>0

Remarks:

1. Each SUBCASE identification, *subid*, must be greater than all previous SUBCASE identification numbers.

Case Control Commands SUBCOM and SUBSEQ

Delimits and identifies a combination SUBCASE and the combining coefficients.

Command Syntax:

SUBCOM *subid*
SUBSEQ = *coeff1* [, *coeff2* , ... , *coeffn*]

Option	*Meaning*	*Data Range*
subid	Specifies a SUBCASE identification number. [1]	Integer>2
coeffi	Defines the coefficients to be applied to the previously occurring SUBCASEs. [2]	Real

Remarks:

1. Each SUBCASE identification number must be greater than all previous SUBCASE identification numbers.
2. The *N* **SUBSEQ** coefficients are applied to the immediately, previously defined *N* SUBCASEs above the current SUBCOM.
3. The **SUBCOM** and **SUBSEQ** commands are only available in Static or Inertia Relief analysis.
4. For Axisymmetric Harmonic Analysis, if pairs of Cosine series and Sine series are being executed to be combined with **SUBCOM**, then all Cosine SUBCASEs should be run one after the other, and then these *N* cases should be followed by *N* Sine SUBCASEs, and then the SUBCOM case should be defined.

Example:

1. Define a **SUBCOM** which combines SUBCASEs **100**, **200** and **300** using the ratio **2.5**, **1.5**, and **1.2**:

```
SUBCASE 100
   ...
SUBCASE 200
   ...
SUBCASE 300
   ...
SUBCOM 400
   SUBSEQ = 2.5,1.5,1.2
      ...
```

Case Control Command SUBTITLE

Defines titling information which will appear on the second heading line of each page of printed output.

Command Syntax:

SUBTITLE = *any_character_string*

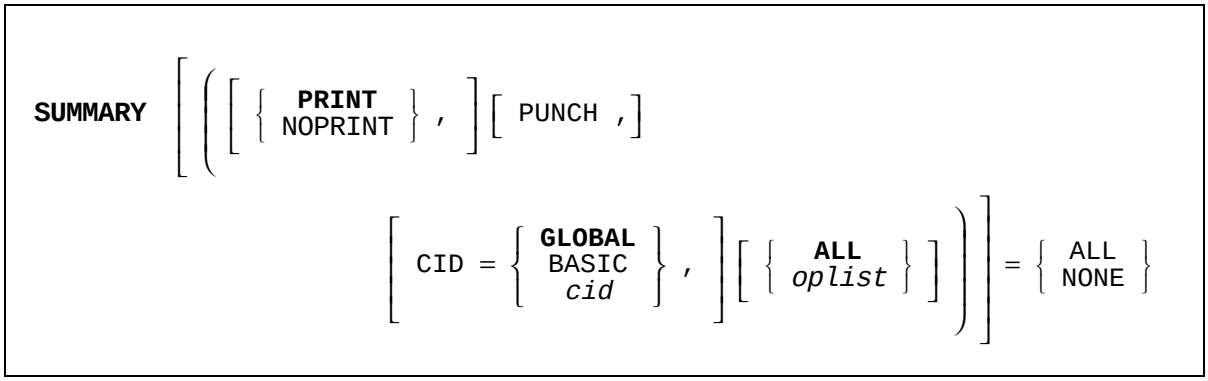
Remarks:

1. A **SUBTITLE** command appearing at the SUBCASE level will title output for that SUBCASE only.
2. A **SUBTITLE** command which you place before all SUBCASEs will title any outputs which are not SUBCASE dependent.
3. If the **SUBTITLE** command is not used, the **SUBTITLE** line is blank.
4. The **SUBTITLE** information is also placed on plots as applicable.
5. The **SUBTITLE** command must be contained on a single input line. It may not be continued.

Case Control Command SUMMARY

Requests a summary of the minimum and maximum GRID point solution quantities.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that min/max summaries be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
ALL	Selects **ALL** summary options which include **DISP**lacement, **OLOAD**, **SPCF**, and **MPCF**.	Keyword
oplist	Specifies a list of one or more summary options selected from **DISP**lacement, **OLOAD**, **SPCF**, or **MPCF**.	Keyword
{ **GLOBAL** / BASIC / *cid* }	Specifies the coordinate system in which the summary calculations will be performed. Selected from the **GLOBAL** coordinate system, the **BASIC** coordinate system, or a coordinate system defined by *cid*.	Keyword or Integer>0
{ ALL / NONE }	Enables the summary request for **ALL** options. **NONE** disables the request. [2]	Keyword

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. If you are using the standard Rigid Formats, the **SUMMARY** command should be placed above the SUBCASE level.
3. If you are using **SOL MULTI**, you may place the **SUMMARY** commands in different disciplines to enable and disable the options. When used with Statics analysis, you should place the request in the first CASE of each different boundary condition.

Rev: V20.1

Example:

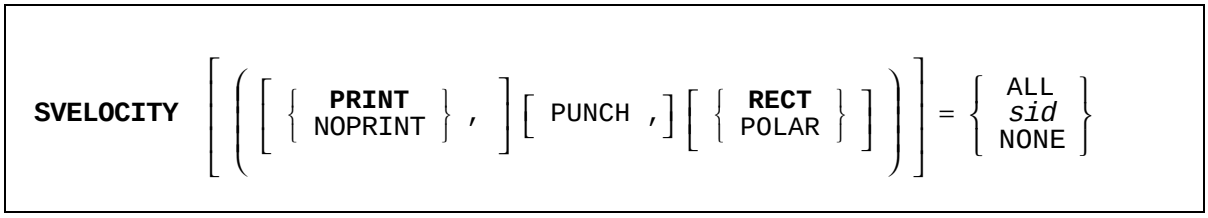
1. Enable the **OLOAD** min/max summary for Static analysis and disable it for the Normal Modes analysis in a **MULTI** job.

```
CASE 1 STATICS
   SUMMARY(OLOAD)=ALL
CASE 2 STATICS
   ...
CASE 3 STATICS
   ...
CASE 4 MODES
   SUMMARY(ALL) = NONE
```

Case Control Command SVELOCITY

Requests solution set velocity results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response analysis.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **SVELOCITY** output are only available for Transient and Frequency Response analyses.

Example:

1. Print the solution set velocities for points in **SET 101** using the **POLAR** format:

```
SVELOCITY(POLAR) = 101
```

Case Control Command SYM

Delimits and identifies a symmetry SUBCASE.

Command Syntax:

```
SYM  subid
```

Option	*Meaning*	*Data Range*
subid	Specifies a SUBCASE identification number.	Integer>0

Remarks:

1. Each SUBCASE identification number must be greater than all previous SUBCASE identification numbers.
2. Overall output requests do not propagate into a **SYM** SUBCASE. All output desired must be requested within the **SYM** SUBCASE.
3. The **SYM** command is only available only for Static or Inertia Relief analysis.

Case Control Commands SYMCOM and SYMSEQ

Delimits and identifies a symmetry combination SUBCASE and the combining coefficients.

Command Syntax:

```
SYMCOM subid
       SYMSEQ = coeff1 [ , coeff2 ,..., coeffn ]
```

Option	*Meaning*	*Data Range*
subid	Specifies a SYM case identification number. [1]	Integer>2
coeffi	Defines coefficients to be applied to previously occurring SYM cases. [2,3]	Real

Remarks:

1. Each SYM identification number must be greater than all previous SYM identification numbers.
2. The *n* **SYMSEQ** coefficients are applied to the *n* SYM cases appearing immediately above the current **SYMCOM**.
3. If the **SYMSEQ** subcommand is not used, all coefficients are set to 1.0.
4. The **SYMCOM** and **SYMSEQ** commands are only available for Static and Inertia Relief analyses.
5. For Axisymmetric Harmonic Analysis, if pairs of Cosine series and Sine series are being executed to be combined with **SYMCOM**, then all Cosine SYM cases should be run one after the other, and then these *n* cases should be followed by *n* Sine SYM cases, and then the **SYMCOM** case should be defined.

Example:

1. Define a **SYMCOM** which combines **SYM** cases **1**, **2** and **3** using the ratio **1.5**, **0.8**, and **0.5**:

```
SYM 1
   ...
SYM 2
   ...
SYM 3
   ...
SYMCOM 4
   SYMSEQ = 1.5,0.8,0.5
      ...
```

Case Control Command SYMMETRY

Specifies a symmetry condition for unsteady aerodynamic matrix generation and flutter analysis.

Command Syntax:

```
SYMMETRY = symtype
```

Option	*Meaning*	*Data Range*
symtype	Selects the symmetry condition. [1,2]	Keyword

Remarks

1. The symmetry conditions are all defined in the XZ and XY planes of the **ACID** aerodynamic coordinate system referenced on the **AEREFS** Bulk Entry.

2. **SYMTYPE** specifies the aerodynamic symmetry condition. The **SYMMETRY** option is used to specify aerodynamic boundary conditions due to planes of symmetry or wind tunnel wall reflections. The available selections listed below are meaningful when the vehicle is properly oriented in the **ACID** coordinate system. The desired orientation would be with the vertical fin located in the XZ plane of the **ACID** system, and the wing generally oriented in the XY plane (except for dihedral). The alternate form is available for cases where the vehicle is modeled in an arbitrary orientation to the **ACID** aero coordinate system. Legal values for *symtype* are shown in the following table:

SYMMETRY CONDITION	*symtype*	ALTERNATE FORM	MEANING
SINGLE	NOSYMM	0 or blank	No Symmetry condition in any plane.
	SYM	+XZ	Symmetric in the XZ plane.
	ANTI	-XZ	Anti-Symmetric in the XZ plane.
	GRND	+XY	Symmetric in the XY plane, also known as *Ground Effect symmetry*.
	BIPL	-XY	Anti-Symmetric in the XY plane, also known as *Biplane symmetry*.
COMBINED	SYMMGRND	+XZ+XY	Combinations of symmetries described above.
	SYMMBIPL	+XZ-XY	
	ANTIGRND	-XZ+XY	
	ANTIBIPL	-XZ-XY	

3. The **SYMMETRY** command is required if you have specified either **CASE AERO** or **CASE FLUTTER**.

Case Control Command TEMPERATURE

Selects the temperature distribution to be used for temperature dependent material property calculations or thermal loadings.

Command Syntax:

TEMPERATURE $\left[\left(\left\{\begin{matrix}\text{MATERIAL}\\ \text{LOAD}\\ \textbf{BOTH}\\ \text{ESTIMATE}\end{matrix}\right\}\right)\right] = sid$

Option	*Meaning*	*Data Range*
{ MATERIAL, LOAD, **BOTH** }	Specifies a temperature distribution that is used to determine either the temperature dependent **MATERIAL** properties, the equivalent static **LOAD**s, or **BOTH**. [1,2]	Keyword
ESTIMATE	Selects initial temperature estimates for Heat Transfer analyses. [3]	
sid	Selects a temperature set identification number. [4]	Integer>0

Remarks:

1. Temperature dependent materials are defined with **MATTi** Bulk Data. These data are not used unless a **TEMP(MATERIAL)** or **TEMP(BOTH)** command is specified.
2. Only one **TEMP(MATERIAL)** or **TEMP(BOTH)** command may appear in your Case Control command packet. If one of these commands is used, it must be placed above the SUBCASE level.
3. The temperature **ESTIMATE** is used as an intial temperature vector for Nonlinear Steay-State Heat Transfer analyses.
4. The identification number *sid* references **TEMP**, **TEMPD**, **TEMPPi**, **TEMPRB** and **TEMPAX** Bulk Data entries. These data will not be used unless selected with the **TEMP** command.
5. Thermal loading is available in Static, Inertia Relief, Differential Stiffness, and Buckling analyses.

Examples:

1. Select a temperature distribution for thermal loading which references **SET 15**:

```
TEMPERATURE(LOAD) = 15
```

2. Select a temperature distribution for temperature dependent material property which references **SET 100**:

```
TEMPERATURE(MATERIAL) = 100
```

Case Control Command TFL

Selects a dynamic Transfer Function set.

Command Syntax:

```
TFL = sid
```

Option	*Meaning*	*Data Range*
sid	Selects a **TF** Bulk Data entry identification number. [1]	Integer>0

Remarks:

1. **TF** Bulk Data entries will not be used unless selected with this command.
2. Transfer Functions, which are a form of direct matrix input, are available in dynamic response analyses only.

Case Control Command THERMAL

Requests temperature vector results in Heat Transfer analysis.

Command Syntax:

```
THERMAL [ ( [ { PRINT } , ] [ PUNCH , ] [ BULK , SID = lid ] ) ] = { ALL }
        [ ( [ { NOPRINT } ] )                                     ]   { sid }
                                                                      { NONE }
```

Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
BULK	Requests that temperature results be output as **TEMP** Bulk Data entries on the **BULK** file. Temperatures for all points will be output. [1]	Keyword
lid	Specifies a load set identification number *lid* that is placed on resulting **TEMP** Bulk Data entries.	Integer>0
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [3]	Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option and a logical file with **USE=BULK** if you use the **BULK** option. Output to both the **PUNCH** and **BULK** files is available for all Heat Transfer Analyses. For linear and nonlinear static analysis and **BULK** file output, the temperature load set identification number is defined by *lid*. For transient analysis, unique load set identification numbers are created from the expression:

   lid = 1000 * *sid* + TIME_STEP_NUMBER

   A table listing the *lid* versus time will be placed in the print file for this transient analysis case.

2. **THERMAL** output is available for Heat Transfer analysis.

3. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.

4. Results are available in **SORT1** order for Static analysis and in either **SORT1** or **SORT2** order for Transient analysis.

Example:

1. Print all of the temperature results and create **TEMP** Bulk Data entries having a temperature load **SET** identification number of **2**:

```
THERMAL(PRINT,BULK,SID=2) = ALL
```

Case Control Command TITLE

Defines titling information which will appear on the first heading line of each page of printed output.

Command Syntax:

```
TITLE = any_character_string
```

Remarks:

1. A **TITLE** command appearing at the SUBCASE level will title output for that SUBCASE only.
2. A **TITLE** command which you place before all SUBCASEs will title any outputs which are not SUBCASE dependent.
3. If the **TITLE** command is not used, the title line contains the date and page number only.
4. The **TITLE** information is also placed on plots as applicable.
5. The **TITLE** command must be contained on a single input line. It may not be continued.

Case Control Command TSTEP

Selects integration and output time steps for Transient Response analysis.

Command Syntax:

TSTEP = *sid*

Option	*Meaning*	*Data Range*
sid	Selects a **TSTEP** Bulk Data entry identification number.	Integer>0

Remarks:

1. A **TSTEP** command is required for Transient Response analyses.

Case Control Command USING

Selects previously computed results from one discipline for use in another discipline.

Command Syntax:

```
USING { [ { COUPLED } ] MODES [CASE] case_id [FROM logical_name [:path_name]] }
      { [ { STRUCTURAL } ]                                                    }
      { [ { FLUID      } ]                                                    }
      { AERO FROM logical_name [:path_name]                                   }
```

Option	*Meaning*	*Data Range*
{ COUPLED / STRUCTURAL / FLUID }	Selects the type of modes to be used from a previous Normal Modes analysis to form generalized coordinates for a new analysis **CASE** which uses the modal approach. [1]	Character
case_id	Identification number of a previously defined **CASE MODES** command whose eigenvectors form generalized coordinates for this modal analysis **CASE**. [2]	Integer>0
logical_name	The logical name of an ***eBase*** database containing the selected data. [3,4]	Character
path_name	Fully-qualified path name describing the location of the required data on the ***eBase*** database selected by *logical_name*. [3,4]	Character
AERO	Specifies that previously computed aerodynamic data will be used for a Flutter analysis CASE. [5]	Keyword

Remarks:

1. The options **COUPLED** and **FLUID** may only be selected when performing Hydroelastic analyses. The option selected must agree with the type of data resulting from the referenced *case_id*.
2. When performing Modal Frequency Response, Modal Transient Response, or Flutter analyses with **SOL MULTI**, the Case Control command:

   CASE *caseid* MODES

   must have been defined ***with the same boundary conditions***. Then, these modes are referenced in the Modal solution CASE definition with the **USING** command.
3. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=ARCHIVE** if you use the **FROM** option.
4. The default *logical_name:path_name* for all **USING** commands is the root directory of the run-time database, i.e. RUNDB:/.
5. The **USING AERO** command is not required unless you have specifically saved the results of an Aerodynamic analysis on an ARCHIVE database. Data on the run-time database are automatically available for all analyses.

Rev: V20.1

Examples:

1. Use the normal modes computed in Case **1001** as generalized degrees of freedom for a Modal Frequency Response case, **2001**:

```
CASE 1001 MODES
   MPC = 5
   SPC = 10
   ...
CASE 2001 MFREQUENCY
   MPC = 5
   SPC = 10
   USING MODES 1001
```

2. Use the normal modes computed in Case **11** as generalized degrees of freedom for a Flutter analysis case, **22**, which uses Aerodynamic loads stored on an ***eBase*** database with the logical name my_aero:

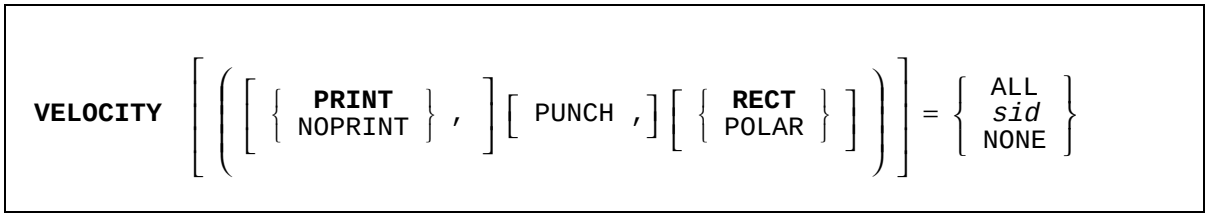
```
MPC = 5
SPC = 10
CASE 11 MODES

   ...
CASE 22 FLUTTER
   FMETHOD = PKITER
   DENS = 100
   VLIST = 1000
   USING MODES 11
   USING AERO FROM my_aero
```

Case Control Command VELOCITY

Requests velocity vector results.

Command Syntax:



Option	*Meaning*	*Data Range*
{ **PRINT** / NOPRINT }	Specifies that results be generated and then written on the print file, or that they only be generated for later post-processing.	Keyword
PUNCH	Requests that results be written on the **PUNCH** file. [1]	Keyword
{ **RECT** / POLAR }	Selects output format for Complex Eigenvalue or Frequency Response analysis.	Keyword
{ ALL / *sid* / NONE }	Enables results generation for **ALL** points, or provides the identification number *sid* of a Case Control Integer **SET** command which lists the points at which output will be generated. **NONE** disables the request. [2]	Keyword or Integer>0

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=PUNCH** if you use the **PUNCH** option.
2. See the introductory remarks in this chapter for additional details providing guidelines for using the various options.
3. The **VELOCITY** output are only available for Transient and Frequency Response analyses.

Example:

1. Print the velocities for all points in **SORT2** using Rectangular format. Also write the results to the **PUNCH** file:

```
SORT = SORT2
...
VELOCITY(RECT,PUNCH) = ALL
```

or, equivalently, since the default format is **RECT**,

```
VELOCITY(PUNCH) = ALL
```

Case Control Command VLIST

Defines the list of velocities used for performing Flutter analysis with the **PKITER** or **PKSWEEP** methods.

Command Syntax:

```
VLIST = { ( V1,V2,...,V1 ) }
        {      vid       }
```

Option	*Meaning*	*Data Range*
V1,V2,...,Vi	Explicit list of velocities. [1,2]	Real>0.0
vid	Identification number of a real **SET** command containing the list of velocities. [1,2]	Integer>0

Remarks

1. The **VLIST** command is only used if you have specified a **CASE FLUTTER**.
2. **VLIST** is only required for **PK** methods if you wish to analyze *off match point* speed conditions. To analyyze only match point conditions, **DENS** Case Control command must reference an **ATMOS** Bulk Data entry, in which case no **VLIST** command is needed.
3. If **FMETHOD=PKSWEEP**, you must also use the **KLIST** Case Control command to define the 'soft point' reduced frequencies. The aerodynamic matrix data at the *hard points* will be automatically interpolated to the *soft point* frequencies. The interpolation method may be selected with the **FLSOLVE** Bulk Data entry.

Examples:

1. Define a set with identification number **1** having three velocities, **0.1**, **0.2** and **0.3**. Select this set as the velocities used in a Flutter analysis.

```
SET 10 = 0.1,0.2,0.3
VLIST = 10
```

2. Define the same three velocities explicitly:

```
VLIST = ( 0.1,0.2,0.3 )
```

This page is intentionally blank.

Chapter 5

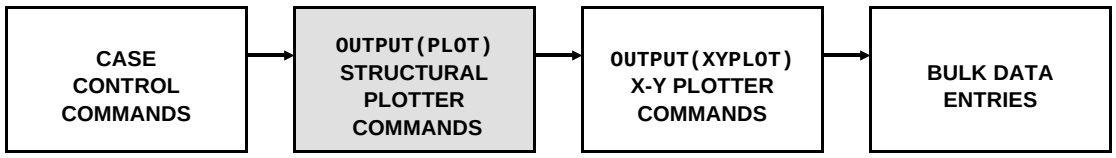
STRUCTURAL PLOTTER COMMANDS

Structural Plotter commands are used to define plots of the finite element model and its GRID point-based structural responses. The actual Structural Plots are not created during your **UAI*/NASTRAN*** execution. Rather, the plot information is placed on a file that is post-processed.

UAI provides four utilities for this purpose. The first two programs, available on most computers, are ***nastplotps*** and ***nastplotgl***. The first of these utilities allows you to send your plots to a postscript laser printer, and the second to a Hewlett-Packard compatible plotting device. The third program, available on Unix host computers is ***nastplot***. This is an interactive program, based on Motif and X-Windows, which allows you to selectively view your plots. The Chapter 1 of this manual provides instructions on how to use these programs for most systems. There is also a program, provided as source code, illustrating how your facility may create their own plotter software, if necessary. In some cases, you must contact your **UAI*/NASTRAN*** Support Specialist for details describing how your site interfaces with these post-processor.

The Structural Plotter commands form a subpacket within the Case Control packet. The location of this subpacket is shown in Figure 5-1.

Figure 5-1. THE STRUCTURAL PLOTTER SUBPACKET



5.1 THE STRUCTURAL PLOTTER COMMANDS

The general form of the Structural Plotter Command packet is:

```
OUTPUT(PLOT)
   option_command
   option_command
   ...
   PLOT_command
   option_command
   option_command
   PLOT_command
   ...
{ OUTPUT(XYPLOT) }
{ BEGIN BULK     }
```

The packet must begin with the **OUTPUT(PLOT)** command and it ends when either an **OUTPUT(XYPLOT)** or **BEGIN BULK** command is encountered. The other commands in the packet define the plot frames that you wish to create. Some of these are *option_command*s which define the characteristics of the plots to be made while others are the *PLOT_command*s which actually specify what is to be plotted. The available Structural Plotter Commands are summarized in Table 5.1 where they are grouped in fuctional categories.

Table 5-1. SUMMARY OF STRUCTURAL PLOTTER COMMANDS

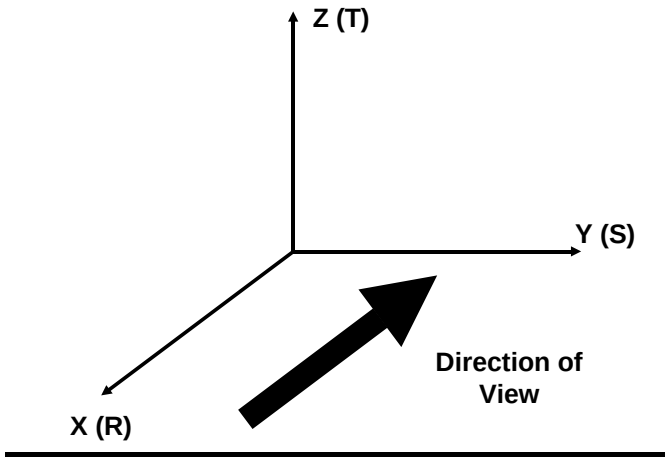
COMMAND TYPE	COMMAND NAME	COMMAND DESCRIPTION
SET DEFINITION COMMANDS	SET	Defines a set of elements that will be plotted.
VIEWING OPTION COMMANDS	AXES	Defines the orientation of the structure coordinate system to the plotter coordinate system.
	PERSPECTIVE FACTOR	Specifies the amount of perspective.
	PROJECTION	Selects orthographic or perspective projection for the plots.
	VANTAGE POINT	Defines your viewing location with respect to the structural model.
	VIEW	Rotates the structural model about the plotter coordinate system.
	ZOOM	Shrinks or enlarges the structural model.
SOLUTION OPTION COMMAND	DEFORMATION SCALE	Defines the scale factor used in plotting GRID point deformations.
PLOT COMMAND	PLOT	Creates a plot of the structural model or its solution results using the active plotter options.
	subplot_list	Defines plotting options for a subset of the model geometry that is to be plotted.
TITLE COMMAND	PTITLE	Specifies a plot frame title.
DEVICE COMMANDS	ASPECT RATIO	Specifies the relative shape of the plotting surface.
	CHARACTER PRECISION	Selects the manner in which characters are plotted by the post-processor.
	CHARACTER SCALE	Specifies a character scaling factor which depends on the size of the plot surface used by the post-processor.
	PEN OPTIMIZATION	Enables or disables pen movement optimization.

5.2 STRUCTURAL PLOTTING TERMINOLOGY

You will find an extensive description of the Structural Plotting capability in the **UAI/*NASTRAN* *User's Guide***. This section reviews important plotting concepts that will assist you in creating plots of your model and its solution results.

Plotter Coordinate System. When performing graphics, there is an underlying coordinate system which is called the plotter coordinate system. The actual structural model coordinate system is mapped to the plotter system prior to plotting. The plotter coordinate system is called the RST-System. This system is fixed with respect to the plot: you are generally looking down the R-axis toward the plotter system origin. Initially, the model coordinate system is aligned with the plotter coordinate system as shown in Figure 5-2. Rotations are always performed with respect to the RST-System.

Figure 5-2. PLOTTER COORDINATE SYSTEM



Graphic Projections. When three-dimensional objects are plotted, there are numerous ways in which their coordinates may be mapped to a two-dimensional plotting surface. **UAI/*NASTRAN*** supports two of these, the ***orthographic projection*** and the ***perspective projection.*** When an object is projected to the surface along parallel lines, it is called orthographic. If, on the other hand, the object is projected along lines that converge to a point, it is called perspective. These two cases are illustrated in Figure 5-3.

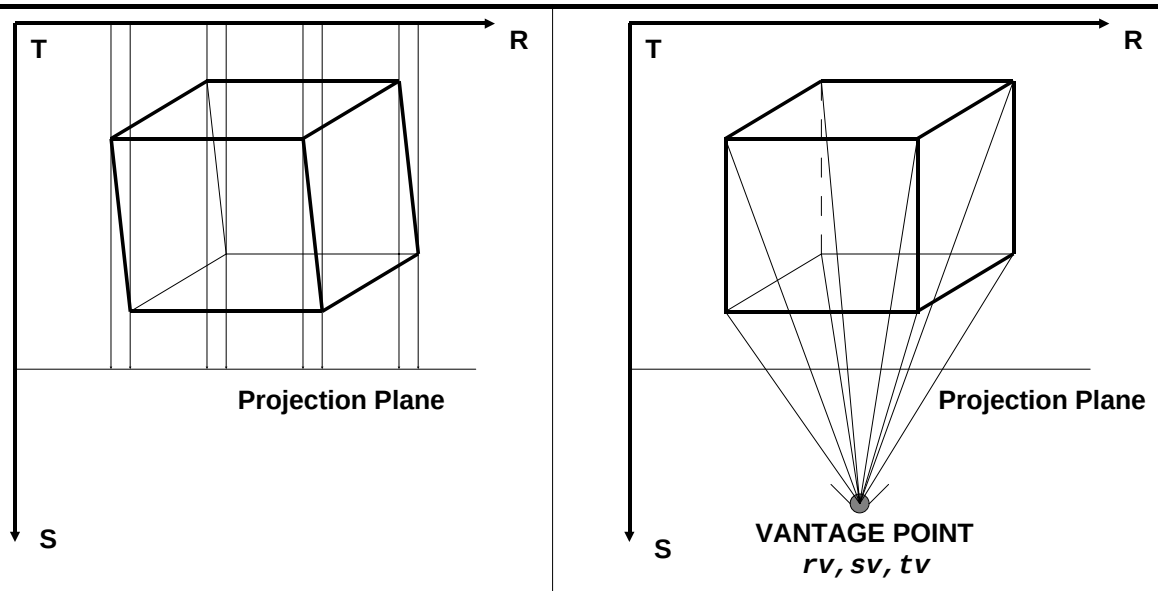
Vantage Point. As seen in Figure 5-3b, the point at which the lines of a perspective projection converge in called the vantage point. By defining the RST-coordinates of your vantage point, you may view your model from a number of different angles.

Plot Sets. A plot set defines a group of elements within your model. Plot sets are specified by using the **SET** command.

Plot. A plot is a single frame which displayes one or more ***subplots*** and, optionally, their solution results. Each of the subplots is defined as a separate plot set and each may have their own graphic characteristics.

Deformed Plot. A deformed plot is one which is created by applying the static deformations to the GRID point locations prior to plotting. Because the deformations are often very small relative to the model, they are scaled in order to be visible. When performing dynamic response analyses, you may also represent the velocities and accelerations as deformations.

Figure 5-3. GRAPHIC PROJECTIONS



Vector Plot. Rather then creating a deformed plot, you may also request that the deformations be plotted in the form of vectors. The vectors originate at the GRID points and point in a direction which is determined by the displacement components that you select.

Labels. You may selectively label the GRID points and finite elements within your model. This may be done when plotting the structural model or its deformed shape.

Coordinate System Triad. Each plot includes a coordinate system triad which shows the orientation of your model coordinate system with respect to the plot surface.

5.3 SET DEFINITION

You may plot all of your model or selected portions of the model which are defined using the **SET** command. The Structural Plotter **SET** command is similar to the Case Control command, but considerably more flexible. The general form of the command is:

```
SET set_id include_part [modifier_part]
```

The *set_id* is a unique integer identification number which is used to reference the set in subsequent Structural Plotter commands. The *include_part* defines the collection of elements that will be members of the set. Its general form is:

```
{ ALL                                                                    }
{                                                                        }
{ [INCLUDE] { element_type } , [ { element_type } ,... ]                 }
{           { element_id   }   [ { element_id   }      ]                 }
{           { element_range }  [ { element_range }     ]                 }
```

You may include **ALL** elements in the model, or you may select one or more *element_type*s from those in the table below.

AXIF2	FLUID2	QUAD8	TRIA3
AXIF3	FLUID4	ROD	TRIA6
AXIF4	HEXA	SHEAR	TRIAAX
BAR	PENTA	TETRA	TRIARG
BEAM	PILE	TORDRG	TUBE
CONE	PLOTEL	TRAPAX	VISC
CONROD	QUAD4	TRAPRG	

You may also select elements by their individual identification numbers, *element_id*, or by specifying an *element_range* of the form:

```
element_id_1 THRU element_id_2
```

The *modifier_part* may then be used to modify the i*nclude_part* by either adding new element types or identifiers, or by excluding selected elements from the previous *include_part*. The syntax of the *modifier_part* is:

```
{ EXCLUDE } { element_type } , [ { element_type } ,... ]
{ EXCEPT  } { element_id   }   [ { element_id   }      ]
{ INCLUDE } { element_range }  [ { element_range }     ]
```

Note that the **EXCLUDE** and **EXCEPT** options are synonymous and may be used interchangeably. The following examples illustrate these different forms.

1. Define **SET 1** as all QUAD4 and TRIA3 elements in your model:

   ```
   SET 1 = QUAD4,TRIA3
   ```

2. Define **SET 3** to contain all HEXA elements except those with identification numbers **601**, **703** and **710**:

   ```
   SET 3 = HEXA EXCLUDE 601,703,710
   ```

3. Define **SET 100** to include all elements except BARs:

   ```
   SET 100 = ALL EXCEPT BAR
   ```

You may select all of the elements in your model by defining:

```
SET n = ALL
```

However, as you will see later, this is not necessary because the Structural Plotter default is to plot the entire model.

5.4 VIEWING OPTION COMMANDS

The viewing option commands are used to specify the characteristics, or elements, of the plot that you wish to create. The three most important viewing options are described in this section.

5.4.1 Viewing Angles

Figure 5-4 shows a typical structural plot with its plot elements annotated. The basic viewing angles, which is shown in the figure, are:

$$(\gamma, \beta, \alpha) = (34.27°, 23.17°, 0.0°)$$

Remember that the order in which the rotations are performed: first γ, then β, and finally α, is crucial. The rotations are specified by the **VIEW** command. The special command **AXES** allows you to perform 90° rotations quickly by simply specifying the correspondence between your models coordinate system and the plotter system. Figure 5-5 shows how the three orthogonal views of a model may be obtained in this manner.

Figure 5-4. BASIC PLOT ELEMENTS

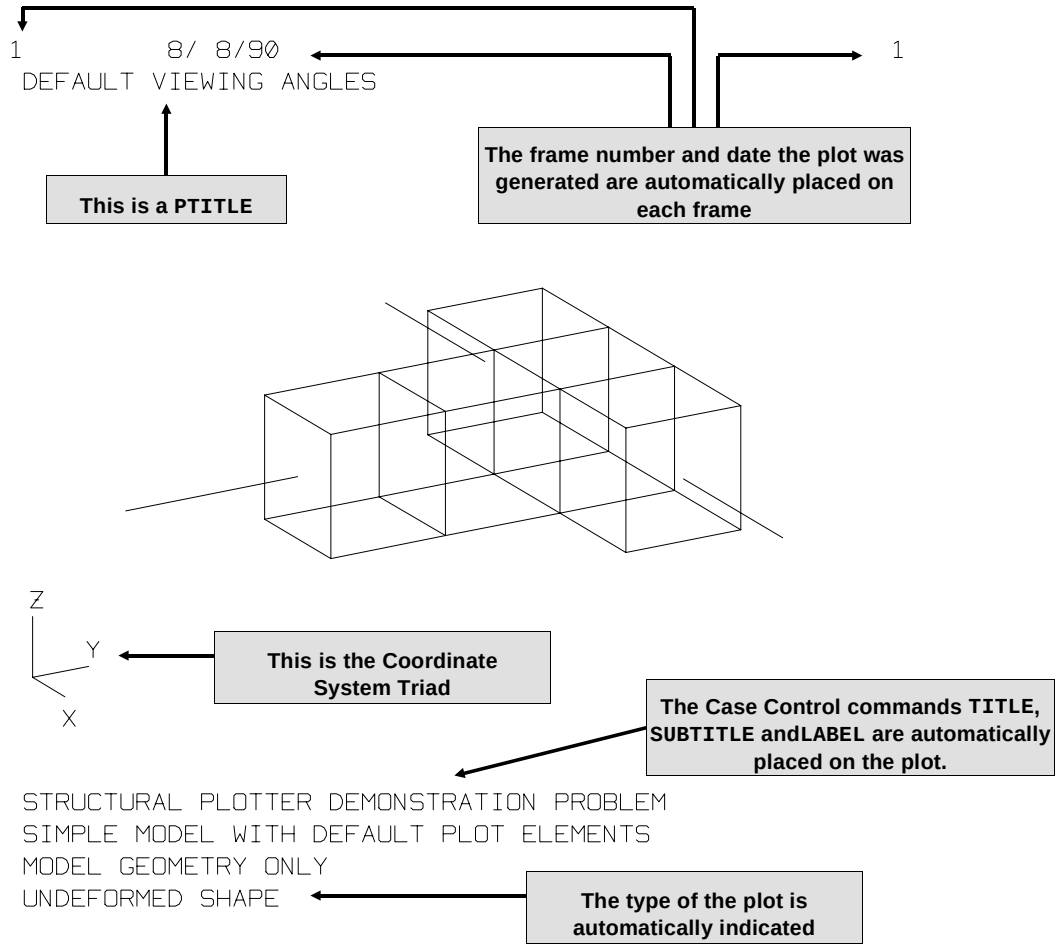
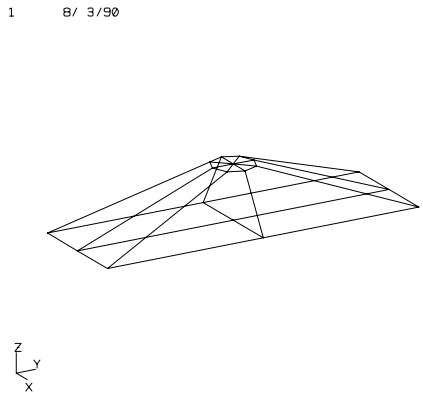
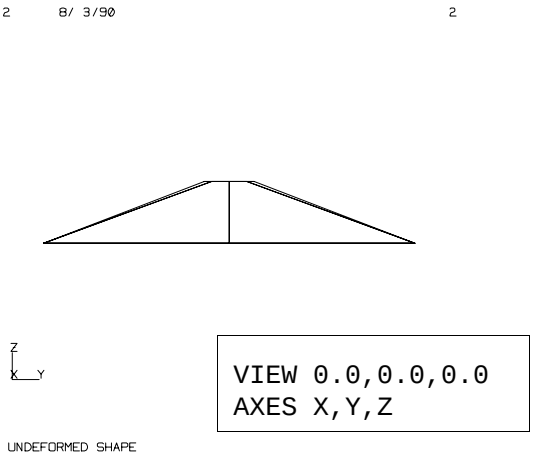


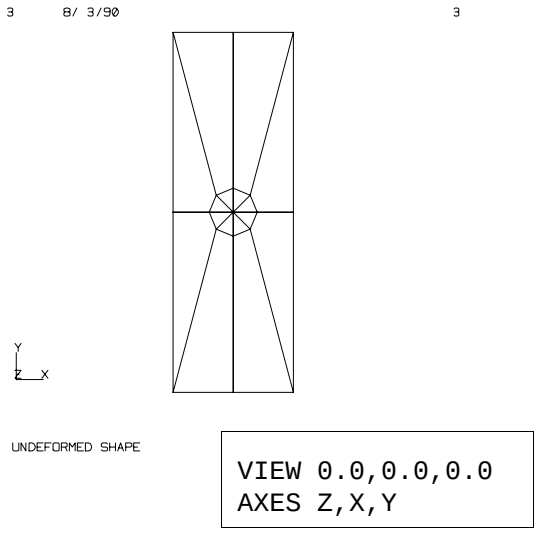
Figure 5-5. USING THE AXES COMMAND



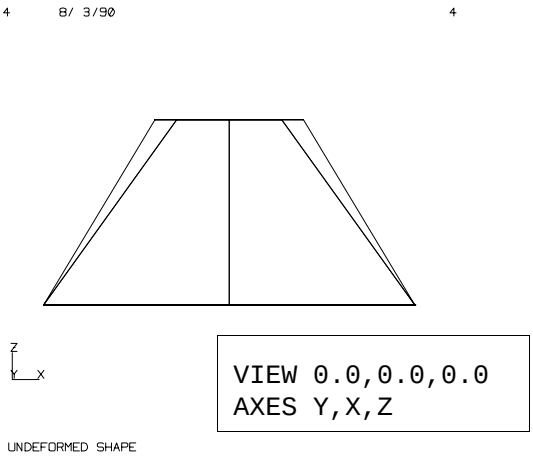
a. DEFAULT VIEWING ANGLES



b. SIDE VIEW



c. TOP VIEW



d. FRONT VIEW

5 PLOT

The viewing angles may also be changed by moving your position relative to the plotter coordinate system. This is done by using the **VANTAGE POINT** command. It is important that you remember the order in which these three commands are performed. The model coordinate system is first aligned with the plotter coordinate system with the **AXES** command. The **VIEW** command is then used to perform specified rotations of the model. Finally, your loaction is moved to the specified **VANTAGE POINT**.

5.4.2 The Graphics Projection

The Structural Plotter provides you with two graphics projections. You select the type of projection by specifying either:

```
ORTHOGRAPHIC PROJECTION or
PERSPECTIVE PROJECTION
```

When using **PERSPECTIVE**, you may also control the **PERSPECTIVE FACTOR** and the **VANTAGE POINT**. These are described in detail in the **UAI/*NASTRAN User's Guide.*** When using the **PERSPECTIVE PROJECTION** the location of the **VANTAGE POINT** is controlled by the **PERSPECTIVE FACTOR** or the **VANTAGE POINT** command. If you use the **PERSPECTIVE FACTOR** command , it is used to automatically compute the distance to the model so that the desired amount of perspective is obtained. When you use a **VANTAGE POINT** command instead, this it is used to determine the distance to the model which, in turn, determines the **PERSPECTIVE FACTOR**. Finally, if you do not use either of these commands, **UAI/*NASTRAN*** automatically determines the distance to the model as twice the maximum dimension of the model.

5.4.3 ZOOMing

The final viewing option command is **ZOOM**. This command allows you to enlarge or reduce the size of the structure about any point that you select.

5.5 PLOTTING THE MODEL GEOMETRY

Once you have defined the general viewing options, or allowed **UAI/*NASTRAN*** to select defaults, and defined any **SET**s that you wish to use, you may create one or more plots of your model. This is done with the command:

```
PLOT [global_options] [subplot_list]
```

The command allows you to specify *global_options*, which control the plot *line_style*, select *label*ing and *symbol* options, and to define the subplots within your model that are to be plotted in the same frame. The *subplot_list* allows you to define any number of subsets of your model and define different characteristics for each of them.

You may use one or more of the **SET**s you have defined in your **PLOT** command. The syntax of the *subplot_list* is:

```
subplot_list ⇒ subplot_term,subplot_term,...
```

Each *subplot_term* in the *subplot_list* selects a **SET** and defines plotting options for it. The general form of the *subplot_term* is:

```
subplot_term ⇒ SET setid [set_options]
```

The *set_options* are similar to the *global_options*, but they only apply to the previously appearing **SET**. Examples of **PLOT** commands using *subplot_lists* are shown below:

1. Create a plot which includes **SET**s **1**, **2**, and **3**, labels all of the GRID points and elements:

   ```
   PLOT LABEL BOTH SET 1, SET 2, SET 3
   ```

2. Using the previous example, place symbol **6** at the GRID point locations in **SET 2**:

   ```
   PLOT LABEL BOTH SET 1, SET 2 SYMBOL 6, SET 3
   ```

3. Plot **SET**s **100**, **200** and **300** using line styles **1**, **2**, and **3**, respectively. Label the GRID points in **SET 100**, the elements in **SET 200**, and place symbol **4** at the GRID point locations in **SET 300**:

   ```
   PLOT SET 1 LINESTYLE 1,LABEL GRID POINTS,
        SET 2 LINESTYLE 2,LABEL ELEMENTS,
        SET 3 LINESTYLE 3,SYMBOL 4
   ```

5.6 PLOTTING SOLUTION RESULTS

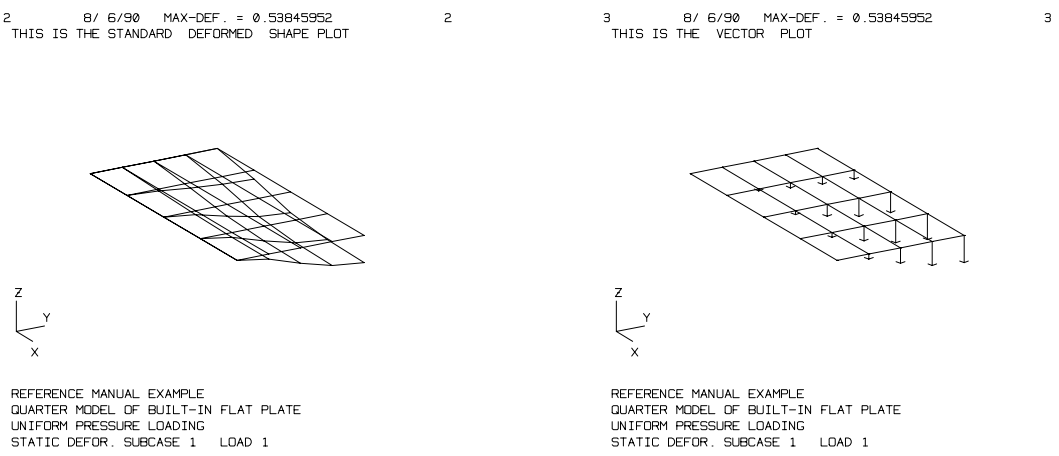
You may also plot geometry-based solution results using the Structural Plotter. Such plots are called ***deformed shapes*** of the structural model. In this case, deformations include the grid point displacements and, for dynamic response analyses, the velocities and accelerations. The deformed plots may be made for selected subcases, time ranges, or frequency ranges, depending on the solution discipline that you are using. The general syntax of the **PLOT** command, when used for solution results, is:

```
PLOT results_type [UNDEFORMED][subcase_list][solution_range]
     [plot_type][global_options] [subplot_list]
```

This form of the **PLOT** command is quite different from that used to plot the structural model. First, you must select the *results_type* to be plotted. This may be displacements, velocities or accelerations depending on the solution discipline that you are using. You may choose to plot the **UNDEFORMED** shape as well as the deformed shape. Then, again depending on the analysis discipline, you may plot the solution results for selected subcases or a *solution_range* of times or frequencies. You also select a *plot_type*. The two types are **DEFORMED**, which draws the deformed structural model, and **VECTOR**, which represents the deformation as a vector. Examples of these two plots are shown in Figure 5-6. In both cases, the **UNDEFORMED** shape has also been plotted. The *subplot_list* differs from that used for model plotting in that each **SET** may specify a different *plot_type*.

5 PLOT

Figure 5-6. SOLUTION RESULTS PLOTS



The following examples illustrate **PLOT** commands for solution results:

1. Plot the static displacements for **SUBCASE**s **1**,**5** and **9** for the entire model:

```
PLOT STATIC DEFORMATION 1,5,9
```

2. Plot the transient accelerations for a portion of the model defined by **SET 10** as **VECTOR**s representing the **X**, **Y**, and **Z** components of motion over the time interval **0** to **5** seconds:

```
PLOT TRANSIENT ACCELERATION TIME 0.,5. VECTOR XYZ SET 10
```

3. Repeat example 2, but suppose the model was partitioned in two sets, **SET 100** and **SET 200**, and that **SET 100** will be plotted as a **DEFORMED** shape and **SET 200** as **VECTOR**s:

```
PLOT TRANSIENT ACCELERATION TIME 0.,5. ,
     SET 100 DEFORMED
     SET 200 VECTOR XYZ
```

5.7 ASSIGNING FILES

When you use the Structural Plotter, you must use the **ASSIGN** Executive Control command to specify the file which will contain the plotting data. This file must have a **USE=PLOT** parameter specified. The general **ASSIGN** command for this purpose is:

```
ASSIGN logical_name=phys_name,NEW,USE=PLOT
```

Often the *logical_name* may be **PLOT** in which case the **USE** parameter is not necessary. The plot file may be written using **TYPE=BINARY**, which is the default, or using **TYPE=FORMATTED**. You must contact your **UAI/*NASTRAN*** Systems Support Specialist to determine which **TYPE** is required by your UAIPLOT post-processor.

5.8 DEVICE COMMANDS

There are four commands that you use to define the characteristics of the graphics device that will be used to create your plots using the UAIPLOT program or an alternate program developed at your site. Consult your **UAI/*NASTRAN*** Systems Support Specialist to get information on these parameters. The post-processor device-related commands are shown in Table 5-1.

5.9 STRUCTURAL PLOTTER COMMAND DESCRIPTIONS

Structural Plotter commands are free-field entries. In presenting general formats for each command and its options, the following conventions are used:

- Many Structural Plotter commands contain ***keywords***. All keywords are shown in capitalized, bold computer type such as: **HARDWARE**. All such keywords must be entered exactly as they are specified, subject only to the exception that they may be abbreviated by their first four characters.
- Some commands contain ***parentheses***. These must be entered if an option requiring them is selected.
- Lower case italicized computer type, such as *results_type*, indicates that you must provide a specific data value.
- Braces { } enclose a list of two or more options from which you may select one.
- Brackets [ ] indicate that the enclosed keywords and parameters may be omitted when you use the command.
- When a choice is to be made from a list, the default choice is presented in boldface type as in the following: { CHOICE1 / CHOICE2 / **DEFAULT** }
- A single command line may not exceed 80 characters. However, you may continue a command by ending the current record with a comma and continuing to the next record. Titling commands may not be continued from one record to the next.

5 PLOT

Additionally, command options have a valid data range and, in some cases, a default value. The following table defines the data range specifications found in this chapter.

DATA RANGE SPECIFIER	MEANING
Integer	The data must be an integer number in the range of indicated values.
Real	The data must be a real number in the range of indicated values.
Keyword	The data must be a character string matching the first four or more unique characters of a keyword option.

Plot Command ASPECT RATIO

Specifies the relative shape of the plotting surface used by the plotting post-processor.

Command Syntax:

```
ASPECT RATIO = ratio
```

Option	*Meaning*	*Data Range*
ratio	Specifies the ratio of the width of the plotting surface to its height. [1,2,3]	Real>0.0

Remarks:

1. If you specify an **ASPECT RATIO**, it will be used for all subsequent plots.
2. If you do not use this command, an **ASPECT RATIO** of **1.0** is used.
3. The results of the **ASPECT RATIO** command will vary depending on your plotting program. Contact your **UAI/*NASTRAN*** System Support Specialist for additional information.

Plot Command **AXES**

Specifies the orientation of the **XYZ** coordinate system of the model to the RST coordinate system of the plotter.

Command Syntax:

AXES *R*, *S*, *T*

Option	*Meaning*	*Data Range*
R, S, T	Defines the plot axes relative to the model axes. [1]	Character { **X**, **Y**, **X**, **MX**, **MY**, **MZ** }

Remarks:

1. The RST values are replaced by a unique combination of the coordinate directions **X**, **Y**, **Z**, **MX**, **MY**, and **MZ**. The default condition is:

   AXES X Y Z

   The prefix **M** inicates the negative axis.
2. This command is equivalent to changing the **VIEW**ing angles by increments of 90°.
3. See the introduction to this chapter for a discussion of the plotter coordinate system.

Example:

1. Define your plot axes such that **Z**, **X**, and **Y** represent the plot RST system:

   AXES Z X Y

Plot Command CHARACTER PRECISION

Specifies the type of characters used on your plots

Command Syntax:

```
CHARACTER PRECISION = { HARDWARE }
                      { STROKED  }
```

Option	*Meaning*	*Data Range*
{ HARDWARE } { STROKED }	Specifies that characters will be drawn automatically by your plotting program, **HARDWARE**, or that characters will be created in graphical form for use by your plotting program, **STROKED**. [1,2]	Keyword

Remarks:

1. If you do not use this command, a **CHARACTER PRECISION** of **HARDWARE** is used.
2. You select between the **HARDWARE** and **STROKED** options based on the implementation of your plotting program. Contact your **UAI/*NASTRAN*** System Support Specialist for complete information.

Plot Command **CHARACTER SCALE**

Specifies the size of characters used for titling and labeling of plots.

Command Syntax:

```
CHARACTER SCALE = factor
```

Option	*Meaning*	*Data Range*
factor	Selects a scale factor for characters.	Real>0.0

Remarks:

1. If you do not use this command, a **CHARACTER SCALE** of **1.0** is used.
3. If you have specified a **CHARACTER PRECISION** of **HARDWARE**, then the actual characters used on your plots will depend on your plotting program. Contact your **UAI/*NASTRAN*** System Support Specialist for complete information.

Plot Command DEFORMATION SCALE

Defines the scale factor used in plotting the solution deformations.

Command Syntax:

```
DEFORMATION SCALE  δ
```

Option	*Meaning*	*Data Range*
δ	The maximum deformation component scale factor. [1]	Real>0.0

Remarks:

1. The value of δ represents the percentage of the structural model size to which the maximum deformation component will be scaled for each subcase. The default value is 5%.

Plot Command **PEN OPTIMIZATION**

Enables or disables pen movement optimization.

Command Syntax:

```
PEN OPTIMIZATION { ON  }
                 { OFF }
```

Option	*Meaning*	*Data Range*
{ ON } { OFF }	Enables or disables pen movement optimization. [1,2]	Keyword

Remarks:

1. **PEN OPTIMIZATION** is **OFF** by default.
2. This option should only be enabled if your graphics device draws plots with a pen. If your device is terminal or a laser printer, then this option will result in wasted CPU utilization which will not help you in any way.

Plot Command PERSPECTIVE FACTOR

Specifies the amount of perspective.

Command Syntax:

```
PERSPECTIVE FACTOR factor
```

Option	*Meaning*	*Data Range*
factor	Specifies the perspective factor. [1]	99.0>Real>0.0

Remarks:

1. The **PERSPECTIVE FACTOR** determines the relative size of a portion of the model in the foreground with a similar portion in the background of the plot. The amount of perspective desired is a matter of taste. **UAI/*NASTRAN*** automatically determines the perspective when you do not use this command. Typically values of *factor* between **10.0** and **50.0** give best results. As the value gets higher, you may obtain unexpected results because you have moved inside your model, or you may obtain severly distorted plots.
2. **PERSPECTIVE FACTOR** is only used when you have selected a **PERSPECTIVE PROJECTION**.

Plot Command PLOT [Model]

Requests the creation of a plot of your finite element model.

Command Syntax:

PLOT [*line_style*][*labels*][*symbols*][*subplot_list*]

Option	*Meaning*	*Data Range*
line_style	Specifies an overall line style to be used for the plots. [1]	Integer>0
labels	Specifies the overall labelling option for the plots [2]	Keyword
symbols	Selects the overall symbol type to be placed at GRID point locations in the plots. [3]	[3]
subplot_list	Selects one or more plot sets defining subplots. [4]	[4]

Remarks:

1. The *line_style* selects the type of line that will be used for drawing your plot, the syntax of this option is:

   LINESTYLE *line_code*

   where *line_code* is an integer code that defines the style of your plotted lines. The default **LINESTYLE** is **1**. The meaning of this code depends on your Structural Plotter program. For example, it might indicate lines of different thickness, dash patterns, or colors. Contact your **UAI/*NASTRAN*** System Support Specialist for detailed information.

2. The *labels* option selects the type of labels that you wish placed on your plot. The syntax of this option is:

   LABEL { GRID POINTS | ELEMENTS | BOTH }

   No labels will appear on your plot unless you request them. Note that if you select **LABEL** without an additional selection, then **GRID POINTS** will be labelled.

5 PLOT

3. The *symbols* option allows you to specify a symbol, or combination of symbols, to be placed at GRID point locations in your plot. Unless you use this option, symbols are not plotted. The syntax of this option is:

   SYMBOL *sym1* [,*sym2*]

   The table below defines the codes that you use to select symbols. If you specify two symbols, then the symbols having the specified codes are superimposed at the same location.

symi	SYMBOL	*symi*	SYMBOL
0	None	5	.
1	X	6	O
2	*	7	□
3	+	8	◊
4	-	9	Δ

4. The *subplot_list* allows you to select the parts of a model which will be plotted, and specify separate plotting options for that part. If *subplot_list* is not specified, then the elements contained in the **first SET** defined in the Structural Plotter command subpacket are plotted. If no **SET** command has been specified, then **ALL** of the elements in the model are plotted. See also the *subplot_list* description.

Examples:

1. Plot your complete model, using all plotter defaults:

   PLOT

2. Plot the portion of you model define by **SET 10** using **LINESTYLE 3** and label the GRID points and elements:

   PLOT LINESTYLE 3 LABEL BOTH SET 10

3. Suppose you have defined two portions of your model in **SET 1** and **SET 2**. Request that the plot be drawn using **LINESTYLE 1** for the former and **LINESTYLE 2** for the latter

   PLOT SET 1 LINESTYLE 1 SET 2 LINESTYLE 2

Plot Command PLOT [Solution Results]

Requests a plot of solution results.

Command Syntax:

```
PLOT   results_type [UNDEFORMED][subcase_list][solution_range]
       [cformat][line_style][labels][symbols][plot_type][subplot_list]
```

Option	*Meaning*	*Data Range*
results_type	Selects the specific solution results that you wish to plot. [1,10]	Keyword
UNDEFORMED	Requests that the undeformed shape also be plotted. [8,10]	Keyword
subcase_list	Selects the subcases for which solution results will be plotted. [2,10]	Integer>0
solution_range	Specifies a range of times, frequencies or buckling factors for which results will be plotted. [1,10]	[1]
cformat	Specifies the type of complex output for complex modal or frequency response analyses. [3]	Keyword
line_style	Specifies an overall line style to be used for the plots. [4]	[4]
labels	Specifies the overall labelling option for the plots. [5]	[5]
symbols	Specifies the overall symbol type for GRID points. [6]	[6]
plot_type	Specifies the overall manner in which solution results will be displayed. [7,8]	Keyword
subplot_list	Selects one or more plot sets defining subplots. [9]	[9]

Remarks:

1. The solution results that you may select depend upon the Rigid Format that you have executed. Some of these also allow a *solution_range* to be specified. The table below gives the *results_type* and *solution_range* options and the disciplines for which they are valid:

results_type	*solution_range*	Rigid Formats
STATIC DISPLACEMENT	Not Used	**1,2,5,14**
MODAL DISPLACEMENT	**RANGE** λlow,λhigh	**3,13,15**
BUCKLING DISPLACEMENT	**FACTOR** flow,fhigh	**5**
CMODAL DISPLACEMENT	**RANGE** λlow,λhigh	**7,10**
FREQUENCY DISPLACEMENT	**RANGE** flow,fhigh	**8,11**
FREQUENCY VELOCITY	**RANGE** flow,fhigh	**8,11**
FREQUENCY ACCELERATION	**RANGE** flow,fhigh	**8,11**

results_type	*solution_range*	Rigid Formats
TRANSIENT DISPLACEMENT	**TIME** tlow,thigh	**9,12**
TRANSIENT VELOCITY	**TIME** tlow,thigh	**9,12**
TRANSIENT ACCELERATION	**TIME** tlow,thigh	**9,12**

2. The *subcase_list* is a list of subcase terms which are either the identification numbers for which you wish to plot the selected results or a specification of a range of subcases of the form:

   i THRU *j*

   If the *subcase_list* is not entered, then results for all subcases will be plotted.

3. The *cformat* specifies the manner in which you wish to have complex deformation quantities plotted. The two options and the calculation performed are shown in the table below:

cformat	PLOTTED VALUE	NOTES
PHASE LAG φ	$u = u_r \cos\varphi - u_i \sin\varphi$	The lag angle φ is a real number. Its default value is **0.0**. The tuple (u_r,u_i) represents the real and imaginary parts of the complex solution result whether displacement, velocity, or acceleration.
MAGNITUDE	$u = \sqrt{u_r^2 + u_i^2}$	

4. The *line_style* selects the type of line that will be used for drawing your plot, the syntax of this option is:

   LINESTYLE *line_code*

   where *line_code* is an integer code that defines the style of your plotted lines. The meaning of this code depends on your Structural Plotter program. For example, it might indicate lines of different thickness, dash patterns, or colors. Contact your **UAI/*NASTRAN*** System Support Specialist for detailed information. If not specified, the *line_style* is taken to be **1**.

5. The *labels* option selects the type of labels that you wish placed on your plot. The format of this parameter is:

   LABEL [GRID POINTS | ELEMENTS | BOTH]

   No labels will appear on your plot unless you request them. Note that if you select **LABEL** without an additional selection, then **GRID POINTS** will be labelled.

6. The *symbols* option allows you to specify a symbol, or combination of symbols, to be placed at GRID point locations in your plot. This syntax of this option is:

   SYMBOL *sym1* [,*sym2*]

   The table below defines the codes that you use to select symbols. If you specify two symbols, then the symbols having the specified codes are superimposed at the same location.

symi	**SYMBOL**	*symi*	**SYMBOL**
0	None	5	.
1	X	6	O
2	*	7	□
3	+	8	◊
4	-	9	Δ

7. The *plot_type* selects the manner in which your solution results will be displayed. The options and their meaning is given in the table below:

plot_type	**DESCRIPTION**
DEFORMED	The selected output quantity will be plotted as a deformed shape where the deformations are calculated using the three translational components of motion. For complex results, the deformations are computed based on the *cformat* option.
VECTOR *component*	The selected output quantity is shown as a vector drawn from the undeformed location to the deformed location. You may select individual *component*s, combinations of *component*s or vector sums of *component*s. The allowable comonent selections are: **X** or **Y** or **Z**; **XY** or **XZ** or **YZ**; **XYZ**. If any of these combinations is prefixed by the character **R**, e.g. **RXYZ**, then then resultant, or vector sum, of the listed *component*s is plotted. Note that the sigle letter **R** may also be used to signify **RXYZ**.

   If not specified, the default *plot_type* is **DEFORMED**.

8. When you select the **VECTOR** *plot_type*, the **UNDEFORMED** shape is automatically plotted along with the vectors.

9. The *subplot_list* allows you to select the parts of a model which will be plotted, and specify separate plotting options for that part. If *subplot_list* is not specified, then the elements contained in the **first SET** defined in the Structural Plotter command subpacket are plotted. If no **SET** command has been specified, then **ALL** of the elements in the model are plotted. See also the *subplot_list* description.

10. The first four command parameters ***must*** appear in the order shown, the remaining parameters may appear in any order.

Examples:

1. Plot the deformed and undeformed model, using all plotter defaults, for subcases 1, 2 and 3 in a static analysis:

```
PLOT STATIC DISPLACEMENT UNDEFORMED 1,2,3 DEFORMED
```

2. Create the same plots as in example 1, but have the deformed plots created in **LINESTYLE 2**:

```
PLOT STATIC DISPLACEMENT UNDEFORMED 1,2,3 LINESTYLE 2 DEFORMED
```

3. Plot the accelerations for a transient analysis as deformed plots in the range 0 to 5 seconds.

```
PLOT TRANSIENT ACCELERATION TIME 0,5.0 DEFORMED
```

4. Plot the resultant vector and undeformed shapes for all subcases of a statics analysis placing symbol 6 at each GRID point of the undeformed model and labeling the GRID points:

```
PLOT MODAL DISPLACEMENT SYMBOL 6 LABEL GRID POINTS VECTOR R
```

Plot Command **PROJECTION**

Specifies the graphic projection to be used for plots.

Command Syntax:

{ **ORTHOGRAPHIC** / **PERSPECTIVE** } PROJECTION

Option	*Meaning*	*Data Range*
ORTHOGRAPHIC	Requests an orthographic geometric projection to be used in creating the plot. [1]	Keyword
PERSPECTIVE	Requests a perspective geometric projection to be used in creating the plot. [1]	Keyword

Remarks:

1. See the introduction to this Chapter for a discussion of the graphic projections.

Plot Command PTITLE

Defines a text string which is used as a frame title.

Command Syntax:

```
PTITLE = any_character_string
```

Remarks:

1. The **PTITLE** will appear on all frames plotted. To clear it, you must enter the command without a character string.
2. If your title string is too long to fit along the specified axis, it is truncated on the right.

Plot Command SET

Defines sets of elements that you wish to have plotted.

Command Syntax:

SET *set_id = include_part* [*modifier_part*]

include_part ⇒ { ALL | [INCLUDE] { *element_type* / *element_id* / *elem_range* } , [{ *element_type* / *element_id* / *elem_range* } ,...] }

modifier_part ⇒ { EXCLUDE / EXCEPT / INCLUDE } { *element_type* / *element_id* / *elem_range* } , [{ *element_type* / *element_id* / *elem_range* } ,...]

Option	*Meaning*	*Data Range*
set_id	Assigns an identification number to the set being defined.	Integer>0
element_type	Specifies the name of an element type. [1]	Keyword
element_id	Specifies the identification number of an element.	Integer>0
elem_range	Defines a range of element identification numbers. [2]	[2]
INCLUDE	Indicates that element types or identification numbers are being included in the set.	Keyword
{ EXCLUDE / EXCEPT }	Indicates that element types or identification numbers are being excluded from the set.	Keyword

Remarks:

1. The allowable element types that you may specify are:

AXIF2	FLUID2	QUAD8	TRIA3
AXIF3	FLUID4	ROD	TRIA6
AXIF4	HEXA	SHEAR	TRIAAX
BAR	PENTA	TETRA	TRIARG
BEAM	PILE	TORDRG	TUBE
CONE	PLOTEL	TRAPAX	VISC
CONROD	QUAD4	TRAPRG	

2. An *elem_range* is specified using the form:

   element_id_1 THRU *element_id_2*

   It is not necessary for all of the elements in the specified range to exist.

Plot Command *subplot_list*

Selects one or more subplots, defined by plot sets, and the options for each. Any option specified for a given set overrides the global options specified on the **PLOT** command.

Command Syntax:

subplot_list ⇒ *subplot_term* , *subplot_term* ,...

subplot_term ⇒ **SET** *sid* [*sym*] [*line_style*] [*labels*] [*symbols*] [*plot_type*]

Option	*Meaning*	*Data Range*
sid	Selects a previously defined element **SET**.	Integer>0
sym	Specifies a symmetry type for the selected **SET**. [1]	Keyword
line_style	Specifies a line style to be used for the selected **SET**. [2]	[2]
labels	Specifies the overall labelling option for the selected **SET**. [3]	[3]
symbols	Selects the type of symbol to be placed at grid point locations in the selected **SET**. [4]	[4]
plot_type	Specifies the manner in which solution results will be displayed for the selected **SET**. [5]	Keyword

Remarks:

1. The *sym* option allows you to request the plotting of a subplot **SET** as a symmetric reflection of its basic geometry by specifying:

   SYMMETRY *plane*

   where *plane* refers to the plane of symmetry. You may specify directions **X**, **Y** or **Z** or combinations of any two of these directions. If you are using a *plot_type* of **DEFORMED**, then the deformations will also be reflected. A second option allows you to display the antisymmetric deformations on the symmetrically reflected **SET**. This is done by specifying:

   ANTISYMMETRY *plane*

   Reiterating, the model is now plotted with the **SYMMETRY** specified by *plane*, but the deformations are **not** reflected.

2. The *line_style* selects the type of line that will be used for drawing your plot, the syntax of this option is:

   LINESTYLE *line_code*

   where *line_code* is an integer code that defines the style of your plotted lines. If you do not specify a *line_code*, the the **PLOT** command global option will be used. If this option has not been set, then *line_code* is set to **1**. The meaning of this code depends on your Structural Plotter program. For example, it might indicate lines of different thickness, dash patterns, or colors. Contact your **UAI/*NASTRAN*** System Support Specialist for detailed information.

5 PLOT

3. The *labels* option selects the type of labels that you wish placed on your plot. The format of this parameter is:

$$\text{LABEL} \left[\begin{array}{l} \text{GRID POINTS} \\ \text{ELEMENTS} \\ \text{BOTH} \end{array} \right]$$

If you do not specify a *labels* option, then the **PLOT** command global option will be used. If this option has not been set, then *labels* will not appear on your plot. Note that if you select **LABEL** without an additional selection, then **GRID POINTS** will be labelled.

4. The *symbols* option allows you to specify a symbol, or combination of symbols, to be placed at GRID point locations in your plot. This syntax of this option is:

SYMBOL *sym1* [,*sym2*]

The table below defines the codes that you use to select symbols. If you specify two symbols, then the symbols having the specified codes are superimposed at the same location.

symi	**SYMBOL**	*symi*	**SYMBOL**
0	None	5	.
1	X	6	O
2	*	7	□
3	+	8	◊
4	-	9	Δ

If you do not specify *symbols* for a **SET**, then the **PLOT** command global option will be used. If this option has not been set, then *symbols* are not plotted.

5. The *plot_type* selects the manner in which your solution results will be displayed. The options and their meaning is given in the table below:

plot_type	**DESCRIPTION**
DEFORMED	The selected output quantity will be plotted as a deformed shape where the deformations are calculated using the three translational components of motion. For complex results, the deformations are computed based on the *cformat* option.
VECTOR *component*	The selected output quantity is shown as a vector drawn from the undeformed location to the deformed location. You may select individual *component*s, combinations of *component*s or vector sums of *component*s. The allowable comonent selections are: **X** or **Y** or **Z**; **XY** or **XZ** or **YZ**; **XYZ**. If any of these combinations is prefixed by the character **R**, e.g. **RXYZ**, then then resultant, or vector sum, of the listed *component*s is plotted. Note that the sigle letter **R** may also be used to signify **RXYZ**. resultant, or vector sum, of the listed components is plotted.

If you do not specify a *plot_type*, then the gobal option specified on the **PLOT** command is used. If this option has not been set, then a **DEFORMED** plot is made.

Plot Command VANTAGE POINT

Defines your location with respect to the the structural model.

Command Syntax:

```
VANTAGE POINT  rv , sv , tv
```

Option	*Meaning*	*Data Range*
rv,sv,tv	Specifies your coordinates in the plotter coordinate system. [1,2,3]	Real

Remarks:

1. See the introduction of this chapter for a discussion of the plotter coordinate system.
2. If you are using the **PERSPECTIVE PROJECTION**, and you have not specified a **PERSPECTIVE FACTOR**, then the **VANTAGE POINT** is used to determine your distance from the model which, in turn, determines the amount of perspective.
3. All model rotations specified with **AXES** or **VIEW** commands are performed before the **VANTAGE POINT** command is applied.

Plot Commands **VIEW**

Rotates the structural model in the plotter coordinate system.

Command Syntax:

VIEW γ , β , α

Option	*Meaning*	*Data Range*
γ, β, α	Angle of rotation, in degrees, about the **T**, **S** and **R** plot axes, respectively.	Real

Remarks:

1. See the introduction of this chapter for a discussion of the plotter coordinate system.
2. The rotations are performed in the order indicated.
3. The default rotation angles are **(34.27,23.17,0.0)**.

Plot Command ZOOM

Enlarges the structural model about the plot origin or an optional specified point.

Command Syntax:

```
ZOOM [ factor [ spos , tpos] ]
```

Option	*Meaning*	*Data Range*
factor	**ZOOM** factor. [1,2]	Real>0.0
spos, tpos	Specifies the point, in unit plotter coordinates, which will be the center of the **ZOOM**ed view. [3]	0.0<Real<1.0

Remarks:

1. **ZOOM** factors greater than 1.0 enlarge your plot, while those less than 1.0 reduce its size. If you enter a negative **ZOOM** factor, it will be ignored.
2. If you specify a **ZOOM** factor and center of zoom, the command will apply to all subsequent plots until you clear the command by entering:

   ZOOM
3. You may **ZOOM** about any point in the plotting area by specifying the center of **ZOOM** in unit plotter coordinates which have a range of 0.0 to 1.0. If you enter a negative value, it will be set to 0.0 and if you enter a value greater than 1.0, it will be set to 1.0.

This page is intentionally blank.

Chapter 6

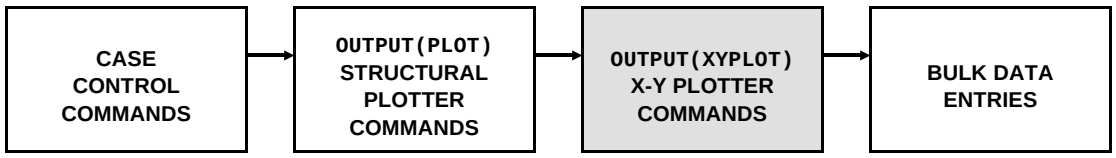
X-Y PLOTTER COMMANDS

X-Y Plotter commands are used to define graph type plots that you wish to generate. Such plots typically graph a structural response quantity such as an acceleration or displacement versus either a frequency range or time domain. X-Y Plotter capability in **UAI/*NASTRAN*** may also be used to plot structural response for nonlinear load step number or even the case of structural response versus applied load.

As in the case of Structural Plots, the actual X-Y Plots are not created during your **UAI/*NASTRAN*** execution. Rather, the plot information is placed on a file that is post-processed. UAI provides four utilities for this purpose. The first two programs, available on most computers, are ***nastplotps*** and ***nastplotgl***. The first of these utilities allows you to send your plots to a postscript laser printer, and the second to a Hewlett-Packard compatible plotting device. The third program, available on Unix host computers is ***nastplot***. This is an interactive program, based on Motif and X-Windows, which allows you to selectively view your plots. The Chapter 1 of this manual provides instructions on how to use these programs for most systems. There is also a program, provided as source code, illustrating how your facility may create their own plotter software, if necessary. In some cases, you must contact your **UAI/*NASTRAN*** Support Specialist for details describing how your site interfaces with these post-processor.

The X-Y Plotter commands form a subpacket within the Case Control packet. The location of this subpacket is shown in Figure 6.1.

Figure 6-1. LOCATION OF THE X-Y PLOTTER SUBPACKET



This packet must begin with the command:

```
OUTPUT(XYPLOT)
```

and ends either when **BEGIN BULK** or a Structural Plotter subpacket is encountered.

6.1 THE X-Y PLOTTER COMMANDS

This section provides you with a summary of the available X-Y Plotter commands. It is recommended that you review the **UAI/*NASTRAN User's Guide*** prior to using this feature.

6.1.1 The X-Y Plotter Functions

The X-Y Plotting feature of **UAI/*NASTRAN*** provides five different output processing options as shown in Table 6-1. In addition to creating graphic output using the **XYPLOT** command, you may also: print the actual data that are to be plotted, using the **XYPRINT** command; you may use **XYPUNCH** to write these data on a punch file; you may request a printed summary of the maximum and minimum values of the selected data using **XYPEAK**; and, finally, if you do not have a graphics device, you may create printer plots using the **XYPAPER** command. Note that you may specify more than one function on a single command so that you may, for example, request **XYPRINT** and **XYPUNCH** for the same data.

Table 6-1. X-Y PLOTTER FUNCTIONS

COMMAND	DESCRIPTION
XYPLOT	Creates X-Y plots for subsequent plotting on a graphic device.
XYPRINT	Prints the data points that will be plotted.
XYPUNCH	Writes the data points on a punch file.
XYPEAK	Prints a summary of the maximum and minimum values of the selected plot data.

6.1.2 ASSIGNing Files

When you use the X-Y Plotter, you must use the **ASSIGN** Executive Control command to specify the file which will contain the plotting data. This file must have a **USE=PLOT** parameter specified. The general **ASSIGN** command for this purpose is:

```
ASSIGN logical_name=phys_name,NEW,USE=PLOT
```

Similarly, if you select the **XYPUNCH** command, you must **ASSIGN** a file:

```
ASSIGN logical_name=phys_name,NEW,USE=PUNCH
```

Often, in both of the above cases the *logical_name* is either **PLOT** or **PUNCH** as the case dictates, and the **USE** parameter is not necessary. The plot file may be written using **TYPE=BINARY**, which is the default, or using **TYPE=FORMATTED**. You must contact your **UAI/*NASTRAN*** Systems Support Specialist to determine which **TYPE** is required by your UAIPLOT post-processor.

6.1.3 Specifying Plotter Controls

There are four commands that you use to define the characteristics of the graphics device that will be used to create your plots using the UAIPLOT program or an alternate program developed at your site. Consult your **UAI/*NASTRAN*** Systems Support Specialist to get information on these parameters. The post-processor device-related commands are shown in Table 6-2.

Table 6-2. X-Y PLOTTER COMMANDS FOR POST-PROCESSORS

COMMAND	DESCRIPTION
ASPECT RATIO	Maps plotter coordinates to the physical plotting surface.
CHARACTER PRECISION	Selects the manner in which characters are plotted by the post-processor.
CHARACTER SCALE	Specifies a character scaling factor which depends on the size of the plot surface used by the post-processor.
LINESTYLE	Selects a line style.

6.1.4 The Plot Elements

You have considerable flexibility in specifying the data to be plotted and the format of the resulting plots. The various formatting options are called plot elements. Table 6-3 summarizes the available plot elements. Figure 6-2 illustrates a typical whole frame plot with plot elements annotated. Figure 6-3 similarly shows two half frame plots. The default values and interactions between plot element commands is detailed in Section 6.7.

6.1.5 Plot Titling

You may define titles for each plot frame and for all of the axes within each plot. The commands used to do this are summarized in Table 6-4.

6.1.6 Data Scaling

You may control the scaling of data by selecting whether plots will be drawn using linear or logarithmic scales, the range of X-axis values and Y-axis values for which plots may be made, and so forth. A summary of the available commands which perform these functions is given in Table 6-5.

Table 6-3. X-Y PLOTTER COMMANDS FOR PLOT ELEMENTS

COMMAND	DESCRIPTION
XAXIS, YAXIS	Enables or disables drawing of the X-Axis or Y-Axis.
XTAXIS, XBAXIS	Enables or disables drawing of the X-Axis on top and bottom half frame plots.
XGRID, YGRID	Enables or disables drawing of grid lines parallel to the X-Axis or Y-Axis.
XTGRID, XBGRID	Enables or disables drawing of grid lines parallel to the X-Axis on top or bottom half frame plots.
YTGRID, YBGRID	Enables or disables drawing of grid lines parallel to the Y-Axis on top or bottom half frame plots.
DRAW LINE	Enables or disables the drawing of line segments between data points on the plot.
SYMBOL	Enables or disables the plotting of graphic symbols at each data point and selects the symbol and increment at which they are drawn.
UPPER SCALES LOWER SCALES	Enables or disables the drawing of tic marks and values along the upper and lower frame lines of the plot.
LEFT SCALES RIGHT SCALES	Enables or disables the drawing of tic marks and values along the left and right frame line of the plot.
TLEFT SCALES TRIGHT SCALES	Enables or disables the drawing of tic marks and values along the left and right frame line of top half frame plots.
BLEFT SCALES BRIGHT SCALES	Enables or disables the drawing of tic marks and values along the left and right frame line of bottom half frame plots.
XVALUE SCALES YVALUE SCALES	Specifies the number of tic marks to skip between printed values on the X-Axis or Y-Axis.
YTVALUE SCALES YBVALUE SCALES	Specifies the number of tic marks to skip between printed values on the Y-Axis of top and bottom half frame plots.
CLEAR	Returns selected plot elements to their default values.

Table 6-4. X-Y PLOTTER COMMANDS FOR TITLING

COMMAND	DESCRIPTION
FTITLE	Specifies a title for a plot frame.
XTITLE YTITLE	Specifies a title to be placed on the X-Axis or the Y-Axis.
YTTITLE YBTITLE	Specifies a title to be placed on the Y-Axis of top or bottom half frame plots.

Table 6-5. X-Y PLOTTER COMMANDS FOR DATA SCALING

COMMAND	DESCRIPTION
XMIN, XMAX	Specifies the minimum and maximum range of X-Values that will be plotted.
YMIN, YMAX	Specifies the minimum and maximum range of Y-Values that will be plotted.
YTMIN, YTMAX	Specifies the minimum and maximum range of Y-Values that will be plotted on the top half frame plot.
YBMIN, YBMAX	Specifies the minimum and maximum range of Y-Values that will be plotted on the bottom half frame plot.
XINTERCEPT YINTERCEPT	Specifies the X- and Y-Values at which requested axes will intersect.
YTINTERCEPT YBINTERCEPT	Specifies the Y-Value at which requested axes will intersect on top or bottom half frame plots.
XLOG	Requests that a logarithmic scale be used for the X-Values
YLOG	Requests that a logarithmic scale be used for the Y-Values
YTLOG, YBLOG	Requests that a logarithmic scale be used for the Y-Values on the top or bottom half frame plots.
XDIVISIONS	Specifies the approximate number of labeled divisions along the X-Axis.
YDIVISIONS	Specifies the approximate number of labeled divisions along the Y-Axis.
YTDIVISIONS YBDIVISIONS	Specifies the approximate number of labeled divisions along the Y-Axis on the top or bottom half frame plots.

Figure 6-2. PLOT ELEMENTS FOR WHOLE FRAMES

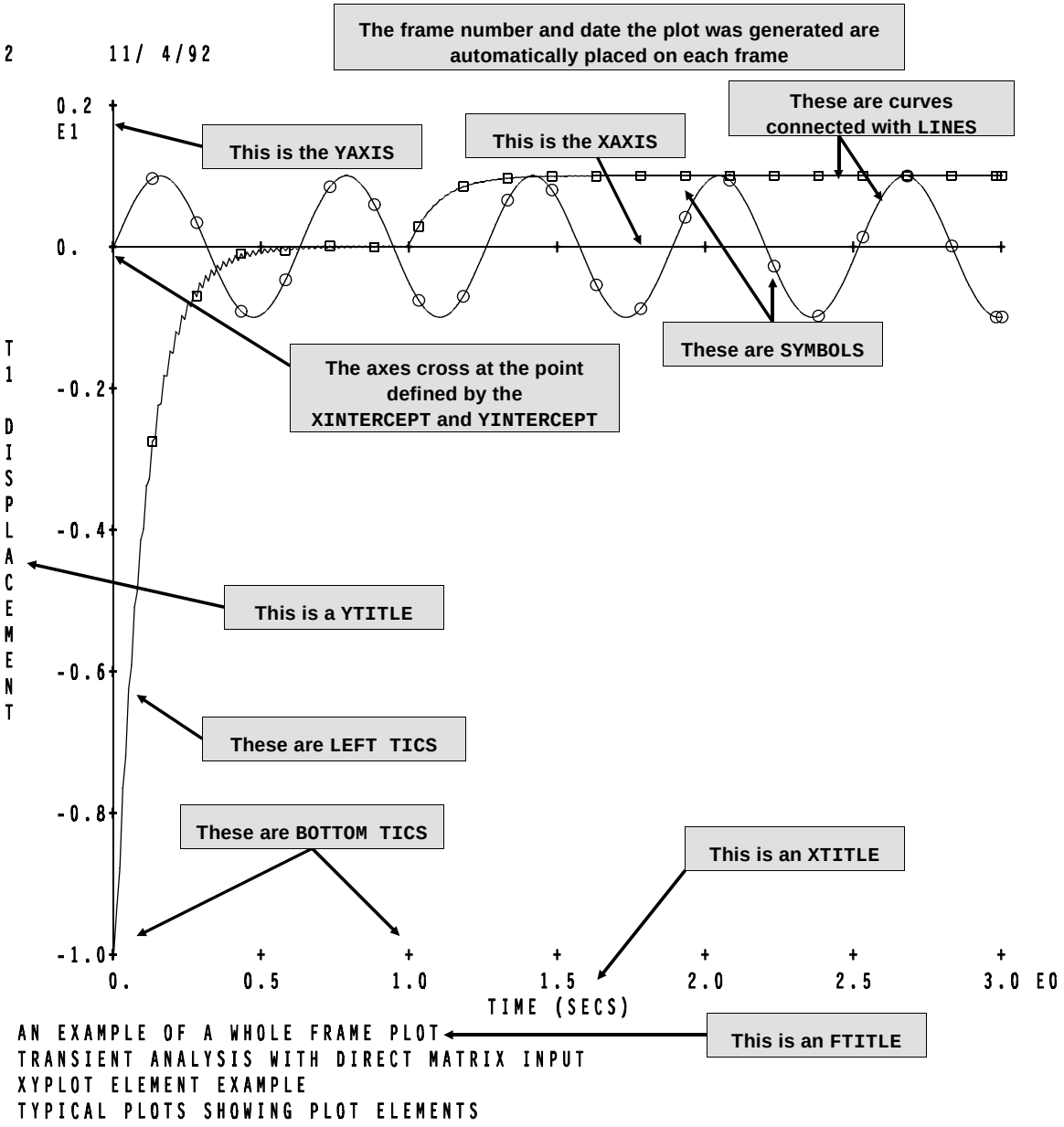
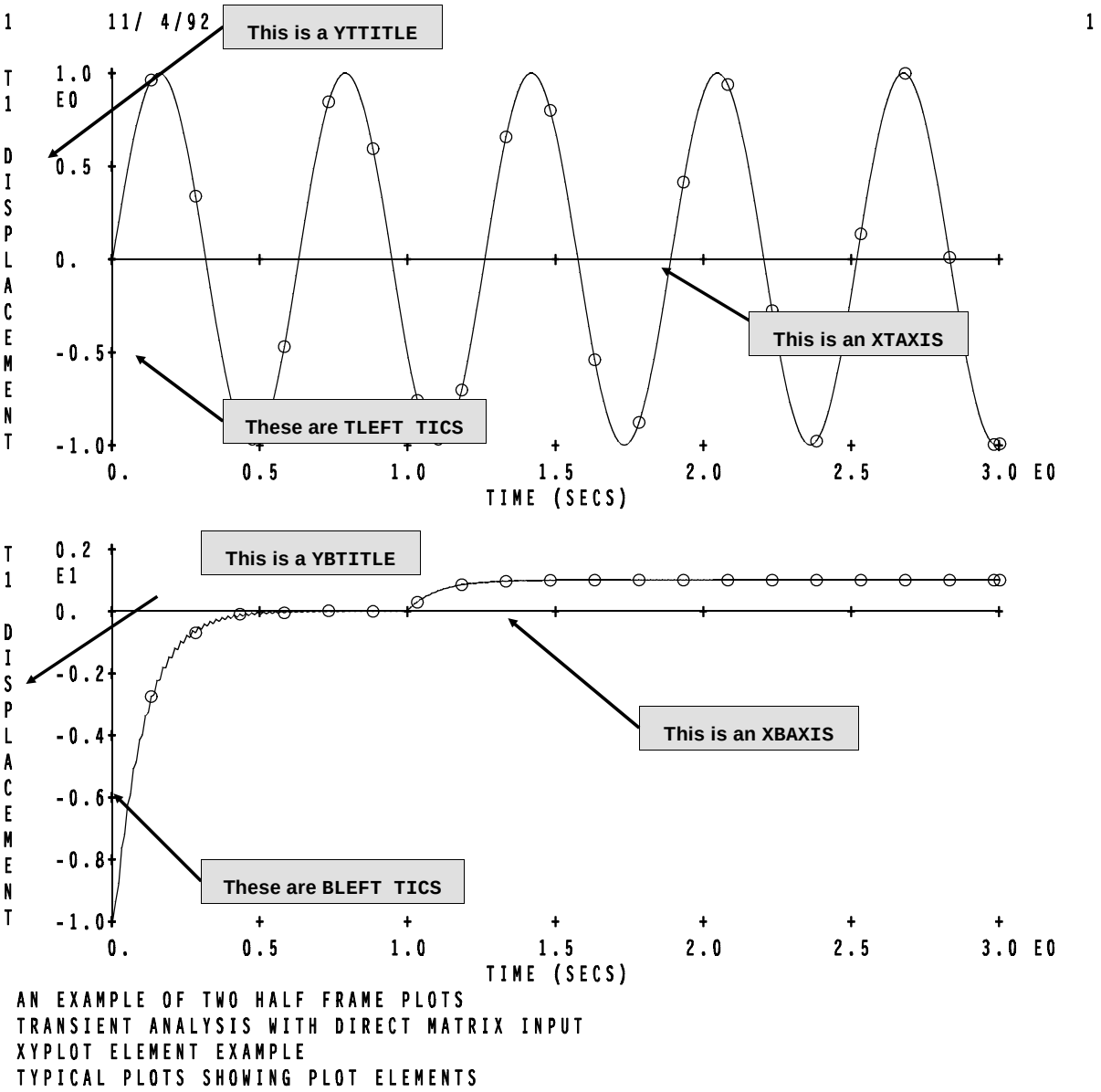


Figure 6-3. PLOT ELEMENTS FOR HALF FRAMES



6 XYPLOT

6.1.7 Selecting SUBCASES

When you create response output using either ***History*** or ***Correlation*** plots, you may specify an optional list of SUBCASE identification numbers for which the plots will be created. If you do not provide the identification numbers, then the specified plots will be created for ***all*** SUBCASEs. The specification of this list is most useful in the dynamic response disciplines where it provides the time or frequency histories of responses.

6.1.8 Defining Frames and Curves

X-Y Plotter commands act on a *frame_list* which defines the exact plots that you will create. The syntax of the *frame_list* differs depending on whether you are creating full or half frame plots and on whether you are requesting ***History*** or ***Correlation*** plots. History plots are a typical grid or element quantity response plotted versus either SUBCASE, time or frequency, depending on the type of analysis performed. Correlation plots are plots of one response quantity versus another response or applied load quantity.

Specific differences are described later under details for they **XYPLOT** command. However, the general form is:

```
/ frame_1 / frame_2 / ...
```

Each frame may contain many curves which may be directed to full or half frames. The curves are defined within the slashes and separated by commas:

```
curve_1,curve_2,...
```

The following sections descibe the various forms used for defining curves.

Whole Frame Plots

When you request whole frame plots, each curve has the form:

```
grid_id_1(resp_code_1)
or
elem_id_1(resp_code_1)
```

where each *grid_id* or *elem_id* specifies a GRID point or element from which response data will be obtained. The *resp_code*s are a response code that differs depending on the response type selected in the plot. The number of curves drawn in the frame is equal to the number of terms in the list.

Half Frame Plots

When you request half frame plots, each frame has a slightly different form:

```
grid_id(top_resp_code,bot_resp_code)
or
elem_id(top_resp_code,bot_resp_code)
```

where, as before, the *grid_id* or *elem_id* specifies the GRID point or element from which response data will be obtained. Now, however, there are two *resp_code*s; one that will be plotted on the top half frame, and one that will be printed on the bottom half frame. If some GRID point or element responses are not to be plotted in one of the half frames, then the *resp_code* is omitted as in:

```
101(T1,),201(T2,T3)
or
55(,3),56(3,7)
```

Note that the comma must always be present in the *resp_code* list.

CORRELATION Plots

Recall that when you create correlation plots you are plotting one response quantity against another. Therefore, you must specify the X-axis and Y-axis response quantities. This is done using:

```
grid_or_elem_id_1(top_resp_code_1,bot_resp_code_1) VS
grid_or_elem_id_2(top_resp_code_2,bot_resp_code_2)
```


The *resp_code*s must match the type of the response that you requested for the X-axis and Y-axis in the **XYPLOT** command. For eaxmple, to plot the **T1** displacement of **GRID 101** against the **T2** displacement of **GRID 201**, you could request:

```
XYPLOT CORRELATION DISP VS DISP / 101(T1) VS 201(T1)
```

All rules regarding whole and half frame plots also apply to the **CORRELATION** plots.

6.2 SOLUTION RESPONSE CODES

The solution response codes, including element stresses and strains and element forces quantities that may be plotted are shown in tables following.

When selecting the correct ITEM CODE, it is important to consider the analysis discipline being used. All output in all analyses except the following use the REAL RESULTS codes. Most output in the following analyses is obtained using the COMPLEX RESULTS codes:

- ❒ Direct Complex Eigenvalue Analysis
- ❒ Direct Frequency Response Analysis
- ❒ Modal Complex Eigenvalue Analysis
- ❒ Modal Frequency Response Analysis

However, Power Spectral Density and Autocorrelation function plots, which are available from Random Response calculations performed with the Frequency Response Rigid Formats, are obtained using the *REAL or MAGNITUDE* code portion of the COMPLEX RESULTS codes.

ELEMENT STRESS AND STRAIN ITEM CODES

ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
AXIF2	Radial — Axis		2	2	6
	Axial — Axis		3	3	7
	Tangential — Edge		4	4	8
	Circumferential — Edge		5	5	9
AXIF3	Centroid	Radial stress	2	2	11
		Circumferential stress	3	3	12
		Axial stress	4	4	13
	Edge N N = 1,…,3	Tangential stress	3+2N	3+2N	14+2N
		Circumferential stress	4+2N	4+2N	15+2N
AXIF4	Centroid	Radial stress	2	2	11
		Circumferential stress	3	3	12
		Axial stress	4	4	13
	Edge N N = 1,…,4	Tangential stress	3+2N	3+2N	14+2N
		Circumferential stress	4+2N	4+2N	15+2N

ELEMENT STRESS AND STRAIN ITEM CODES					
ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
BAR	Stress END-A, Recovery point 1		4	4	8
	Stress END-A, Recovery point 2		5	5	9
	Stress END-A, Recovery point 3		6	6	10
	Stress END-A, Recovery point 4		7	7	11
	Maximum stress, END-A		8	–	–
	Minimum stress, END-A		9	–	–
	Tensile margin of safety		10	–	–
	Compressive margin of safety		11	–	–
	Stress END-B, Recovery point 1		14	14	18
	Stress END-B, Recovery point 2		15	15	19
	Stress END-B, Recovery point 3		16	16	20
	Stress END-B, Recovery point 4		17	17	21
	Maximum stress, END-B		18	–	–
	Minimum stress, END-B		19	–	–
BEAM	At Section A	Stress END-A, Recovery point 1	4	4	8
		Stress END-A, Recovery point 2	5	5	9
		Stress END-A, Recovery point 3	6	6	10
		Stress END-A, Recovery point 4	7	7	11
		Maximum stress, END-A	8	–	–
		Minimum stress, END-A	9	–	–
		Tensile margin of safety	10	–	–
		Compressive margin of safety	11	–	–
	At Section N N = 2,…11	Stress, Recovery point 1	6+8(N-1)	4+10(N-1)	8+10(N-1)
		Stress, Recovery point 2	7+8(N-1)	5+10(N-1)	9+10(N-1)
		Stress, Recovery point 3	8+8(N-1)	6+10(N-1)	10+10(N-1)
		Stress, Recovery point 4	9+8(N-1)	7+10(N-1)	11+10(N-1)
		Maximum stress	10+8(N-1)	–	–
		Minimum stress	11+8(N-1)	–	–

ELEMENT STRESS AND STRAIN ITEM CODES					
ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
CONEAX	At Fiber DIstance Z1	Normal -u	4		
		Normal-v	5		
		Shear-uv	6		
		Shear angle	7		
		Major principal	8		
		Minor principal	9		
		Maximum shear	10		
	At Fiber DIstance Z2	Normal -u	12		
		Normal-v	13		
		Shear-uv	14		
		Shear angle	15		
		Major principal	16		
		Minor principal	17		
		Maximum shear	18		
CONROD	Axial stress		2	2	3
	Axial margin of safety		3	–	–
	Torsional stress		4	4	5
	Torsional margin of safety		5	–	–
ELAS1 ELAS2 ELAS3 ELAS4 ELASNL	Stress		2	2	3

ELEMENT STRESS AND STRAIN ITEM CODES

ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
HEXA	At Center Point	Normal-x	9	9	15
		Shear stress - xy	10	12	18
		Principal stress - A	11	—	—
		Direction cosine - A_x	12	—	—
		Direction cosine - B_x	13	—	—
		Direction cosine - C_x	14	—	—
		Mean pressure	15	—	—
		Octahedral shear stress / Von Mises	16	—	—
		Normal - y	17	10	16
		Shear stress - yz	18	13	19
		Principal stress - B	19	—	—
		Direction cosine - A_y	20	—	—
		Direction cosine - B_y	21	—	—
		Direction cosine - C_y	22	—	—
		Normal stress - z	23	11	17
		Shear stress - zx	24	14	20
		Principal stress - C	25	—	—
		Direction cosine - A_z	26	—	—
		Direction cosine - B_z	27	—	—
		Direction cosine - C_z	28	—	—
	At Corner Point N N = 1,…,8	Normal-x	9+21N	9+13N	15+13N
		Shear stress - xy	10+21N	12+13N	18+13N
		Principal stress - A	11+21N	—	—
		Direction cosine - A_x	12+21N	—	—
		Direction cosine - B_x	13+21N	—	—
		Direction cosine - C_x	14+21N	—	—
		Mean pressure	15+21N	—	—
		Octahedral shear stress / Von Mises	16+21N	—	—
		Normal - y	17+21N	10+13N	16+13N
		Shear stress - yz	18+21N	13+13N	19+13N
		Principal stress - B	19+21N	—	—
		Direction cosine - A_y	20+21N	—	—
		Direction cosine - B_y	21+21N	—	—
		Direction cosine - C_y	22+21N	—	—
		Normal stress - z	23+21N	11+13N	17+13N
		Shear stress - zx	24+21N	14+13N	20+13N
		Principal stress - C	25+21N	—	—
		Direction cosine - A_z	26+21N	—	—
		Direction cosine - B_z	27+21N	—	—
		Direction cosine - C_z	28+21N	—	—

ELEMENT STRESS AND STRAIN ITEM CODES					
ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
PENTA	At Center Point	Normal-x	9	9	15
		Shear stress - xy	10	12	18
		Principal stress - A	11	—	—
		Direction cosine - A_x	12	—	—
		Direction cosine - B_x	13	—	—
		Direction cosine - C_x	14	—	—
		Mean pressure	15	—	—
		Octahedral shear stress / Von Mises	16	—	—
		Normal - y	17	10	16
		Shear stress - yz	18	13	19
		Principal stress - B	19	—	—
		Direction cosine - A_y	20	—	—
		Direction cosine - B_y	21	—	—
		Direction cosine - C_y	22	—	—
		Normal stress - z	23	11	17
		Shear stress - zx	24	14	20
		Principal stress - C	25	—	—
		Direction cosine - A_z	26	—	—
		Direction cosine - B_z	27	—	—
		Direction cosine - C_z	28	—	—
	At Corner Point N N = 1,…,6	Normal-x	9+21N	9+13N	15+13N
		Shear stress - xy	10+21N	12+13N	18+13N
		Principal stress - A	11+21N	—	—
		Direction cosine - A_x	12+21N	—	—
		Direction cosine - B_x	13+21N	—	—
		Direction cosine - C_x	14+21N	—	—
		Mean pressure	15+21N	—	—
		Octahedral shear stress / Von Mises	16+21N	—	—
		Normal - y	17+21N	10+13N	16+13N
		Shear stress - yz	18+21N	13+13N	19+13N
		Principal stress - B	19+21N	—	—
		Direction cosine - A_y	20+21N	—	—
		Direction cosine - B_y	21+21N	—	—
		Direction cosine - C_y	22+21N	—	—
		Normal stress - z	23+21N	11+13N	17+13N
		Shear stress - zx	24+21N	14+13N	20+13N
		Principal stress - C	25+21N	—	—
		Direction cosine - A_z	26+21N	—	—
		Direction cosine - B_z	27+21N	—	—
		Direction cosine - C_z	28+21N	—	—

ELEMENT STRESS AND STRAIN ITEM CODES

ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
PILE	Stress END-A, Recovery point 1		4	4	8
	Stress END-A, Recovery point 2		5	5	9
	Stress END-A, Recovery point 3		6	6	10
	Stress END-A, Recovery point 4		7	7	11
	Maximum stress, END-A		8	–	–
	Minimum stress, END-A		9	–	–
	Tensile margin of safety		10	–	–
	Compressive margin of safety		11	–	–
	Stress END-B, Recovery point 1		14	14	18
	Stress END-B, Recovery point 2		15	15	19
	Stress END-B, Recovery point 3		16	16	20
	Stress END-B, Recovery point 4		17	17	21
	Maximum stress, END-B		18	–	–
	Minimum stress, END-B		19	–	–
PIPE	Bending stress END-A, Recovery point 1		2	2	7
	Bending stress END-A, Recovery point 2		3	3	8
	Bending stress END-A, Recovery point 3		4	4	9
	Bending stress END-A, Recovery point 4		5	5	10
	Axial stress END-A		6	6	11
	Maximum stress END-A		7	–	–
	Bending stress END-B, Recovery point 1		8	12	17
	Bending stress END-B, Recovery point 2		9	13	18
	Bending stress END-B, Recovery point 3		10	14	19
	Bending stress END-B, Recovery point 4		11	15	20
	Axial stress END-B		12	16	21
	Maximum stress END-B		13	–	–
QUAD4 QUADR	At Fiber Distance Z1	Normal-x	4	4	5
		Normal-y	5	6	7
		Shear-xy	6	8	9
		Shear angle	7	–	–
		Major principal	8	–	–
		Minor principal	9	–	–
		Maximum shear / Von Mises	10	–	–
	At Fiber Distance Z2	Normal-x	12	11	12
		Normal-y	13	13	14
		Shear-xy	14	15	16
		Shear angle	15	–	–
		Major principal	16	–	–
		Minor principal	17	–	–
		Maximum shear / Von Mises	18	–	–

ELEMENT STRESS AND STRAIN ITEM CODES						
ELEMENT	STRESS OR STRAIN COMPONENT			ITEM CODES		
				REAL RESULTS	COMPLEX RESULTS	
				MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
QUAD8	At Center Point	At Fiber Distance Z1	Normal-x	4	4	5
			Normal-y	5	6	7
			Shear-xy	6	8	9
			Shear angle	7	–	–
			Major principal	8	–	–
			Minor principal	9	–	–
			Maximum shear / Von Mises	10	–	–
		At Fiber Distance Z2	Normal-x	12	11	12
			Normal-y	13	13	14
			Shear-xy	14	15	16
			Shear angle	15	–	–
			Major principal	16	–	–
			Minor principal	17	–	–
			Maximum shear / Von Mises	18	–	–
	At Corner Point N N=1,...4	At Fiber Distance Z1	Normal-x	4+17N	4+15N	5+15N
			Normal-y	5+17N	6+15N	7+15N
			Shear-xy	6+17N	8+15N	9+15N
			Shear angle	7+17N	–	–
			Major principal	8+17N	–	–
			Minor principal	9+17N	–	–
			Maximum shear / Von Mises	10+17N	–	–
		At Fiber Distance Z2	Normal-x	12+17N	11+15N	12+15N
			Normal-y	13+17N	13+15N	14+15N
			Shear-xy	14+17N	15+15N	16+15N
			Shear angle	15+17N	–	–
			Major principal	16+17N	–	–
			Minor principal	17+17N	–	–
			Maximum shear / Von Mises	18+17N	–	–
ROD	Axial stress			2	2	3
	Axial margin of safety			3	–	–
	Torsional stress			4	4	5
	Torsional margin of safety			5	–	–
SHEAR	Maximum shear			2	2	3
	Average shear			3	4	5
	Margin of safety			4	–	–

ELEMENT STRESS AND STRAIN ITEM CODES

ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
TETRA	At Center Point	Normal-x	9	9	15
		Shear stress - xy	10	12	18
		Principal stress - A	11	–	–
		Direction cosine - A_x	12	–	–
		Direction cosine - B_x	13	–	–
		Direction cosine - C_x	14	–	–
		Mean pressure	15	–	–
		Octahedral shear stress / Von Mises	16	–	–
		Normal - y	17	10	16
		Shear stress - yz	18	13	19
		Principal stress - B	19	–	–
		Direction cosine - A_y	20	–	–
		Direction cosine - B_y	21	–	–
		Direction cosine - C_y	22	–	–
		Normal stress - z	23	11	17
		Shear stress - zx	24	14	20
		Principal stress - C	25	–	–
		Direction cosine - A_z	26	–	–
		Direction cosine - B_z	27	–	–
		Direction cosine - C_z	28	–	–
	At Corner Point N N = 1,…,4	Normal-x	9+21N	9+13N	15+13N
		Shear stress - xy	10+21N	12+13N	18+13N
		Principal stress - A	11+21N	–	–
		Direction cosine - A_x	12+21N	–	–
		Direction cosine - B_x	13+21N	–	–
		Direction cosine - C_x	14+21N	–	–
		Mean pressure	15+21N	–	–
		Octahedral shear stress / Von Mises	16+21N	–	–
		Normal - y	17+21N	10+13N	16+13N
		Shear stress - yz	18+21N	13+13N	19+13N
		Principal stress - B	19+21N	–	–
		Direction cosine - A_y	20+21N	–	–
		Direction cosine - B_y	21+21N	–	–
		Direction cosine - C_y	22+21N	–	–
		Normal stress - z	23+21N	11+13N	17+13N
		Shear stress - zx	24+21N	14+13N	20+13N
		Principal stress - C	25+21N	–	–
		Direction cosine - A_z	26+21N	–	–
		Direction cosine - B_z	27+21N	–	–
		Direction cosine - C_z	28+21N	–	–

ELEMENT STRESS AND STRAIN ITEM CODES					
ELEMENT	STRESS OR STRAIN COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
TORDRG	At Point N N = 1,...,3	Tangential membrane stress	2+5(N-1)		
		Circumferential membrane stress	3+5(N-1)		
		Tangential flexural stress	4+5(N-1)		
		Circumferential flexural stress	5+5(N-1)		
		Shear force	6+5(N-1)		
TRAPRG	At Point N N = 1,...,5	Radial stress - r	2+4(N-1)		
		Circumferential stress - θ	3+4(N-1)		
		Axial stress - z	4+4(N-1)		
		Shear stress - rz	5+4(N-1)		
TRIA3 TRIAR	At Fiber Distance Z1	Normal-x	4	4	5
		Normal-y	5	6	7
		Shear-xy	6	8	9
		Shear angle	7	–	–
		Major principal	8	–	–
		Minor principal	9	–	–
		Maximum shear / Von Mises	10	–	–
	At Fiber Distance Z2	Normal-x	12	11	12
		Normal-y	13	13	14
		Shear-xy	14	15	16
		Shear angle	15	–	–
		Major principal	16	–	–
		Minor principal	17	–	–
		Maximum shear / Von Mises	18	–	–

ELEMENT STRESS AND STRAIN ITEM CODES						
ELEMENT	STRESS OR STRAIN COMPONENT			ITEM CODES		
				REAL RESULTS	COMPLEX RESULTS	
				MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
TRIA6	At Center Point	At Fiber Distance Z1	Normal-x	4	4	5
			Normal-y	5	6	7
			Shear-xy	6	8	9
			Shear angle	7	–	–
			Major principal	8	–	–
			Minor principal	9	–	–
			Maximum shear / Von Mises	10	–	–
		At Fiber Distance Z2	Normal-x	12	11	12
			Normal-y	13	13	14
			Shear-xy	14	15	16
			Shear angle	15	–	–
			Major principal	16	–	–
			Minor principal	17	–	–
			Maximum shear / Von Mises	18	–	–
	At Corner Point N N=1,..3	At Fiber Distance Z1	Normal-x	4+17N	4+15N	5+15N
			Normal-y	5+17N	6+15N	7+15N
			Shear-xy	6+17N	8+15N	9+15N
			Shear angle	7+17N	–	–
			Major principal	8+17N	–	–
			Minor principal	9+17N	–	–
			Maximum shear / Von Mises	10+17N	–	–
		At Fiber Distance Z2	Normal-x	12+17N	11+15N	12+15N
			Normal-y	13+17N	13+15N	14+15N
			Shear-xy	14+17N	15+15N	16+15N
			Shear angle	15+17N	–	–
			Major principal	16+17N	–	–
			Minor principal	17+17N	–	–
			Maximum shear / Von Mises	18+17N	–	–
TRIAAX	Harmonic or point angle — θ			2		
	Radial stress — r			3		
	Axial stress — z			4		
	Circumferential stress — θ			5		
	Shear stress — zr			6		
	Shear stress — r_θ			7		
	Shear stress — z_θ			8		
TRIARG	Radial stress - r			2		
	Circumferential stress - θ			3		
	Axial stress - z			4		
	Shear stress - rz			5		

6 XYPLOT

ELEMENT STRESS AND STRAIN ITEM CODES				
ELEMENT	STRESS OR STRAIN COMPONENT	ITEM CODES		
		REAL RESULTS	COMPLEX RESULTS	
		MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
TUBE	Axial stress	2	2	3
	Axial margin of safety	3	–	–
	Torsional stress	4	4	5
	Torsional margin of safety	5	–	–
TWIST	Maximum shear	2	2	4
	Average shear	3	3	5
	Margin of safety	4	–	–

ELEMENT FORCE ITEM CODES					
ELEMENT	FORCE COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
BAR	Bending Moment END-A, Plane 1		4	4	10
	Bending Moment END-A, Plane 2		5	5	11
	Shear, Plane 1		6	6	12
	Shear, Plane 2		7	7	13
	Axial Force		8	8	14
	Torque		9	9	15
	Bending Moment END-B, Plane 1		12	18	24
	Bending Moment END-B, Plane 2		13	19	25
BEAM	At Section A	Bending Moment, Plane 1	4	4	11
		Bending Moment, Plane 2	5	5	12
		Shear, Plane 1	6	6	13
		Shear, Plane 2	7	7	14
		Axial Force	8	8	15
		Torque	9	9	16
		Warping Torque	10	10	17
	At Section N N = 2,...11	Bending Moment, Plane 1	4+9(N-1)	4+16(N-1)	11+16(N-1)
		Bending Moment, Plane 2	5+9(N-1)	5+16(N-1)	12+16(N-1)
		Shear, Plane 1	6+9(N-1)	6+16(N-1)	13+16(N-1)
		Shear, Plane 2	7+9(N-1)	7+16(N-1)	14+16(N-1)
		Axial Force	8+9(N-1)	8+16(N-1)	15+16(N-1)
		Torque	9+9(N-1)	9+16(N-1)	16+16(N-1)
		Warping Torque	10+9(N-1)	10+16(N-1)	17+16(N-1)
CONROD	Axial Force		2	2	3
	Torque		3	4	5
ELAS1 ELAS2 ELAS3 ELAS4 ELASNL	Force		2	2	3

ELEMENT FORCE ITEM CODES				
ELEMENT	FORCE COMPONENT	ITEM CODES		
		REAL RESULTS	COMPLEX RESULTS	
		MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
PILE	Bending Moment END-A, Plane 1	4	4	10
	Bending Moment END-A, Plane 2	5	5	11
	Shear, Plane 1	6	6	12
	Shear, Plane 2	7	7	13
	Axial Force	8	8	14
	Torque	9	9	15
	Bending Moment END-B, Plane 1	12	18	24
	Bending Moment END-B, Plane 2	13	19	25
PIPE	Bending Moment - y, END-A	2	2	8
	Bending Moment - z, END-A	3	3	9
	Torque, END-A	4	4	10
	Axial Force, END-A	5	5	11
	Shear - y, END-A	6	6	12
	Shear - z, END-A	7	7	13
	Bending Moment - y, END-B	8	14	20
	Bending Moment - z, END-B	9	15	21
	Torque, END-A	10	16	22
	Axial Force, END-A	11	17	23
	Shear - y, END-A	12	18	24
	Shear - z, END-A	13	19	25
QUAD4 QUADR	In-plane-x	3	3	11
	In-plane-y	4	4	12
	In-plane Shear-xy	5	5	13
	Bending moment - x	6	6	14
	Bending moment - y	7	7	15
	Twist moment - xy	8	8	16
	Transverse shear - zx	9	9	17
	Transverse shear - yz	10	10	18

ELEMENT FORCE ITEM CODES					
ELEMENT	FORCE COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
QUAD8	At Center Point	In-plane-x	3	3	11
		In-plane-y	4	4	12
		In-plane Shear-xy	5	5	13
		Bending moment - x	6	6	14
		Bending moment - y	7	7	15
		Twist moment - xy	8	8	16
		Transverse shear - zx	9	9	17
		Transverse shear - yz	10	10	18
	At Corner Point N N=1,...4	In-plane-x	3+9N	3+17N	11+17N
		In-plane-y	4+9N	4+17N	12+17N
		In-plane Shear-xy	5+9N	5+17N	13+17N
		Bending moment - x	6+9N	6+17N	14+17N
		Bending moment - y	7+9N	7+17N	15+17N
		Twist moment - xy	8+9N	8+17N	16+17N
		Transverse shear - zx	9+9N	9+17N	17+17N
		Transverse shear - yz	10+9N	10+17N	18+17N
ROD	Axial Force		2	2	3
	Torque		3	4	5
SHEAR	Force, points 1,3		2	2	3
	Force, points 2,4		3	4	5
TRAPAX	Harmonic or point angle — θ		2		
	At Point N N=1,...4	Radial force — r	3+3(N-1)		
		Circumferential force — θ	4+3(N-1)		
		Axial force — z	5+3(N-1)		
TRIA3 TRIAR	In-plane-x		3	3	11
	In-plane-y		4	4	12
	In-plane Shear-xy		5	5	13
	Bending moment - x		6	6	14
	Bending moment - y		7	7	15
	Twist moment - xy		8	8	16
	Transverse shear - zx		9	9	17
	Transverse shear - yz		10	10	18

ELEMENT FORCE ITEM CODES					
ELEMENT	FORCE COMPONENT		ITEM CODES		
			REAL RESULTS	COMPLEX RESULTS	
			MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
TRIA6	At Center Point	In-plane-x	3	3	11
		In-plane-y	4	4	12
		In-plane Shear-xy	5	5	13
		Bending moment - x	6	6	14
		Bending moment - y	7	7	15
		Twist moment - xy	8	8	16
		Transverse shear - zx	9	9	17
		Transverse shear - yz	10	10	18
	At Corner Point N N=1,...3	In-plane-x	3+9N	3+17N	11+17N
		In-plane-y	4+9N	4+17N	12+17N
		In-plane Shear-xy	5+9N	5+17N	13+17N
		Bending moment - x	6+9N	6+17N	14+17N
		Bending moment - y	7+9N	7+17N	15+17N
		Twist moment - xy	8+9N	8+17N	16+17N
		Transverse shear - zx	9+9N	9+17N	17+17N
		Transverse shear - yz	10+9N	10+17N	18+17N
TRIAAX	Harmonic or point angle — θ		2		
	At Point N N=1,...3	Radial force — r	3+3(N-1)		
		Circumferential force — θ	4+3(N-1)		
		Axial force — z	5+3(N-1)		
TUBE	Axial Force		2	2	3
	Torque		3	4	5
TWIST	Moment, points 1,3		2	2	4
	Moment, points 2,4		3	3	5

GRID AND SCALAR POINT ITEM CODES				
RESPONSE TYPE	RESPONSE COMPONENT	ITEM CODES		
		REAL RESULTS	COMPLEX RESULTS	
		MAGNITUDE	REAL or MAGNITUDE	IMAGINARY or PHASE
ACCE **DISP** **NONL** **OLOAD** **SACCE** **SDISP** **SPCF** **SVELO** **VELO**	Translational component in direction T1 Translational component in direction T2 Translational component in direction T3 Rotational component about direction T1 Rotational component about direction T2 Rotational component about direction T3	T1 T2 T3 R1 R2 R3	T1RM T2RM T3RM R1RM R2RM R3RM	T1IP T2IP T3IP R1IP R2IP R3IP
PRESS	Pressure RMS value Pressure level in dB Effective pressure level dB(A) Combined pressure level	P1 P2 P3 — —	P1RM P2RM P3RM P4RM P5RM	P1IP P2IP P3IP P4IP —
Note: For SCALAR points, the reponse code **T1** is used.				

6.3 X-Y PLOTTER COMMAND DESCRIPTIONS

X-Y Plotter commands are free-field entries. In presenting general formats for each command and its options, the following conventions are used:

- Many X-Y Plotter commands contain ***keywords***. All keywords are shown in capitalized, bold computer type such as: **HARDWARE**. All such keywords must be entered exactly as they are specified, subject only to the exception that they may be abbreviated by their first four characters.
- Some commands contain ***parentheses***. These must be entered if an option requiring them is selected.
- Lower case italicized computer type, such as *ratio*, indicates that you must provide a specific data value.
- Braces { } enclose a list of two or more options from which you may select one.
- Brackets [ ] indicate that the enclosed keywords and parameters may be omitted when you use the command.
- When a choice is to be made from a list, the default choice is presented in boldface type as in the following: { CHOICE1 / CHOICE2 / **DEFAULT** }
- A single command line may not exceed 72 characters. However, you may continue a command by ending the current record with a comma and continuing to the next record. Titling commands may not be continued from one record to the next.

Additionally, command options have a valid data range and, in some cases, a default value. The following table defines the data range specifications found in this chapter.

DATA RANGE SPECIFIER	MEANING
Integer	The data must be an integer number in the range of indicated values.
Real	The data must be a real number in the range of indicated values.
Keyword	The data must be a character string matching the first four or more unique characters of a keyword option.

This page is intentionally blank.

Plot Command ASPECT RATIO

Specifies the relative shape of the plotting surface used by the plotting post-processor.

Command Syntax:

```
ASPECT RATIO = ratio
```

Option	*Meaning*	*Data Range*
ratio	Specifies the ratio of the width of the plotting surface to its height. [1,2,3]	Real > 0.0

Remarks:

1. If you specify an **ASPECT RATIO**, it will be used for every plot until you explicitly reset the default value by using the **CLEAR ASPECT RATIO** command.
2. If you do not use this command, an **ASPECT RATIO** of **1.0** is used.
3. The results of the **ASPECT RATIO** command will vary depending on your X-Y Plotter post-processor program. Contact your **UAI/*NASTRAN*** System Support Specialist for additional information.

Plot Command Family AXIS

Enables or disables the drawing of grid lines parallel to the specified axes.

Command Syntax:

$$\left\{\begin{array}{l}\textbf{XAXIS}\\\textbf{YAXIS}\\\textbf{XTAXIS}\\\textbf{XBAXIS}\end{array}\right\} = \left\{\begin{array}{l}\texttt{ON}\\\texttt{OFF}\end{array}\right\}$$

Option	*Meaning*	*Data Range*
XAXIS	Operates on the **XAXIS** plot element. If enabled, a solid line is drawn perpendicular representing the X-axis. [1]	Keyword
YAXIS	Operates on the **YAXIS** plot element. If enabled, a solid line is drawn representing the Y-axis. [2]	Keyword
XTAXIS XBAXIS	Operates on the **XTAXIS** or **XBAXIS** plot elements. If enabled, a line is drawn representing the X-axis on either the top, **XTAXIS**, or bottom, **XBAXIS**, half frame plots. [3]	Keyword
{ ON } { OFF }	Enables or disables the selected plot element.	Keyword

Remarks:

1. If you enable or disable the **XAXIS** plot element, then the half frame elements **XTAXIS** and **XBAXIS** are automatically enabled or disabled. You must explicitly change the half frame requests if you want them to be different.
2. If you enable the **YAXIS** plot element, then all full and half frame plots will display the Y-axis until you disable it.
3. The **XAXIS**, **YAXIS**, **XTAXIS** and **XBAXIS** plot elements are **ON** by default.
4. The **XAXIS**, **XTAXIS** and **XBAXIS** are drawn along the line specified by **YINTERCEPT**, **YTINTERCEPT** and **YBINTERCEPT**, respectively. Similarly, the **YAXIS** is drawn along the line specified by the **XINTERCEPT**.

Plot Command CHARACTER PRECISION

Specifies the type of characters used on your plots

Command Syntax:

CHARACTER PRECISION = { HARDWARE / STROKED }

Option	*Meaning*	*Data Range*
{ HARDWARE / STROKED }	Specifies that characters will be drawn automatically by your plotting program, **HARDWARE**, or that characters will be created in graphical form for use by your plotting program, **STROKED**. [1,2,3]	Keyword

Remarks:

1. If you specify a **CHARACTER PRECISION**, it will be used for every plot until you explicitly reset the default value by using the **CLEAR CHARACTER PRECISION** command.
2. If you do not use this command, a **CHARACTER PRECISION** of **HARDWARE** is used.
3. You select between the **HARDWARE** and **STROKED** options based on the implementation of your plotting program. Contact your **UAI/*NASTRAN*** System Support Specialist for complete information.

Plot Command CHARACTER SCALE

Specifies the size of characters used for titling and labeling of plots.

Command Syntax:

```
CHARACTER SCALE = factor
```

Option	*Meaning*	*Data Range*
factor	Selects a scale factor for characters.	Real>0.0

Remarks:

1. If you specify a **CHARACTER SCALE**, it will be used for every plot until you explicitly reset the default value by using the **CLEAR CHARACTER SCALE** command.
2. If you do not use this command, a **CHARACTER SCALE** of **1.0** is used.
3. If you have specified a **CHARACTER PRECISION** of **HARDWARE**, then the actual characters used on your plots will depend on your plotting program. Contact your **UAI*/NASTRAN*** System Support Specialist for complete information.

Plot Command CLEAR

Clears current plot element settings back to their default values or selectively clears titles or data scaling values.

Command Syntax:

CLEAR [*plot_command*]

Option	*Meaning*	*Data Range*
plot_command	Requests that the current settings for the selected titling, plot element or data scaling command be cleared and returned to its default values. [1,2]	Keyword

Remarks:

1. You may **CLEAR** the Plot commands given in the table below:

ASPECT RATIO	XINTERCEPT	YMAX	XTITLE
CHARACTER SCALE	YINTERCEPT	YBMIN	YTITLE
CHAR PRECISION	YBINTERCEPT	YBMAX	YBTITLE
XDIVISIONS	YTINTERCEPT	YTMIN	YTTITLE
YDIVISIONS	XMIN	YTMAX	SYMBOLS
YBDIVISIONS	XMAX	TITLES	
YTDIVISIONS	YMIN	FTITLE	

2. When **CLEAR** is used without specifying a *plot_command*, then **all** of the plot elements are returned to their default values as summarized in the following table.

PLOT COMMAND OR COMMAND FAMILY	PLOT ELEMENT	DEFAULT
ASPECT RATIO	ASPECT RATIO	1.0
AXIS	XAXIS	ON
	YAXIS	ON
	XTAXIS, XBAXIS	XAXIS
CHARACTER	CHARACTER PRECISION	HARDWARE
	CHARACTER SCALE	1.0
DIVISIONS	XDIVISIONS	5[1]
	YDIVISION	5[1]
	YTDIVISIONS, YBDIVISIONS	YDIVISIONS
GRID	XGRID	OFF
	YGRID	OFF
	XTGRID, XBGRID	XGRID
	YTGRID, YBGRID	YGRID
INTERCEPT	XINTERCEPT	0.0[2]
	YINTERCEPT	0.0[2]
	YTINTERCEPT,YBINTERCEPT	YINTERCEPT
DRAW LINE	DRAW LINE	ON
LINE STYLE	LINE STYLE	1
LOG	XLOG	OFF
	YLOG	OFF
	YTLOG, YBLOG	YLOG
MINMAX	XMIN, XMAX	Determined automatically from the data
	YMIN, YMAX	
	YTMIN, YTMAX	YMIN, YMAX
	YBMIN, YBMAX	YMIN, YMAX
SYMBOL	SYMBOL	OFF
SCALES	UPPER SCALES	OFF
	LOWER SCALES	ON
	LEFT SCALES	ON
	RIGHT SCALES	OFF
	TLEFT SCALES, BLEFT SCALES	LEFT SCALES
	TRIGHT SCALES, BRIGHT SCALES	RIGHT SCALES
	XVALUE SCALES, YVALUE SCALES	0
	YTVALUE SCALES, YBVALUE SCALES	0
TITLE	FTITLE	NONE
	XTITLE	NONE
	YTITLE	NONE
	YTTITLE, TBTITLE	YTITLE

1. The number of **DIVISIONS** is approximate. It is adjusted automatically to give rounded values.

2. The **XINTERCEPT** and **YINTERCEPT** are not 0.0 for logarithmic scales, rather they appear on the edges of the plot unless set to a nonzero value.

Plot Command Family DIVISIONS

Specifies the approximate number of divisions along a specified linear axis.

Command Syntax:

$$\left\{\begin{array}{l}\textbf{XDIVISIONS}\\ \textbf{YDIVISIONS}\\ \textbf{YTDIVISIONS}\\ \textbf{YBDIVISIONS}\end{array}\right\} = num\_div$$

Option	*Meaning*	*Data Range*
XDIVISIONS	Selects the approximate number of divisions along the X-axis. [1]	Integer>0
YDIVISIONS	Selects the approximate number of divisions along the Y-axis. [2]	Integer>0
YTDIVISIONS YBDIVISIONS	Selects the approximate number of divisions along the Y-Axes of the selected top or bottom half frame axis. [2]	Integer>0
num_div	Specifies the approximate number of divisions. [3]	Integer>0

Remarks:

1. If you specify **XDIVISIONS**, then this approximate number of divisions will appear on every plot until you explicitly reset the default using the **CLEAR XDIVISIONS** command.
2. If you specify **YDIVISIONS**, then its value is automatically used for **YTDIVISIONS** and **YBDIVISIONS**. The **CLEAR YDIVISIONS** command also resets the default values for the half frame plots.
3. If these commands are not used, the number of **XDIVISIONS**, **YDIVISIONS**, **YTDIVISIONSAXIS** and **YBDIVISIONS** is **5** by default. Note that the actual number of divisions is adjusted automatically to result in round values for each division.
4. You may not specify the number of divisions for a logarithmic axis. These are determined automatically based on the plot data. The table below indicates where divisions are placed as a function of the number of log cycles along an axis.

NUMBER OF CYCLES	DIVISIONS ARE AT:
1 or 2	2., 3., 4., 5., 6., 7., 8., 9.
3	2., 3., 5., 7., 9.
4	2., 4., 6., 8.
5	2., 5., 8.
6 or 7	3., 6.
8 or 9 or 10	3.

Plot Command **DRAW LINES**

Enables or disables the drawing of lines between plotted points.

Command Syntax:

DRAWLINES = { ON / OFF }

Option	*Meaning*	*Data Range*
{ ON / OFF }	Toggles line drawing **ON** or **OFF**. [1,2]	Keyword

Remarks:

1. If you do not use this command, lines are drawn. You must explicitly turn them off if you do not want them displayed.
2. You may not disable the drawing of lines unless you have also requested that symbols be placed at your data points by using the **SYMBOL** command.
3. You may also place symbols on the data points used in your plot. Refer to the **SYMBOL** command.

Plot Command Family GRID

Enables or disables the plotting of grid lines parallel to the specified axes.

Command Syntax:

$$\left\{\begin{array}{l}\texttt{XGRID}\\ \texttt{YGRID}\\ \texttt{XTGRID}\\ \texttt{XBGRID}\\ \texttt{YTGRID}\\ \texttt{YBGRID}\end{array}\right\} = \left\{\begin{array}{l}\texttt{ON}\\ \texttt{OFF}\end{array}\right\}$$

Option	*Meaning*	*Data Range*
XGRID	Operates on the **XGRID** plot element. If enabled, lines are drawn perpendicular to the X-axis at each labelled axis division. [1]	Keyword
YGRID	Operates on the **YGRID** plot element. If enabled, lines are drawn perpendicular to the Y-axis at each labelled axis division. [2]	Keyword
XTGRID XBGRID	Operates on the **XTGRID** or **XBGRID** plot elements. If enabled, lines are drawn perpendicular to the X-axis on the top, **XTGRID**, or bottom, **XBGRID**, half frame plots, respectively at each labelled axis division. [1]	Keyword
YTGRID YBGRID	Operates on the **YTGRID** or **YBGRID** plot elements. If enabled, lines are drawn perpendicular to the Y-axis on the top, **YTGRID**, or bottom, **YBGRID**, half frame plots, respectively at each labelled axis division. [2]	Keyword
$\left\{\begin{array}{l}\texttt{ON}\\ \texttt{OFF}\end{array}\right\}$	Enables or disables the specified plot element. [3]	Keyword

Remarks:

1. If you enable or disable the **XGRID** plot element, then the half frame elements **XTGRID** and **XBGRID** are automatically enabled or disabled. Therefore, you must explicitly enable or disable the half frame elements if you want them to be different.
2. If you enable or disable the **YGRID** plot element, then the half frame elements **YTGRID** and **YBGRID** are automatically enabled. Therefore, you must explicitly enable or disable the half frame elements if you want them to be different.
3. The **XGRID**, **YGRID**, **XTGRID**, **XBGRID**, **YTGRID** and **YBGRID** plot elements are **OFF** by default.

Plot Command Family INTERCEPT

Defines an X-Y coordinate where the X-axis and Y-axis will intersect.

Command Syntax:

$$\left\{ \begin{array}{l} \textbf{XINTERCEPT} \\ \textbf{YINTERCEPT} \\ \textbf{YTINTERCEPT} \\ \textbf{YBINTERCEPT} \end{array} \right\} = \textit{coord\_value}$$

Option	*Meaning*	*Data Range*
XINTERCEPT	Defines an X coordinate along which the Y-axis, if requested, will be drawn. [1]	Keyword
YINTERCEPT	Defines a Y coordinate along which the X-axis, if requested, will be drawn. [2]	Keyword
YTINTERCEPT YTINTERCEPT	Define Y coordinates along which the Y-Axes, if requested, will be drawn for top, **YTINTERCEPT**, or bottom, **YBINTERCEPT**, half frame plots. [2]	Keyword
coord_val	Specifies the coordinate value for the selected **INTERCEPT**. [3]	Real

Remarks:

1. If you select an **XINTERCEPT**, then its value is used for all full and half frame plots.
2. If you select a **YINTERCEPT**, then its value is automatically used for **YTINTERCEPT** and **YBINTERCEPT**. You must explicitly clear their values using the **CLEAR YTINTERCEPT** or **CLEAR YBINTERCEPT** commands if you want them reset to the default value. Similarly, if you use the **CLEAR YINTERCEPT** command, it automatically resets the defaults for **YTINTERCEPT** and **YBINTERCEPT**.
3. If any of these commands is not used, its corresponding *coord_value* is **0.0** by default.

Plot Command LINESTYLE

Selects the style of the lines drawn on your plot.

Command Syntax:

```
LINESTYLE = line_code
```

Option	*Meaning*	*Data Range*
line_code	Specifies a code that defines the style of the plotted lines. [1]	Integer>0

Remarks:

1. The meaning of the *line_code* depends on your X-Y Plotter program. For example, it might indicate lines of different thicknesses, dash patterns, or colors. Contact your **UAI/*NASTRAN*** System Support Specialist for detailed information.

Plot Command Family LOG

Enables or disables logarithmic scales for the specified axes.

Command Syntax:

$$\left\{ \begin{matrix} \textbf{XLOG} \\ \textbf{YLOG} \\ \textbf{YTLOG} \\ \textbf{YBLOG} \end{matrix} \right\} = \left\{ \begin{matrix} \text{ON} \\ \text{OFF} \end{matrix} \right\}$$

Option	*Meaning*	*Data Range*
XLOG	Enables or disables the **XLOG** plot element. If enabled, a logarithmic scale is used for the X-axis. [1]	Keyword
YLOG	Enables or disables the **YLOG** plot element. If enabled, a logarithmic scale is used for the Y-axis. [2]	Keyword
YTLOG YBLOG	Enables or disables the **YTLOG** or **YBLOG** plot elements. If enabled, a logarithmic scale is used for the Y-axis of the specified top, **YTLOG**, or bottom, **YBLOG**, frame plots. [2]	Keyword
{ ON } { OFF }	Enables or disables the specified plot element. [3]	Keyword

Remarks:

1. If you enable the **XLOG** plot element, then all full and half frame plots will have logarithmic X-Axes until you explicitly disable them.
2. If you enable or disable the **YLOG** plot element, then the half frame elements **YTLOG** and **YBLOG** are automatically enabled or disabled. Therefore, you must explicitly enable or disable the half frame elements if you want them to be different.
3. If any of these commands is not used, the corresponding **LOG** plot element is **OFF** by default.

Plot Command Family MINMAX

Defines and minimum and maximum X-Y coordinate ranges to be plotted. These commands allow you to control the regions of your plot.

Command Syntax:

```
{ XMIN  }
{ XMAX  }
{ YMIN  }
{ YMAX  } = coord_value
{ YTMIN }
{ YTMAX }
{ YBMIN }
{ YBMAX }
```

Option	*Meaning*	*Data Range*
XMIN XMAX	Define the minimum and maximum X-Axis values that will be plotted. [1]	Keyword
YMIN YMAX	Define the minimum and maximum Y-Axis values that will be plotted. [2]	Keyword
YBMIN YBMAX YTMIN YTMAX	Define the minimum and maximum Y-Axis values that will be plotted for top, **YTMAX** and **YTMIN**, or bottom, **YBMAX** and **YBMIN**, half frame plots. [2]	Keyword
coord_val	Specifies the coordinate value for the selected minimum or maximum. [3]	Real

Remarks:

1. If you specify a value for **XMIN** or **XMAX**, then these values will be used for all full and half frame plots. You may reset these values to the default by using the **CLEAR XMIN** or **CLEAR XMAX** commands.
2. If you specify a value for **YMIN** or **YMAX**, then its value is is automatically used for **YTMIN**, **YTMAX**, **YBMIN**, and **YBMAX**. You must explicitly clear their values using the **CLEAR YTMIN**, **CLEAR YTMAX**, **CLEAR YBMIN** or **CLEAR YBMAX** commands if you want them reset to the default value. Similarly, the **CLEAR YMIN** and **CLEAR YMAX** commands will also reset the defaults for the half frame elements.
3. If you do not use any of these commands, then the minimum and maximum coordinate values for all axes are determined automatically by **UAI/*NASTRAN*** so that all data points will be displayed on your plots.

Plot Command Family SCALES

Requests that tic marks and scale values be placed on one or more of the plot edges and control the frequency with which scale values are drawn.

Command Syntax:

```
{ UPPER SCALES  }
{ LOWER SCALES  }
{ LEFT SCALES   }
{ RIGHT SCALES  } = { ON       }
{ TLEFT SCALES  }   { OFF      }
{ TRIGHT SCALES }   { TICS ONLY }
{ BLEFT SCALES  }
{ BRIGHT SCALES }

{ XVALUE SCALES  }
{ YVALUE SCALES  }
{ YTVALUE SCALES } = skip_value
{ YBVALUE SCALES }
```

Option	*Meaning*	*Data Range*
UPPER SCALES	Operates on the **UPPER SCALES** plot element. [1]	Keyword
LOWER SCALES	Operates on the **LOWER SCALES** plot element. [1]	Keyword
LEFT SCALES	Operates on the **LEFT SCALES** plot element. [2]	Keyword
RIGHT SCALES	Operates on the **RIGHT SCALES** plot element. [3]	Keyword
TLEFT SCALES TRIGHT SCALES BLEFT SCALES BRIGHT SCALES	Operates on tics along the left side, **TLEFT SCALES** or **BLEFT SCALES**, or right side, **TRIGHT SCALES** and **BRIGHT SCALES**, of half frame plots. [2,3]	Keyword
{ ON } { OFF }	Enables or disables the specified plot element. [4]	Keyword
TICS ONLY	Requests that tic marks be placed on the requested coordinate axis but that data scales be omitted.	Keyword
XVALUE SCALES	Operates on the scales of the **XAXIS** plot element.	Keyword
YVALUE SCALES	Operates on the scales of the **YAXIS** plot element.	Keyword
YTVALUE SCALES YBVALUE SCALES	Operates on the scales of the top, **YTVALUE SCALES**, or bottom, **YBVALUE SCALES**, plot elements of half frame plots.	Keyword
skip_value	Selects the number of tic marks to skip between labelled values along the selected axis.	Keyword

Remarks:

1. If enabled, both **UPPER SCALES** and **LOWER SCALES** will appear on all full frame and half frame plots until you explicitly disable them.

2. If you enable or disable **LEFT SCALES**, then the half frame elements **TLEFT SCALES** and **BLEFT SCALES** are automatically enabled or disabled. You must explicitly request the half frame elements if you want different results.

3. If you enable **RIGHT SCALES**, then the half frame elements **TRIGHT SCALES** and **BRIGHT SCALES** are automatically enabled. If you do not want them on the half frame plots, then you explicitly disable them.

4. If you do not use any of these commands, the following defaults are selected:

PLOT ELEMENT	DEFAULT	PLOT ELEMENT	DEFAULT
UPPER SCALES	OFF	BRIGHT SCALES	OFF
LOWER SCALES	ON	BLEFT SCALES	ON
LEFT SCALES	ON	XVALUE SCALES	0
RIGHT SCALES	OFF	YVALUE SCALES	0
TRIGHT SCALES	OFF	YTVALUE SCALES	0
TLEFT SCALES	ON	YBVALUE SCALES	0

All of the values that you request will be printed unless there are so many that they overwrite one another. In such cases, some values will be skipped to make your scales readable.

6. The X-Axis and Y-Axis are always drawn with tic marks. You may not remove them from the plot. This is true even when one or both axes coincide with the left, right, top or bottom scales. For example, if your **XAXIS** lies along the **LOWER SCALE** line, then the command:

```
LOWER SCALES = OFF
```

will not result in the removal of the tic marks.

Plot Command **SYMBOL**

Requests that graphic symbols be placed at the location of plotted points.

Command Syntax:

```
SYMBOL = code [,inc]
```

Option	*Meaning*	*Data Range*
code	Specifies the code of the first symbol to be used. [1]	Integer>0
inc	Defines the frequency with which the symbols will be plotted. [2]	Integer>0

Remarks:

1. The symbol codes are defined in the following table. If you are plotting multiple curves on a single plot, then the first curve will use the symbol you select and successive curves will use the next symbols in the sequence shown.

code	SYMBOL	*code*	SYMBOL
0	None	5	.
1	X	6	O
2	*	7	□
3	+	8	◊
4	-	9	Δ

3. The default *inc* is one, in which case a symbol is placed at every data point. If you have a large number of data points, then you request the plotting of every inc^{th} one.

3. You may also connect your data points with straight line segments. Refer to the **DRAWLINES** command.

Plot Command Family TITLE

Defines a text string which is used as either a frame title or as a title for a specified axis.

Command Syntax:

```
{ FTITLE  }
{ XTITLE  }
{ YTITLE  } = text
{ YTTITLE }
{ YBTITLE }
```

Option	*Meaning*	*Data Range*
FTITLE	Indicates that you are defining a frame title. [1]	Keyword
XTITLE	Indicates that you are defining an X-axis title. [2]	Keyword
YTITLE	Indicates that you are defining a Y-axis title. [3]	Keyword
YTTITLE YBTITLE	Indicates that you are defining a Y-axis title for either top or bottom half frame plots. [4]	Keyword
text	Defines the text to be used for the selected title. [5,6]	Character

Remarks:

1. The frame title, **FTITLE**, will appear on all frames plotted unless you clear it with the **CLEAR FTITLE** command.
2. The X-axis title, **XTITLE**, will appear on all full or half frames plotted unless you clear it with the **CLEAR XTITLE** command.
3. The Y-axis title, **YTITLE**, will appear on all full or half frames plotted unless you clear it with the **CLEAR YTITLE** command.
4. If you do not specify either a **YTTITLE** or a **YBTITLE**, then the **YTITLE**, if defined, will appear on all top and bottom half frame plots. To clear these you use the commands **CLEAR YTTITLE** and **CLEAR YBTITLE**.
5. All axis titling text is centered on the axis. Therefore, any leading or trailing blanks that you enter are ignored. The **FTITLE** is always left justified.
6. If your title *text* is too long to fit along the specified axis, it is truncated on the right.
7. None of the **TITLE** commands may be continued to the next input record.

Plot Command Family **XYPLOT — HISTORY**

Requests the creation of one or more frames containing response history plots.

Command Syntax:

```
{ XYPLOT  }   { ACCE      }
{ XYPRINT }   { DISP      }
{ XYPUNCH }   { FORCE     }
{ XYPEAK  }   { NONLINEAR }
{ XYPAPER }   { OLOAD     }   [ RESPONSE ] [ subcase_list ] frame_list
              { SACCE     }
              { SDISP     }   { AUTO }
              { SPCF      }   { PSDF }      frame_list
              { STRAIN    }
              { STRESS    }
              { SVELO     }
              { VECTOR    }
              { VELO      }
```

Option	*Meaning*	*Data Range*
{ XYPLOT / XYPRINT / XYPUNCH / XYPEAK / XYPAPER }	Selects the command verb that will act on the selected data. [1]	Keyword
{ ACCE / DISP / FORCE / NONLINEAR / OLOAD / PRESS / SACCE / SDISP / SPCF / STRAIN / STRESS / SVELO / VECTOR / VELO }	Selects the response quantity to be plotted as the dependent, Y-axis, variable. [2,3]	Keyword
subcase_list	Selects one or more SUBCASEs for which the response history will be plotted. [4]	[4]
frame_list	Defines each of the frames and curves that will be plotted. [5]	[5]

Remarks:

1. A single command may include more than one command verb, such as:

   ```
   XYPLOT, XYPUNCH DISP RESPONSE 1,2,3 4(T3)
   ```

   If you select the command verb **XYPLOT**, then you must **ASSIGN** a file with **USE=PLOT**. When using the **XYPUNCH** verb, you must **ASSIGN** a file with **USE=PUNCH**.

6 XYPLOT

2. The response quantities available for plotting depend on the analytical discipline that you are using. The table below indicates the meaning and availablity of them.

SYMBOL	DESCRIPTION	RF 1,2	RF 8,11	RF 9,12
ACCE	Acceleration in the physical set		■	■
DISP	Displacement in the physical set.	■	■	■
FORCE	Element force.	■	■	■
NONLINEAR	Nonlinear load.		■	
OLOAD	Applied load.	■	■	■
PRESS	Acoustic pressure.		■	■
SACCE	Acceleration in the solution set.		■	■
SDISP	Displacement in the solution set.		■	■
SPCF	Single-point constraint forces.	■	■	■
STRAIN	Element Strain	■	■	■
STRESS	Element stresses.	■	■	■
SVELO	Velocity in the solution set.		■	■
VELO	Velocity in the physical set.		■	■

3. The independent, or X-axis, variable is automatically determined by the discipline. These are described in the table below.

Rigid Format	APP	DESCRIPTION	Default X-axis
1	DISP NONLIN	STATICS MATERIAL NONLINEAR	SUBCASE
1	HEAT	STEADY-STATE	Not Available
2	DISP	INERTIA RELIEF	SUBCASE
3	DISP HEAT	NORMAL MODES NONLINEAR HEAT TRANSFER	Not Available
4	DISP	DIFFERENTIAL STIFFNESS	Not Available
5	DISP	BUCKLING	Not Available
7	DISP	DIRECT COMPLEX EIGENVALUES	Not Available
8	DISP	DIRECT FREQUENCY RESPONSE	FREQ
9	DISP HEAT	DIRECT TRANSIENT RESPONSE TRANSIENT HEAT TRANSFER	TIME
10	DISP	MODAL COMPLEX EIGENVALUES	Not Available
11	DISP	MODAL FREQUENCY RESPONSE	FREQ
12	DISP	MODAL TRANSIENT RESPONSE	TIME

Rigid Format	APP	DESCRIPTION	Default X-axis
13	DISP	DIFFERENTIAL STIFFNESS MODES	Not Available
14	DISP	CYCLIC STATICS	Not Available
15	DISP	CYCLIC MODES	Not Available
16	DISP	CYCLIC BUCKLING	Not Available
51	DISP	SENSITIVITY STATICS	Not Available
52	DISP	SENSITIVITY MODES	Not Available

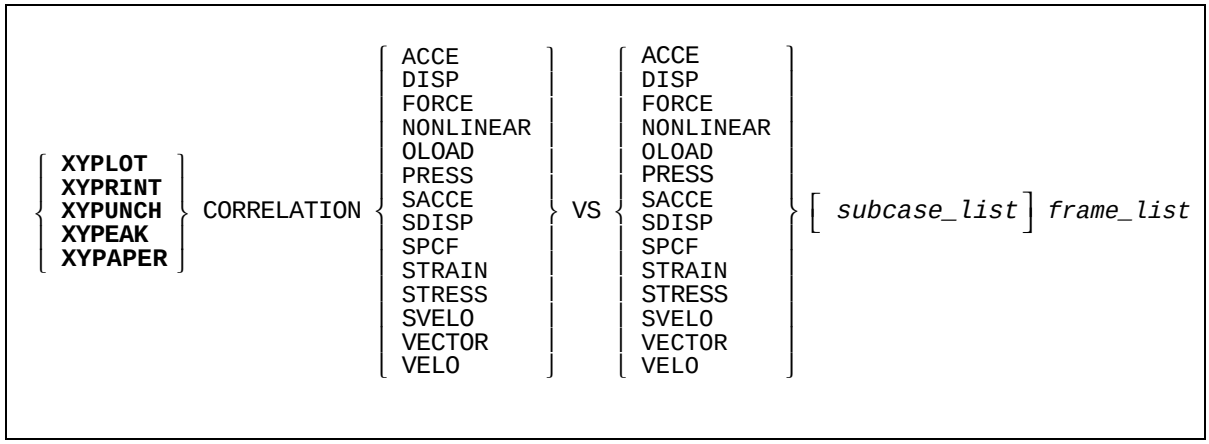
4. The optional *subcase_list* is a list of one or more SUBCASE Identification numbers for which plots will be made. If not specified, all SUBCASEs will be plotted. You will find a description of this list in Section 6.1.7.

5. The *frame_list* is a list of the curves that you wish to plot. You will find a description of this list in Section 6.1.8.

Plot Command Family XYPLOT — CORRELATION

Requests the creation of one or more frames containing correlation plots of one response quantity to another.

Command Syntax:



Option	*Meaning*	*Data Range*
{ XYPLOT, XYPRINT, XYPUNCH, XYPEAK, XYPAPER }	Selects the command verb that will act on the selected data. [1]	Keyword
{ ACCE, DISP, FORCE, NONLINEAR, OLOAD, PRESS, SACCE, SDISP, SPCF, STRESS, SVELO, VECTOR, VELO }	Selects the response quantity to be plotted as the dependent, Y-axis, variable. [2]	Keyword
{ TIME, FREQ, SUBCASE }	Selects the independent, X-axis, variable. [3]	Keyword
subcase_list	Selects one or more SUBCASEs for which the response history will be plotted. [4]	Keyword
frame_list	IDefines each of the frames and curves that will be plotted. [5]	Keyword

Remarks:

1. A single command may include more than one command verb, such as:

   ```
   XYPLOT, XYPUNCH DISP RESPONSE 1,2,3 4(T3)
   ```

   If you select the command verb **XYPLOT**, then you must **ASSIGN** a file with **USE=PLOT**. When using the **XYPUNCH** verb, you must **ASSIGN** a file with **USE=PUNCH**.

2. The response quantities available for plotting depend on the analytical discipline that you are using. See the **XYPLOT — HISTORY** command for a description of these.

3. The independent, or X-axis, variable is automatically determined by the discipline. See the **XYPLOT — HISTORY** command for a description of these.

4. The optional *subcase_list* is a list of one or more SUBCASE Identification numbers for which plots will be made. If not specified, all SUBCASEs will be plotted. You will find a description of this list in Section 6.1.7.

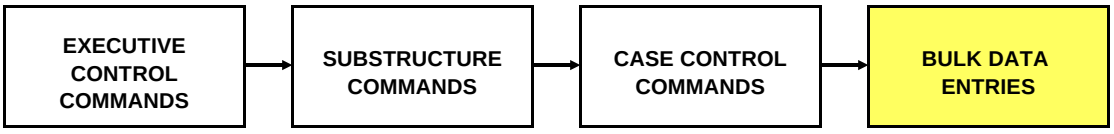
5. The *frame_list* is a list of the curves that you wish to plot. You will find a description of this list in Section 6.1.8.

Chapter 7

BULK DATA

The Bulk Data packet contains all of the data describing the finite element model. These data include geometry, constraints, loads, material properties, and additional control information. This packet is the last one in the input data stream:

Figure 7-1. BULK DATA PACKET LOCATION



The Bulk Data entries must immediately follow the command **BEGIN BULK** and they must end with the command **ENDDATA**.

Most Bulk Data entries are used automatically by **UAI/*NASTRAN***, but others must be referenced by one of the Case Control commands described in Chapter 4. If data are of the latter type, they may remain in the Bulk Data packet even if not referenced with a Case Control command.

7.1 FORMAT OF BULK DATA ENTRIES

The Bulk Data packet consists of a series of ***physical records*** each of which contains 80 or fewer characters. Bulk Data ***entries***, which define the specific information required by **UAI/*NASTRAN***, are logical entities which may encompass one or more physical records. All of these logical entries begin with a key word, or ***mnemonic***, that identifies the Bulk Data entry name.

An entry is subdivided into data ***fields***, each usually containing a maximum of 8 characters of data. This is called ***fixed,*** or ***fixed-field,*** format. With the exception of the first field on an entry, these fields may have the data entered anywhere within the field. For example, it is not necessary to right-justify integer data as would be required using a Fortran format statement. However, the first field has the unique requirement that the data entry name be left-justified in the field. **UAI/*NASTRAN*** also provides a means for using 16 character wide fields, called ***high-precision*** format, for entering numeric data requiring greater accuracy.

Originally, the format for Bulk Data was developed assuming that the data would be in the form of physical punched cards containing 80 columns of data. Because of this, the data were organized in fields of fixed width, with either 8 or 16 characters per field. The detailed documentation of each data entry is presented using this fixed format, although an alternate ***free-field*** format is available and frequently used when generating data interactively with a text editor. When pre-processor computer programs are used to generate the majority of Bulk Data, the data is frequently stored in the fixed format. Any subsequent modifications that you make to the Bulk Data packet may be in the free-field format. In the following sections, the rules for creating Bulk Data entries are presented: for the free-field format, for the 8 character fixed format, for the more accurate 16 character high-precision format, and finally for a data generation feature.

7.1.1 Free-Field Data Entry

The free-field data entry consists of a legal Bulk Data entry beginning with the first character of the record and including a comma (as a field delimiter) within the first 10 characters. **UAI/*NASTRAN*** automatically reformats the data to the proper fixed field entry for subsequent processing. In the simplest form, each free-field entry represents an 8 field, 80 character entry. In the more complex forms, each free-field entry may generate several fixed field or high-precision entries.

The following examples illustrate the possible forms of data input, and the resulting translation to fixed format. Note that the program always echos (to the print file) the input Bulk Data in the translated fixed-field format. Consider the two simple examples:

The free-field entry:

```
GRID,100,,1.0,0.0,0.0,,456
```

is translated to:

GRID	100		1.0	0.0	0.0		456		

and the entries:

```
LOAD,10,1.0,1.0,100,.5,101,-.5,102,+LOD-A
+LOD-A,1.0,103
```

are translated to:

LOAD	10	1.0	1.0	100	.5	101	-.5	102	+LOD-A
+LOD-A	1.0	103							

The second part of the example introduces the concept of the ***parent*** and ***continuation*** of a Bulk Data entry. As shown above, field 10 of the **LOAD** entry (the parent) contains a unique character string which is matched by field 1 of the next entry (the continuation). Parent and continuation entries are more fully described in Sections 7.1.2 and 7.1.3. Continuation entries may be handled in several ways as described in those sections.

For convenience, data normally required on continuation entries when using fixed formats may be included with the parent data on the original free-field entry as shown next. Two forms of input for this case are allowed.

In the first form the continuation mnemonics are included with the data. For example, the free-field entry:

```
SPC1,100,12456,1,2,3,4,5,6,+SPC-A,+SPC-A,7,8,9,10
```

is translated to:

SPC1	100	12456	1	2	3	4	5	6	+SPC-A
+SPC-A	7	8	9	10					

In the second form, the continuation mnemonics are not included because they are not required, and the more convenient entry:

```
SPC1,100,12456,1,2,3,4,5,6,7,8,9,10
```

is translated to:

SPC1	100	12456	1	2	3	4	5	6	
-CONT-	7	8	9	10					

When you review the Bulk Data echo in the print file you will notice that continuation entries are signified by using the

```
-CONT-
```

mnemonic as the first field of the continuation entries.

The second form illustrated above is part of a more general capability relating to continuations. If more than 80 characters of data are required, the free-field entry may be continued by *terminating the parent with a comma.* The next entry will be a logical continuation of the first. It is not required to end the first entry at any specific point. This is illustrated by the entry:

```
CHEXA,200,200,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,
17,18,19,20
```

which is translated to:

CHEXA	200	200	1	2	3	4	5	6	
-CONT-	7	8	9	10	11	12	13	14	
-CONT-	15	16	17	18	19	20			

Because of this feature allowing more than 10 fields of data to be entered on one free-field entry, it ***is not allowed*** to terminate a single free-field entry with a comma. If this is done, the program expects the next input record to be a logical continuation of the free-field input.

With some Bulk Data entries it may be desirable for the free-field continuation entry to begin a new physical record. Using the first method, the data are coded exactly as required by the fixed format, using the comma delimiter to indicate the unused fields. For example, the entries:

```
TSTEP,50,10,0.1,1,,,,,+TS-A
+TS-A,,100,.2,2,,,,,+TS-B
+TS-B,,50,.1,1
```

are translated to:

TSTEP	50	10	0.1	1					+TS-A
+TS-A		100	.2	2					+TS-B
+TS-B		50	.1	1					

Using the second approach, the continuation fields are neither entered nor counted when creating the delimiter (comma) entries. This is shown by the entries:

```
TSTEP,50,10,0.1,1,,,,,
,100,.2,2,,,,,
,50,.1,1
```

which is equivalent to:

```
TSTEP,50,10,0.1,1,,,,,,100,.2,2,,,,,,50,.1,1
```

both of which result in:

TSTEP	50	10	0.1	1					
-CONT-		100	.2	2					
-CONT-		50	.1	1					

This approach is prone to errors since the unused fields between commas must be counted. To simplify data input of this type, the

```
/
```

character may be entered at the beginning a data record to indicate the beginning of a new physical entry. This causes that record to begin a new fixed-field entry as shown next. The previous example could then be entered:

```
TSTEP,50,10,0.1,1,
/,100,.2,2,
/,50,.1,1
```

and will be expanded to the correct fixed-field format as shown above. Notice that this procedure removes the need to "count" the remaining blank fields on parent entries and indicate their spacing with commas.

Another example which illustrates this feature is the case of table data entry where a ***header*** entry, containing little data, is required:

```
TABLED1,1100,
/0.0,.00,10.0,.34,20.0,.23,30.0,.43,
/40.0,.48,ENDT
```

The above free-field entries are expanded to:

TABLED1	1100								
-CONT-	0.0	.00	10.0	.34	20.0	.23	30.0	.43	
-CONT-	40.0	.48	ENDT						

The previous examples used the default translation of free-field to fixed-field data entries.

The same techniques illustrated may also be used to force translation to the high-precision format data entry as shown below. Note that the use of the high-precision entry is initially indicated by the presence of the asterisk (*****) immediately following the data entry mnemonic (with no blank space preceding the *****). The entry:

```
GRID*,100,,1.0,0.0,1.0,,456
```

results in the high-precision translation:

1a	2	3	4	5	10a
GRID	100		1.0	0.0	

1b	6	7	8	9	10b
CONT	1.0		456		

Additional rules for using the high-precision format are presented later in Section 7.1.3.

7.1.2 Fixed-Field Data Entry

The fixed-field data entry form consists of ten fields each of which are eight (8) characters long as indicated in the following diagram:

1	2	3	4	5	6	7	8	9	10
1 8	9 16	17 24	25 32	33 40	41 48	49 56	57 64	65 72	73 80

The Bulk Data mnemonic is entered in field 1 beginning with character 1. Fields 2-9 are for data items. There are limitations on data items. They must lie completely within the designated field. They may have no embedded blanks and they must be of the proper type, i.e., blank, integer, real, double precision, or character. *All real numbers, including zero, must contain a decimal point.* A blank will be interpreted as required. Real numbers may be encoded in various ways. For example, the following are all valid and equivalent real numbers:

```
7. or 7.0 or .7E1 or 0.7+1 or 70.-1 or .70+1
```

A double precision number must contain both a decimal point and an exponent with the character **D** such as:

```
3.5D1 or 2.7D+1 or 70.D-1
```

Double precision data values are only allowed in a few special cases, i.e. when using **DMI** and **DMIG** Bulk Data entries.

Character data values consist of one to eight alphanumeric characters, the first of which must be alphabetic.

Normally field 10 is used to signify that a continuation entry will follow the current parent entry. If the current entry can not have a continuation, field 10 may be used for optional user identification data. The continuation entry contains the symbol

```
+
```

in column 1 followed by the same seven characters that appeared in positions 74-80 of field 10 of the parent entry. This allows the Bulk Data packet to be input with entries out of alphanumeric sort. Also, the continuation entries need not follow the parent when continuation mnemonics are used. Although the **+** sign is not specifically required in column 73 of the parent when the fixed-field format is used, its use is good practice. The **+** character is required when using the free-field input form as presented earlier in Section 7.1.1.

The fixed-field data entry should be more than adequate for data normally associated with structural engineering problems. Since abbreviated forms of floating point numbers are allowed, up to seven significant decimal digits may be used in an eight-character field. Occasionally, however, the input is generated by another computer program or is available in a form where greater precision is desired. In this case, a high-precision format with a sixteen (16) character data field is provided, as described next.

7.1.3 High-Precision Data Entry

Each entry for the high precision format requires the equivalent of two 80 character records as shown below:

1a	2	3	4	5	10a
1 8	9 24	25 40	41 56	57 72	73 80

1b	6	7	8	9	10b
1 8	9 24	25 40	41 56	57 72	73 80

A high-precision entry is denoted by placing the symbol

```
*
```

after the mnemonic in field 1a and some unique character configuration in the last 7 columns of field 10a. The second physical record contains the symbol ***** in column 1 followed by the same seven characters that appeared after column 73 in field 10a of the first entry. The second record may, in turn, be used to point to a high-precision fixed-field continuation entry, depending on whether the continuation entry contains the symbol ***** or the symbol **+** in column 1. Note that the *two physical records* which define a single *logical* high-precision entry must always appear consecu-

tively in your Bulk Data packet. This is different from the fixed-field entries which may appear out of sequence if the continuation mnemonic is used. The use of multiple entries and high-precision entries is illustrated in the following examples:

Fixed-Field Entry with Fixed-Field Continuation.

NAME									+A1
+A1									

High-Precision Entry

NAME*					*B1
*B1					

High-Precision Entry with High-Precision Continuation.

NAME*					*B1
*B1					*B2
*B2					*B3
*B3					

High-Precision Entry with both a Fixed-Field Continuation and a High-Precision Continuation.

NAME*									*B1
*B1									+A2
+A2									*B3
*B3									*B4
*B4									

Fixed-Field Entry with High-Precision Continuation.

NAME									*B1
*B1									*B2
*B2									

Note also that the continuation fields connecting the two records are not required. However, you must still place an asterisk in the first column of the second record to indicate that it contains high-precision data.

7.1.4 Integer List Data Entry

Many Bulk Data entries require a list of integers, such as a list of GRID points or elements. Whenever your list contains more than two entries you may use a general form for an integer list entry. This form allows you to specify the list using a syntax defined as follows:

```
[I1, I2, ...In], [J1 THRU J2 [BY J3]], [K1, K2, ...Kn]
```

The use of **THRU** or **BY** is optional. **THRU** and **THRU/BY** clauses may appear more than once in the list and may be separated by integers not involved in the clause. An example of this feature is illustrated using the **SPC1** Bulk Data entry:

```
SPC1,3,456,1,5,6,7,10,THRU,100,BY,10,200,300,401,THRU,499,655,656
```

7.2 AUTOMATIC DATA GENERATION

UAI/*NASTRAN* provides the capability to automatically generate groups of Bulk Data entries. This is very useful for generating models with a high degree of geometric redundancy. The generation, or more appropriately the replication, of Bulk Data entries is controlled by two special input entries, a ***replication entry*** and a ***counter entry***. These entries provide the rules used to modify desired fields of a normal Bulk Data entry which is called the ***template entry***.

7.2.1 TEMPLATE Entries

The template entry is any standard fixed-field or free-field Bulk Data entry that is to be used as the basis for creating new Bulk Data entries. There are several restrictions on the type of template entries that may be used.

- ❒ The template Bulk Data entry must have only ten fields or less. If the entry is in free-field format, it must correspond directly to the equivalent fixed-field entry. The conveniences that allow for arbitrary continuations of free-field entries is not supported for use with template entries. Continuation fields may be replicated so Bulk Data entries that require continuations can still be handled.
- ❒ The use of high precision template entries *is not allowed.*

7.2.2 REPLICATION Entries

The replication entry defines the actions to be taken to modify each field of the template entry. The available actions are:

- ❒ Duplicate a field
- ❒ Replace a field
- ❒ Increment a field
- ❒ Delete a field
- ❒ Create a new field

Any number of replication entries may be used following a single template entry. Each entry generates a single new Bulk Data entry and will redefine the modification actions (rules) for all fields of the previous replication entry.

The first action rule causes field duplication. It is signified by placing the symbol

```
=
```

in the corresponding field. The special symbol

```
==
```

is used to indicate that this field and all subsequent fields are to be duplicated.

The second rule causes a field to be replaced. The new value is simply entered on the replication entry in the appropriate field. The following example shows the replication of a **GRID** entry using both forms of field duplication and the replacement of the **GRID** id and x-coordinate field.

```
GRID,1000,,1000.0,0.0,0.0
=   ,1001,,1001.0,==
```

The third action rule causes the value of a field to be incremented. This is performed by placing the symbols

```
*(inc) or *inc
```

in the corresponding field. The value of ***inc*** will be added to the value on the template entry so care should be taken so that the data types, integer or real, match. Note that the parenthesis are optional. The following modification of the previous example shows how the **GRID** id and x-coordinate fields can be incremented instead of replaced.

```
GRID,1000,,1000.0,0.0,0.0
=   ,*(1),,*(1.0),==
```

UAI/*NASTRAN* allows the replication feature to be used with continuation fields. When incrementing continuation fields 1 and 10 of a Bulk Data entry, the template field must be of the format

```
+integer
```

and the replicator increment must also be an integer. The next example shows an example of replicating an 8 node **CQUAD8** entry which requires a continuation entry.

```
CQUAD8,1001,1,1001,1002,2002,2001,3001,5002,+1
=     ,*(1),=,*(1),*(1),*(1),*(1),*(1),*(1),*(1)
=(2)
+1    ,4001,5001
*(1),*(1),*(1)
=(2)
```

A field of the template entry can be deleted if the replication entry contains a blank in the appropriate field. In the following example, the permanent single-point constraint field data of the parent **GRID** entry will be deleted on the replicated **GRID** entry.

```
GRID,1000,,1000.0,0.0,0.0,,123456
=   ,*(1),,*(1000.0),=,=
```

The final rule allows the creation of new fields. If the replication entry contains more fields than the template, then the extra fields will be created on the generated Bulk Data entry. If an increment rule applies to blank fields on the template it is assumed that the template value is zero. The following example shows the creation and incrementing of new fields.

```
GRID,1000
=   ,*(1),,*(1.0),,,,5
```

The replication entry itself may by input as either a fixed-field or free-field entry. Since a free-field entry is always converted to fixed-field before processing you are still limited to eight characters per field.

Finally, in order for an input entry to be recognized as a replication entry it must contain either of the symbols:

```
= or *(
```

in the first 10 columns. The sorted Bulk Data echo does not list replication entries. To get a listing of them in the print file, the Case Control command **ECHO** must be used with either the **UNSORT** or **BOTH** options.

7.2.3 COUNTER Entries

The counter entry cause the generation of additional Bulk Data entries using the actions requested with the last replication entry. A replication entry *must* precede the counter entry. The format of this entry is the symbol

```
=(number) or =number
```

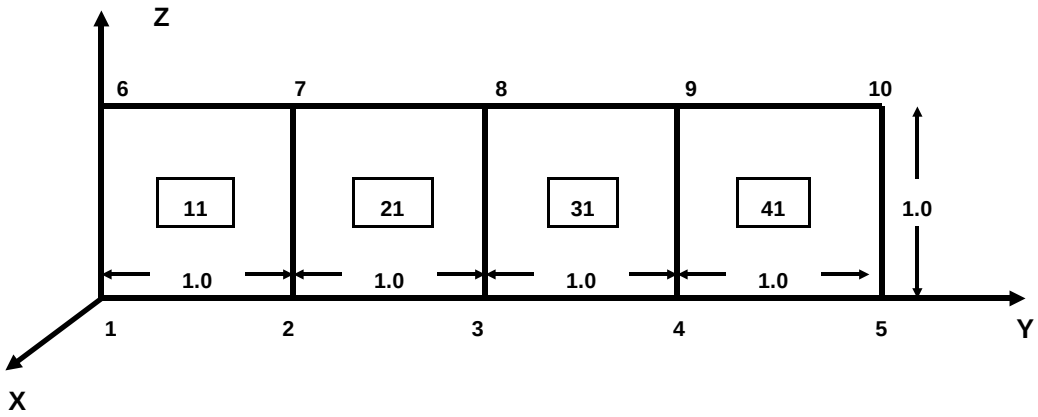
where *number* is the desired number of additional Bulk Data entries to generate. The parentheses are optional. In the following example, a counter entry is used to generate 8 additional **GRID** entries to be added to the template entry and the replication entry. Thus, for this example, 10 **GRID** entries will exist.

```
GRID,1000,,1000.0,0.0,0.0
=   ,*(1),,*(1.0),==
=(8)
```

As with the replication entry, the Case Control command **ECHO** must be used if the entry is to be listed in the print file.

7.2.4 Replication Examples

This section presents several examples to illustrate the use of these replication features. A simple finite element model is defined for which Bulk Data are generated. This model is shown below:



This model represents a flat plate composed of four QUAD4 elements with the dimensions indicated in the figure. The following Bulk Data packet, using replication features, could be used to develop all the geometric data:

```
$ CREATE GRID POINTS 1 THROUGH 5 ALONG THE LINE Z=0.0
GRID,1,,0.0,0.0,0.0
=,*(1),,*(1.0),=
=(3)
$ CREATE GRID POINTS 6 THROUGH 10 ALONG THE LINE Z=1.0
GRID,6,,0.0,0.0,1.0
=,*(1),,=,=,*(1.0)
=(3)
$ CREATE THE QUAD4 ELEMENTS
CQUAD4,11,101,1,2,7,6
=,*(10),=,*(1),*(1),*(1),*(1)
=(2)
```

7.2.5 Restrictions on Replication

There are four restrictions on the replication feature:

- ❒ The template Bulk Data entry must have ten or less fields.
- ❒ High precision entries can not be replicated.
- ❒ None of the fields may contain more than eight characters.
- ❒ The symbol = must appear as one of the first ten characters on the increment and counter entries.

Comment information may be placed on any of the replication entries by using a **$** followed by the commentary. The sorted Bulk Data echo does not list the replication entries. To get a listing of them, the Case Control command **ECHO** must be used with either the **UNSORT** or **BOTH** option.

7.3 BULK DATA DESCRIPTIONS

The detailed descriptions of the Bulk Data entries are contained in this section in alphabetical order. The subsections below summarize the documentation format used for these descriptions.

7.3.1 Format and Examples

The first section of each Bulk Data entry description provides you with the entry format and one or more examples of its use. The figure below illustrates the general format for these descriptions. They are presented in fixed-format for readability. Each field is assigned a name which is referenced in subsequent sections of the description. Fields which are heavily shaded indicate that they are not used for data entry and must be left blank. Those which are lightly shaded contain optional data that you may or may not enter. Fields which are not shaded usually require input. However, many special cases exist wherein these fields may be left blank and a default value, if indicated, will be used.

1	2	3	4	5	6	7	8	9	10
MNEM	FNAME1	FNAME2	FNAME3	FNAME4	FNAME5	FNAME6			-cont-
-cont-		FNAME7	FNAME8						

If an entry may have continuations, this is indicated by the appearence of the symbol

```
-cont-
```

in field 10. Open-ended Bulk Data entries, those which may contain long, open-ended lists of data are shown somewhat differently:

1	2	3	4	5	6	7	8	9	10
MNEM	FNAME1	FNAME2	FNAME3	FNAME4	GR1-1	GR1-2	GR2-1	GR2-2	-cont-
-cont-	GR3-1	GR3-2	GR4-1	GR4-2	*CONTINUES IN GROUPS OF 2*				

In this example, after four fields of fixed data there is an open-ended list of data in groups of two items. The format shown is that used throughout the documentation.

Finally, there are some Bulk Data entries which offer several ways of entering data. These are shown as:

1	2	3	4	5	6	7	8	9	10
MNEM	FNAME1	FNAME2	FNAME3	FNAME4	OP1a	OP2	OP3		
					OP1b				

This format illustrates a case where either (1) fields 6, 7 and 8 should contain one type of related data, or (2) field 6 should contain an alternate form of data and fields 7 and 8 should be left blank.

7.3.2 Field Definitions

Each of the fields of the Bulk Data entry are then described in the section with the following headings:

- ❒ Field
- ❒ Contents
- ❒ Data Range
- ❒ Default

The ***Field*** column gives the name of the field as shown in the format of the data entry. Field names have been selected in a manner which is intended to provide you with memory cues for their use. This is followed by a description of the ***Contents*** of the field. This is then followed by the allowable ***Data Range*** of the values which you may place in the field. The following table defines the data range specifications found in this chapter.

Data Range Specifier	Meaning
Integer>0	Requires that you enter a positive integer number which does not contain a decimal point. The number may be prefixed with a + sign.
Integer	Requires that you enter an integer number which does not contain a decimal point. The number may be prefixed with a + or - sign.
Real	Requires that you enter a real number. **UAI*/NASTRAN*** allows three ways in which to enter such values: **-3.656** The fixed point representation. **3.2E+06** The floating point representation. **-1.65-4** Floating point without the exponent symbol **E**.
DOF Code	The **UAI*/NASTRAN*** DOF (degree-of-freedom) Code. For GRID points, this code is a single digit from 1 to 6, or a concatenated list of these digits with no more than one of each digit specified. For SCALAR points, the DOF Code must be 0 or blank.
Harmonic Code	The **UAI*/NASTRAN*** Harmonic Code is used when performing Axisymmetric Harmonic Analyses. This code allows you to specify a range of harmonics of the form: S*n1*T*n2* *n1* is the first harmonic and *n2* the last harmonic in a sequence. For example, to select harmonics 0 through 10, you would use: S0T10
Character	Requires that you enter a character string of 8 or fewer letters that represents a keyword.
Name	Requires that you enter a character string of 8 or fewer letters that represents the name of a substructure. These names must begin with an alphabetic character.

7 BULK

The fourth descriptor provided is the ***Default*** value for the field. The default value is the value assigned to the field if you leave it blank.

Finally, all fields, which are required and which do not have default values defined, must be entered or you will get an error when you execute the program.

7.3.3 Remarks

For each Bulk Data entry description, there is a section containing remarks. These remarks provide you with additional information describing the way in which the particular entry may be used or to describe additional options that may be available or restrictions that may apply. References to the remarks are cited in the other sections of the description by being placed in square brackets thusly, [1,2].

7.3.4 Usage

If the Bulk Data entry is particularly complex, an additional section is given which provides examples of the usage of the entry. More detailed usage information can be found in the **UAI/*NASTRAN User's Guide***.

Bulk Data Entry $ [Comment]

Defines a comment entry.

Format and Example:

```
   1       2       3       4       5       6       7       8       9       10
$   ANY TEXT CONTAINING LEGITIMATE CHARACTERS
```

```
$ THE FOLLOWING DATA DEFINE THE FRONT BUMPER STRUCTURE
```

Remarks:

1. Comment entries only appear in an unsorted Bulk Data echo.

Bulk Data Entry / [DELETE]

Requests that Bulk Data entries be removed from the RESTART File.

Format and Example:

1	2	3	4	5	6	7	8	9	10
/	K1	K2							

/	10	12							

Field	*Contents*	*Data Range*	*Default*
K1	Sorted sequence number of first entry in sequence to be removed. [1]	Integer>0	Required
K2	Sorted sequence number of last entry in sequence to be removed.	Integer>0	[1,2,3]

Remarks:

1. This entry removes Bulk Data entries having sorted sequence numbers **K1** through **K2** from the Bulk Data stored on the RESTART File.
2. If **K2** is blank, only entry **K1** is removed from the Bulk Data RESTART File.
3. The Bulk Data entry sorted sequence numbers are obtained by using the Case Control command:

   ECHO = SORT

   in the CHECKPOINT execution.
4. These entries are used only when performing a RESTART.

Bulk Data Entry ACCEL

Defines static acceleration loads, which may vary over a region of the model, of the form:

$$\vec{f} = \mathbf{VAL} \cdot \vec{\mathbf{V}}$$

The load variation, **VAL**, is based upon tabular input defined on this Bulk Data entry.

Format and Example:

1	2	3	4	5	6	7	8	9	10
ACCEL	LID	CID	V1	V2	V3	DIR			-cont-
-cont-	LOC1	VAL1	LOC2	VAL2	*CONTINUES IN GROUPS OF 2*				-cont-
ACCEL	100	0	0.0	0.0	1.0	X1			
	0.0	1.0	1000.0	3.0					

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
CID	Coordinate system identification number used for definition of the acceleration vector. [3]	Integer≥0	**0**
Vi	Components of the acceleration vector measured in coordinate system **CID**.	Real	[4]
DIR	Component direction of acceleration variation. [5]	Character { **X1** **X2** **X3** }	Required
LOCi	Location along direction **DIR** in coordinate system **CID** for specification of a load factor. [6]	Real	Required
VALi	The load factor associated with location **LOCi**. [6]	Real	Required

Remarks:

1. Acceleration loads may be combined with other loads, such as **FORCE**, **MOMENT** and **GRAV** loads by using **LOAD** Bulk Data entries. The **LID** on the **ACCEL** entry may not be the same as that of any other load data.
2. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

3. The acceleration vector is defined in coordinate system **CID**. If **CID** is not specified, the acceleration vector is defined in the Basic Coordinate System.

7 BULK

4. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.

5. The **DIR** field must contain one of the character strings **X1**, **X2** or **X3**. The **DIR** direction defines the direction of acceleration load variation along direction 1, 2 or 3 respectively of coordinate system **CID**.

6. A minimum of two pairs of (**LOCi**, **VALi**) data must be defined. The scale factor **VAL** is found by interpolating the **DIR** coordinate in the table. If the GRID point coordinate is outside the range of the table, **VAL** is determined either from **VAL1** or **VALn** (the last value).

Bulk Data Entry ACCEL1

Defines static acceleration loads at individual GRID points of the model of the form:

$$\vec{f} = \mathbf{G} \cdot \vec{\mathbf{V}},$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
ACCEL1	LID	CID	G	V1	V2	V3			-cont-
-cont-	*GRID ID LIST*								-cont-

ACCEL1	101	0	386.0	0.0	1.0	1.0			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
CID	Coordinate system identification number used for definition of the acceleration vector. [3]	Integer≥0	**0**
G	Constant multiplier for the acceleration vector. [4]	Real	Required
Vi	Components of the acceleration vector measured in coordinate system **CID**.	Real	[4]
GRID ID LIST	List of one or more GRID or SCALAR point identification numbers.	Integer>0	Required

Remarks:

1. Acceleration loads may be combined with other loads, such as **FORCE**, **MOMENT** and **GRAV** loads by using **LOAD** Bulk Data entries. The **LID** on the **ACCEL** entry may not be the same as that of any other load data.
2. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

3. The acceleration vector is defined in coordinate system **CID**. If **CID** is not specified, the acceleration vector is defined in the Basic Coordinate System.
4. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.

7 BULK

Bulk Data Entry ACORFAC

Defines a matrix of correction factors for the DLM or CPM aerodynamic matrix data.

Format and Example:

1	2	3	4	5	6	7	8	9	10
ACORFAC	SYMMETRY	MACH	KFREQ	SF	TID				
ACORFAC	ANTI	0.85	0.2	0.9	101				

Field	*Comments*	*Data Range*	*Default*
SYMMETRY	Symmetry condition for which the referenced correction factors apply. [1]	Character	Required
MACH	Mach number at which the referenced correction factors are applied.	Real≥0.0	Required
KFREQ	Reduced Frequency at which the referenced correction factors are applied.	Real≥0.0	Required
SF	Effectiveness Scale Factor to be used with the correction factors. [2]	0.0≤Real≤1.0	Required
TID	Table identification number of a **TABLEA1** Bulk data entry that defines the input Correction factors.	Integer >0	Required

Remarks:

1. The correction factor matrix $\boldsymbol{W}_{kk}$ is a premultiplier matrix on the unsteady aerodynamic forces, where:

$$\boldsymbol{Q}_{hh} = \boldsymbol{\Phi}_{dh}^{T}\boldsymbol{G}_{kd}^{T}\ \boldsymbol{W}_{kk}\,\boldsymbol{S}_{kj}\ \boldsymbol{Q}_{jj}\ \boldsymbol{D}_{jk}\ \boldsymbol{G}_{kd}\,\boldsymbol{\Phi}_{dh}$$

2. The scale factor can be used to modify the input correction factor values to reduced effectiveness at selected values of reduced frequency. The correction factors given by the **TABLEA1** are modified by the following equation:

$$W_{kk}^{scaled} = 1.0 - \mathbf{SF}\,(\,1.0 - W_{kk}^{input}\,)$$

where W_{kk}^{scaled} is the effective weighting factors used, and W_{kk}^{input} is input from the **TABLEA1** entry.

7 BULK

Bulk Data Entry ACSRCE

Defines an acoustic source of the form:

$$\dot{Q}(f) = A\ \dot{q}(f)\, e^{i(\theta - 2\pi f\tau)}$$

for use in Fluid-Structure Interaction analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
ACSRCE	LID	ADEF	τDEF	θDEF	CTAB	RHO	B		

ACSRCE	106	104			102	1.0	15.0		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
ADEF	Identification number of a **DAREA** Bulk Data entry which defines *A*.	Integer>0	Required
τDEF	Identification number of a **DELAY** set which defines τ.	Integer≥0	[3]
θDEF	Identification number of a **DPHASE** set which defines θ.	Integer≥0	[3]
CTAB	Identification number of a **TABLEDi** entry which define power versus frequency.	Integer>0	Required
RHO	Density of the fluid.	Real>0.0	Required
B	Bulk modulus of the fluid.	Real>0.0	Required

Remarks:

1. The acoustic source must be selected in the Case Control packet with the command:

   DLOAD = LID

2. **LID** must be unique for all **ACSRCE**, **DLOAD**, **RLOAD1**, **RLOAD2**, **TLOAD1**, and **TLOAD2** Bulk Data entries.

3. If either of **τDEF** or **θDEF** is blank or zero, then the corresponding τ or θ will be zero.

4. The source strength $\dot{q}(f)$ is computed from **CTAB**, **RHO**, and **B**.

7 BULK

Bulk Data Entry AEFACT

Specifies a list of division points that define chordwise or spanwise Doublet Lattice and supersonic Constant Pressure Panel aerodynamic element distributions.

Format and Example:

1	2	3	4	5	6	7	8	9	10
AEFACT	SID	D1	D2	D3	D4	D5	D6	D7	-cont-
-cont-	D8	*CONTINUES WITH LIST OF DIVISION POINTS.*							-cont-

AEFACT	SID	0.0	0.2	0.5	0.8	1.0			

Field	*Comments*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
Di	Division points as decimal fraction of total span or total chord. [2,3]	Real≥0.0	Two Points Required

Remarks:

1. The **AEFACT** entry is selected by the **NCHORD** or **NSPAN** fields on the **CAERO1** Bulk Data entry when unequal panel divisions are desired.
2. The number of divisions generated is one less than the number of division points specified.
3. The value of the first and last divisions are not required to be 0.0 and 1.0. If the the first and last divisions are not 0.0 and 1.0, then only that portion of the panel geometry defined by the **CAERO1** entry will be meshed with aerodynamic box elements.

7 BULK

Bulk Data Entry AEREFS

Defines aerodynamic reference parameters.

Format and Example:

1	2	3	4	5	6	7	8	9	10
AEREFS	ACID	RCID	CBAR						
AEREFS	1	2	162.5						

Field	*Comments*	*Data Range*	*Default*
ACID	Aerodynamic coordinate system identification number. [1]	Integer	**0**
RCID	Static Aero Reference coordinate system identification number.[2]	Integer	**0**
CBAR	Wing Reference Chord. [3,4]	Real≥0.0	**1.0**

Remarks:

1. The **ACID** aero coordinate system is used to define the direction of positive flow. The x-axis of this system defines the direction of positive fluid flow, and the z- and y-axes define the direction of positive vertical and lateral gust velocity.
2. The **RCID** coordinate system is used to locate the reference axes for the nondimensionalized stability derivatives.
3. The Reference Chord, **CBAR** is used for determining the reduced frequency:

   $$k = \frac{\omega\ \textbf{CBAR}}{2V}$$

   and for non-dimensionalizing the pitching moment stability derivatives in static aeroelastic problems.
4. Note that only one **AEREFS** entry is allowed in the Bulk Data input packet.

7 BULK

Bulk Data Entry AEUNITS

Defines conversion factors for aeroelastic analysis parameters.

Format and Example:

1	2	3	4	5	6	7	8	9	10
AEUNITS	VFAC	VUNITS							
AEUNITS	20.24	KNOTS							

Field	*Comments*	*Data Range*	*Default*
VFAC	Conversion factor on air velocity, for Flutter and Dynamic Aeroelastic Response. [1,2]	Real>0.0	**1.0**
VUNITS	Identifier label on units of air velocity, for Flutter and Dynamic Aeroelastic Response.	Character	Blank

Remarks:

1. This Bulk Data entry is used to define a convenient set of units for specific variables. The conversion factors are used to convert user input quantities to consistent units for internal calculations, and to convert results back to the user selected units for output.

   The conversion factor is used as a multiplier on the input quantity to convert to consistent modeling units, and as a divisor on results in model units to yield output quantities in the input units. In the general case then:

   $$X_{model\ units} = X_{factor}\ X_{input\ units}$$

   Note also that any effected output quantity is printed with the related user defined a **UNITS** label. See Remark 3 for practical examples.

2. **VFAC** and **VUNITS** provide units conversion on vehicle airspeeds. The input airspeeds effected are input on the **ATMOS** Bulk Data entries, and they are referenced by the **VLIST** Case Control command.

3. A typical application of these factors is the case of a Flutter solution using airspeed units of Knots equivalent airspeed and a structural model that uses *lb-inch-sec* units. The desired value for **VFAC** would be 20.24 to convert 1.0 Knot to 20.24 in/sec. Whether the input velocities are interpreted as True or Equivalent depends on the **DENS** Case Control command and the **ATMOS** Bulk Data entry.

7 BULK

Bulk Data Entry: ASET

Defines component degrees-of-freedom that will be placed in the analysis set.

Format and Example:

1	2	3	4	5	6	7	8	9	10
ASET	GID1	DOF1	GID2	DOF2	GID3	DOF3	GID4	DOF4	
ASET	16	1	23	123	38	23			

Field	*Contents*	*Data Range*	*Default*
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom. [1]	DOF Code	Required

Remarks:

1. A component referenced on this entry may not appear as a dependent component in a multi-point constraint relation (**MPC**) or Rigid element, nor may it be referenced on an **SPC**, **SPC1**, **SPCD**, **OMIT**, **OMIT1**, or **SUPORT** entry or on a **GRID** entry as a permanent single-point constraint.
2. When **ASET** and/or **ASET1** entries are present, all degrees of freedom not otherwise constrained will be placed in the omitted set (o-set).

7 BULK

Bulk Data Entry ASET1

Defines component degrees-of-freedom that will be placed in the analysis set.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
ASET1	DOF	*GRID ID LIST*							-cont-
ASET1	345	2	4	8	106	134	18	1001	+A
+A	1003	1007							

Field	*Contents*	*Data Range*	*Default*
DOF	List of degrees of freedom. [1]	DOF Code	Required
GRID ID LIST	List of one or more GRID or SCALAR point identification numbers.	Integer>0	Required

Remarks:

1. A component referenced on this entry may not appear as a dependent component in a Multipoint Constraint relation (**MPC**) or Rigid element, nor may it be referenced on an **SPC**, **SPCD**, **SPC1**, **OMIT**, **OMIT1**, or **SUPORT** entry or on a **GRID** entry as a permanent Single-Point constraint.
2. When **ASET** or **ASET1** entries are present, all degrees-of-freedom not otherwise constrained will be placed in the omitted set (o-set).

7 BULK

Bulk Data Entry ASETAX

Defines the set of generalized harmonic motions that define the allowable motions of axisymmetric rings.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
ASETAX	RID1	HID1 HCODE1	DOF1	RID2	HID2 HCODE2	DOF2			

Using an Explicit Harmonic List:

ASETAX	37	0	246	38	1	135			

Using a Harmonic Sequence Specifier:

ASETAX	39	S1T2	123	40	S0T4	456			

Field	*Contents*	*Data Range*	*Default*
RIDi	Axisymmetric ring, RINGAX, identification number.	Integer>0	Required
HIDi	Harmonic identification number.	Integer≥0	Required
HCODEi	Harmonic sequence specifier.	Harmonic Code	Required
DOFi	List of degrees of freedom. [1]	DOF Code	Required

Remarks:

1. When **ASETAX** or **ASETAX1** entries are present, all degrees-of-freedom not otherwise constrained will be placed in the omitted set, o-set.

7 BULK

Bulk Data Entry ASETAX1

Defines the set of generalized harmonic motions that define the allowable motions of axisymmetric rings.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
ASETAX1	HID HCODE	DOF	*RINGAX ID LIST*						-cont-

Using a Harmonic Identifier:

ASETAX1	0	135	1	2	3				

Using a Harmonic Sequence Specifier:

ASETAX1	S0T4	246	27	49	19	23	67		

Field	*Contents*	*Data Range*	*Default*
HID	Harmonic identification number.	Integer≥0	Required
HCODE	Harmonic sequence specifier.	Harmonic Code	Required
DOF	List of degrees of freedom. [1]	DOF Code	Required
RINGAX ID LIST	List of one or more axisymmetric ring, RINGAX, identification numbers.	Integer>0	Required

Remarks:

1. When **ASETAX** or **ASETAX1** entries are present, all degrees of freedom not otherwise constrained will be placed in the omitted set, o-set.

7 BULK

Bulk Data Entry ATMOS

Defines a density ratio list and a relation between Mach number, density ratio and airspeed to represent a standard atmosphere model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
ATMOS	AID	RHO0							-cont-
-cont-	LABEL1	DENS1	VELM11						-cont-
-cont-	LABEL2	DENS2	VELM12						-cont-
-cont-	*CONTINUES IN GROUPS OF THREE*								-cont-

ATMOS	11	1.147-7							+STD1
+STD1	30K	0.37472	360.58						+STD2
+STD2	10K	0.73859	548.24						+STD3
+STD3	SL	1.0	661.0						

Field	*Comments*	*Data Range*	*Default*
AID	Atmosphere model identification number. [1,2]	Integer	Required
RHO0	Reference fluid density and specification of input velocities as True or Equivalent Airspeeds. [3,4]	Real>0.0	**1.0**
LABELi	Identifier of Density / Altitude line.	Character	Required
DENSi	Density ratio for current Density / altitude line. [5]	Real<0.0	Required
VELM1i	Velocity at Mach=1.0 for current Density Ratio. [4]	Real>0.0	Required

Remarks

1. An **ATMOS** Entry must be selected in the Case Control packet with the command:

   DENS = AID

   This will cause the Flutter analysis to loop over all values of density ratio defined by the single **ATMOS** entry. If no velocity list is specified in the **FLUTTER** Case , then a single velocity corresponding to the *match point* will be run for each Mach-Density combination. See the Case Control subcommand **VLIST** for more information.

2. The **ATMOS** Bulk Data entry is used to identify a standard atmosphere model, for example a *Hot Day* or a *Cold Day*. The specific number of density-altitude points is used to control automated looping analyses in the Flutter discipline. Remarks 4 and 5 provide further discussion of Flutter analysis control.

3. The fluid density used in the flutter analysis is determined by the product of the density ratio **DENSi** and the Reference Fluid Density **RHO0**. The density for the i^{th} density ratio is:

   ρ_i = **DENSi * RHO0**

   The Reference Fluid Density **RHO0** (usually the sea level air density) is also used to determine the equivalent airspeed in a flutter analysis at the i^{th} density ratio where,

   $V_{equiv} = \sqrt{\sigma_i}\, V_{true}$

7 BULK

4. The interpretation of input velocities depends on the value of **RHO0**. If **RHO0** is **1.0**, then the **VELM1i** fields are interpreted as velocities in true airspeed. On the other hand, if **RHO0** is not **1.0**, then the **VELM1i** fields are assumed to be velocities in equivalent airspeed. In addition, the units of the input velocities can be modified, see the description of the **VFAC** and **VUNITS** fields on the **AEUNITS** Bulk Data Entry.

5. This entry applies to vehicles in atmospheric flight and assumes that speed and Mach number are proportional for a fixed Density Ratio. The atmosphere is defined by a series of points relating Airspeed at Mach=1.0 to a density ratio. Each point establishes a line of constant Density / altitude on a plot of Equivalent Airspeed vs. Mach Number. This is sufficient to determine the correct velocity at any Mach number along the given density / altitude line. This constraint between Mach, density and velocity is used for determining the match point flutter analysis conditions.

Bulk Data Entry AXIC

Defines the highest harmonic number for an axisymmetric harmonic analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
AXIC	H								

Field	*Contents*	*Data Range*	*Default*
H	Highest harmonic number defined for the problem.	0≤Integer≤998	Required

Remarks:

1. This entry is required when axisymmetric harmonic elements (CONEAX, TRAPAX and TRIAAX) are used. Motions for harmonic displacements **0** through **H** will be generated for all RINGAX and RINGFL points in the model.
2. Only one **AXIC** entry is allowed in the Bulk Data packet.
3. The **AXIC** entry is not allowed in heat transfer analysis. The axisymmetric harmonic elements may not be used for heat transfer. Instead, the TORDRG, TRAPRG and TRIARG elements may be used.
4. Axisymmetric harmonic elements may be used with three-dimensional modeling. However, only one cylindrical coordinate system with RINGAX points connected by axisymmetric elements may be used in such a model.

Bulk Data Entry BAROR

Defines the default values of property and orientation fields for each **CBAR** entry whose corresponding fields are blank.

Format and Example:

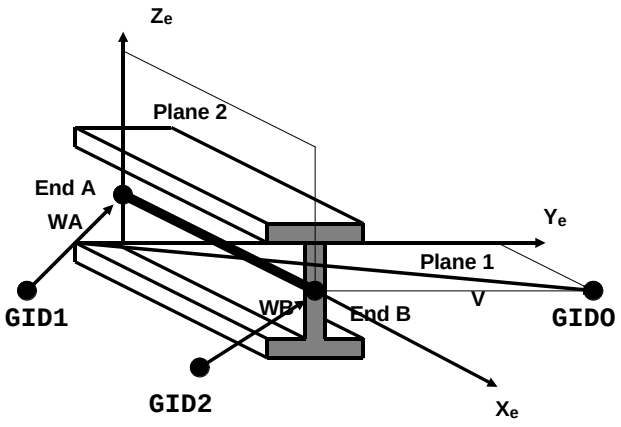
1	2	3	4	5	6	7	8	9	10
BAROR		PID			V1	V2	V3		
					G0				

BAROR		20			1.0	0.0	0.0		

Field	*Contents*	*Data Range*	*Default*
PID	Identification number of a **PBAR** or **PBAR1** property entry.	Integer>0	[1]
Vi	Components of a vector, **V**, originating at End A. This vector, along with the vector from End A to End B, is used to determine the orientation of the element coordinate system. These vectors define the X-Y plane (also called Plane 1) of the element coordinate system. **Vi** must be specified in the output coordinate system for **GID1**.	Real	[1,2]
G0	GRID point identification number used to define the element orientation.	Integer>0	[1,2]

Remarks:

1. At least one of the data fields, **PID** or **GO** or **Vi**, must be non-blank.
2. If Field 6 is an integer then the GRID point **GO** is used to define the BAR element coordinate system orientation as shown in the adjoining figure. If it is blank or real, then the vector **V** is used.
3. Only one **BAROR** entry may appear in the Bulk Data packet.
4. The contents of fields on this entry are used for any **CBAR** entry whose corresponding fields are blank.



7 BULK

Bulk Data Entry BDYC

Defines a boundary set for a REDUCE, MREDUCE, or CREDUCE operation. This boundary set will define the degrees of freedom existing in the reduced substructure. The reduced substructure will also contain any modal degrees of freedom introduced with an MREDUCE or CREDUCE operation.

Format and Example:

1	2	3	4	5	6	7	8	9	10
BDYC	SID	SNAME1	SID1	SNAME2	SID2	SNAME3	SID3		-cont-
-cont-		SNAME4	SID4	*CONTINUES IN GROUPS OF 2*					-cont-

BDYC	157	WINGR	7	MIDWG	15	FUSLG	32		
		POD1	175						

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of the boundary set. [1,2]	Integer>0	Required
SNAMEi	Name of a Basic Substructure which contains the GRID points referenced by boundary set **SIDi**. [3]	Name	Required
SIDi	Identification number of the boundary set (**BDYS** and **BDYS1** entries) associated with Basic Substructure **SNAMEi**. [4]	Integer>0	Required

Remarks:

1. The **SID** number must be unique with respect to all other **BDYC** entries.
2. Boundary sets must be selected in the Substructure Control packet using the subcommand:

   BOUNDARY = SID

   This is a subcommand of the substructure commands REDUCE, MREDUCE and CREDUCE.
3. The same substructure name may not appear more than once per set.
4. The same **SIDi** number may appear for different Basic Substructures. These numbers reference the **SID**s of **BDYS** and **BDYS1** Bulk Data entries. **SID** may also be the same as the **SIDi** numbers.
5. After two or more Basic Substructures are combined, the connected degrees of freedom are actually the same and may be referenced with any one of the Basic Substructure names. However, redundant specification is allowed.

7 BULK

Bulk Data Entry BDYS

Defines a boundary set of GRID points and degrees of freedom. Used by substructuring for a Basic Substructure in the substructure REDUCE, MREDUCE and CREDUCE operations. For other jobs, used to define a set of degrees of freedom that are always retained in the *a-set*.

Format and Example:

1	2	3	4	5	6	7	8	9	10
BDYS	SID	GID1	DOF1	GID2	DOF2	GID3	DOF3		
BDYS	7	13	123	15	123456	17	123		

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of a **BDYS** set. [1]	Integer>0	Required
GIDi	GRID or SCALAR point identification number in a Basic Substructure.	Integer>0	Required
DOFi	List of degees of freedom	DOF Code	Required

Remarks:

1. The same **SID** may appear on more than one **BDYS** entry. The **SID** must be referenced by a **BDYC** entry when performing substructuring, or by a **BOUNDARY** Case Control command in other jobs.
2. The set of boundary points defines the degrees of freedom which are to be retained in the model after the substructure REDUCE, MREDUCE or CREDUCE operations have been performed. An alternate format is provided by the **BDYS1** entry.
3. The **BDYS** data may also be used in any job to insure that specified degrees of freedom will be retained in the *a-set*. By doing so, these degrees of freedom will be uneffected by Dynamic Reduction or the **AUTOOMIT**, **AUTOSPC**, **AUTOREDUCE** and **NLREDUCE** operations.

7 BULK

Bulk Data Entry BDYS1

Defines a boundary set of GRID points and degrees of freedom. Used by substructuring for a Basic Substructure in the substructure REDUCE, MREDUCE and CREDUCE operations. For other jobs, used to define a set of degrees of freedom that are always retained in the *a-set*.

Format and Example:

1	2	3	4	5	6	7	8	9	10
BDYS1	SID	DOF	*GRID ID LIST*						-cont-
BDYS1	15	123456	275	280	THRU	307	320	1001	

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of a **BDYS1** set. [1]	Integer>0	Required
DOF	List of degrees of freedom	DOF Code	Required
GRID ID LIST	List of one or more GRID or SCALAR point identification number of a Basic Substructure.	Integer>0	Required

Remarks:

1. The same **SID** may appear on more than one **BDYS** entry. The **SID** must be referenced by a **BDYC** entry when performing substructuring, or by a **BOUNDARY** Case Control command in other jobs.
2. The set of boundary points defines the degrees of freedom which are to be retained in the model after the substructure REDUCE, MREDUCE or CREDUCE operations have been performed. An alternate format is provided by the **BDYS** entry.
3. The **BDYS1** data may also be used in any job to insure that specified degrees of freedom will be retained in the *a-set*. By doing so, these degrees of freedom will be uneffected by Dynamic Reduction or the **AUTOOMIT**, **AUTOSPC**, **AUTOREDUCE** and **NLREDUCE** operations.

7 BULK

Bulk Data Entry BEAMOR

Defines default values for the property and orientation of each **CBEAM** entry whose corresponding fields are blank.

Format and Example:

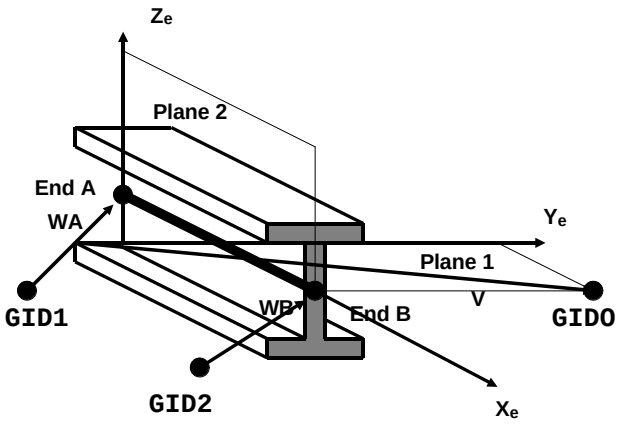
1	2	3	4	5	6	7	8	9	10
BEAMOR		PID			V1	V2	V3		
					G0				

BEAMOR		20			1.0	0.0	0.0		

Field	*Contents*	*Data Range*	*Default*
PID	Identification number of a **PBEAM** property entry.	Integer>0	[1]
Vi	Components of a vector, **V**, originating at End A. This vector, along with the vector from End A to End B, is used to determine the orientation of the element coordinate system. These vectors define the X-Y plane (also called Plane 1) of the element coordinate system. **Vi** must be specified in the output coordinate system for **GID1**.	Real	[1,2]
G0	GRID point identification number used to define the element orientation.	Integer>0	[1,2]

Remarks:

1. At least one of the data fields, **PID** or **G0** or **Vi**, must be non-blank.
2. If Field 6 is an integer, then the GRID point **GO** is used to define the BEAM element coordinate system orientation as shown in the adjoining figure. If it is blank or real, then the vector **V** is used.
3. Only one **BEAMOR** entry may appear in the Bulk Data packet.
4. The contents of fields on this entry are used for any **CBEAM** entry whose corresponding fields are blank.



7 BULK

Bulk Data Entry BMFORCE

Defines an equivalent beam element composed of solid elements. The three-dimensional stress field of the solid elements is converted into equivalent beam forces (moments, shears, axial loads, and torques) that are output at selected stations along the axis of the beam.

Format and Example:

1	2	3	4	5	6	7	8	9	10
BMFORCE	EID	CID							-cont-
-cont-	X1	X2	X3	*CONTINUES WITH LIST OF VALUES*					-cont-
-cont-	EID1	EID2	EID3	*CONTINUES WITH LIST OF VALUES*					-cont-

BMFORCE	101	10							+A
+A	0.0	0.25	0.5	0.75	1.0				+B
+B	1	THRU	100						

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1,2]	Integer>0	Required
CID	Coordinate system identification number. [3]	Integer≥0	**0**
Xi	List of stations along the equivalent beam where forces will be computed. [4]	Real≥0.0	Required
EIDi	List of element identification numbers defining the pseudo-element. [5]	Integer>0	Required

Remarks:

1. **BMFORCE** elements must be selected in the Case Control packet with the command:

   BMFORCE=*sid*

   where *sid* references an integer **SET** which gives the identification numbers of **BMFORC1** and **BMFORCE** elements to be computed.

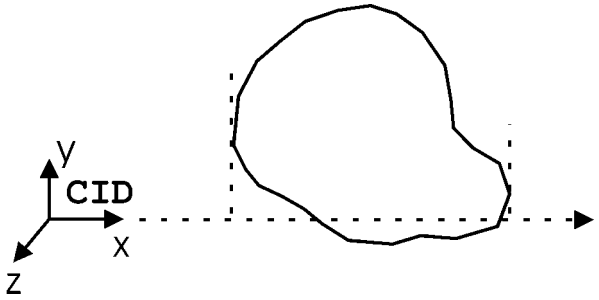
2. The element identification numbers, **EID**, must be unique.

3. The element coordinate system of the equivalent beam is defined by **CID**.

7 BULK

New: V20.1

4. The data recovery stations are defined by a set of real values defined **Xi**. The three-dimensional stress field of the element set given by **EIDi** is converted into equivalent beam forces at normal sections defined by the station values. For stations outside the projection of the **EIDi** elements on the **CID** x-axis, no beam forces can be, or will be, computed. This is illustrated in the following figure:



5. The element identification numbers are defined by a list of integer values. At least one **EIDi** must be input, and the continuation is repeated until all values have been entered. Note that the first **EIDi** must begin in Field 2 of the continuation entry following the one which contains the last **Xi** value.

New: V20.1

Bulk Data Entry BMFORC1

Defines an equivalent beam element composed of solid elements. The three-dimensional stress field of the solid elements is converted into equivalent beam forces (moments, shears, axial loads, and torques) that are output at selected stations along the axis of the beam.

Format and Example:

1	2	3	4	5	6	7	8	9	10
BMFORC1	EID	CID	SSID	ESID					
BMFORC1	101	10	201	301					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1,2]	Integer>0	Required
CID	Coordinate system identification number. [3]	Integer≥0	**0**
SSID	Station set identification number. [4]	Integer>0	Required
ESID	Element set identification number. [5]	Integer>0	Required

Remarks:

1. **BMFORC1** elements must be selected in the Case Control packet with the command:

   BMFORCE=*sid*

   where *sid* references an integer **SET** which gives the identification numbers of **BMFORCE** and **BMFORC1** elements to be computed.

2. The element identification numbers, **EID**, must be unique.

3. The element coordinate system of the equivalent beam is defined by **CID**.

4. The data recovery stations are defined by a set of real values defined by **SETR** Bulk Data entries. The three-dimensional stress field of the element set given by **ESID** is converted into equivalent beam forces at normal sections defined by the station values. For stations outside the projection of the **ESID** elements on the **CID** x-axis, no beam forces can be, or will be, computed. See **BMFORCE** for an illustration.

5. The element identification numbers are defined by a set of integer values defined by either **SETI** or **SETOP** Bulk Data entries.

7 BULK

New: V20.1

Bulk Data Entry CAERO1

Defines an aerodynamic panel and mesh definition for subsonic Doublet Lattice and supersonic Constant Pressure Panel methods of unsteady aerodynamics.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CAERO1	PID		CID	**EQUAL**	NSPAN	**EQUAL**	NCHORD	IGID	-cont-
				or		*or*			
				SET	SSID	**SET**	CSID		
-cont-	X1	Y1	Z1	CROOT	X2	Y2	Z2	CTIP	

CAERO1	101			EQUAL	8	EQUAL	6		+C1
+C1	0.0	0.0	0.0	10.0	0.0	20.0	0.0	5.0	

CAERO1	2001			SET	101	EQUAL	8		+C1
+C1	0.0	0.0	0.0	10.0	0.0	20.0	0.0	5.0	

Field	*Comments*	*Data Range*	*Default*
PID	Aerodynamic panel identification label.	Character	Required
CID	Identification number of coordinate system for locating **Xi**, **Yi** and **Zi**.	Integer>0	**1**
EQUAL NSPAN	Number of equally spaced spanwise divisions. [2]	Integer>0	Required
SET,SSID	Identification number of an **AEFACT** Bulk Data entry containing a list of spanwise divisions. [2]	Integer>0	Required
EQUAL NCHORD	Number of equally spaced chordwise divisions [2]	Integer>0	Required
SET,CSID	Identification number of an **AEFACT** Bulk Data entry containing a list of chordwise divisions. [2]	Integer>0	Required
IGID	Identification number of an interference group. [1]	Integer>0	**1**
Xi,Yi,Zi	Coordinates of the leading edge root and tip of the panel.	Real≥0.0	**0.0**
CROOT,CTIP	Chord length of the edge root and tip of the panel.	Real≥0.0	**0.0**

Remarks:

1. The interference group is a group of **CAERO1** panels that are aerodynamically coupled. Two **CAERO1** panels can be aerodynamically uncoupled by specifying different values of **IGID**.
2. The panel is divided into a spanwise and chordwise mesh of aerodynamic *box* elements. The mesh spacing may be defined in either of two ways. **NSPAN** and **NCHORD** are used to define an equally spaced mesh in the spanwise and chordwise directions, respectively. If you wish to define an unequally spaced mesh, then you specify either an **SSID**, a **CSID**, or both. These identification numbers reference **AEFACT** Bulk Data entries. These entries contain a list of implicit division points measured as a fraction of the total span or local chord.

Bulk Data Entry CBAR

Defines a BAR element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CBAR	EID	PID	GID1	GID2	V1	V2	V3		-cont-
					GID0				
-cont-	PINA	PINB	WA1	WA2	WA3	WB1	WB2	WB3	

CBAR	103	201	121	122	950				+A
+A		126							

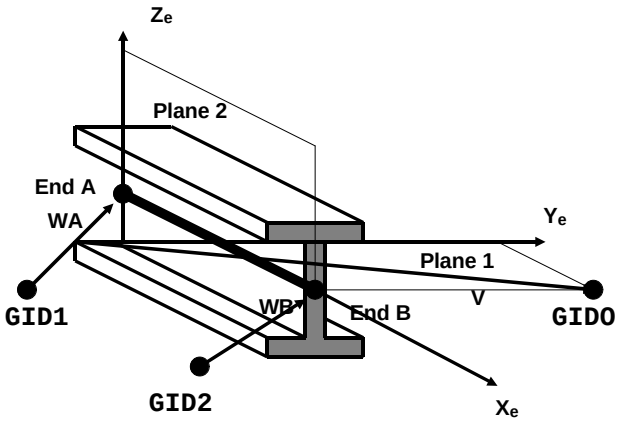
Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PBAR** or **PBAR1** property entry.	Integer>0	EID
GIDi	GRID point identification numbers of connection points.	Integer>0 **GID1≠GID2**	Required
Vi	Components of a vector, **V**, originating at End A. This vector, along with the vector from End A to End B, is used to determine the orientation of the element coordinate system. These vectors define the X-Y plane (also called Plane 1) of the element coordinate system. **Vi** must be specified in the output coordinate system for **GID1**.	Real	[2]
GID0	GRID point identification number used to define the element orientation. Direction of orientation vector is End A to **GID0**.	Integer>0	[2]
PINA, PINB	Pin flags for BAR Ends A and B respectively.	DOF Code	[3,4]
WAi, WBi	Components of offset vectors, measured in the displacement coordinate systems at GRID points **GID1** and **GID2**, from the GRID points to the end points of the axis of shear center.	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. If Field 6 is an integer, then the GRID point **GID0** is used to define the element coordinate system orientation. If Field 6 is blank or real, then the vector **V** is used. Fields 6, 7 and 8 may also be defined using a **BAROR** Bulk Data entry. If all of the Fields 6, 7 and 8 are blank on the **CBAR** entry, the corresponding data values are taken from the **BAROR** entry. The BAR element coordinate system orientation is illustrated in the figure on the following page.

7 BULK

3. The pin flags are used to remove connections between the GRID point and selected degrees of freedom of the BAR. The degrees of freedom are defined in the element coordinate system and the pin flags are applied at the offset ends of the BAR. The BAR must have stiffness associated with the pin flag. For example, if **PINA=4**, the **PBAR** entry must have a nonzero value for **J**, the torsional stiffness.
4. If there are no pin flags or offsets, the continuation entry may be omitted.



Bulk Data Entry CBEAM

Defines a BEAM element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CBEAM	EID	PID	GID1	GID2	V1	V2	V3		-cont-
					GID0				
-cont-	PINA	PINB	WA1	WA2	WA3	WB1	WB2	WB3	-cont-
-cont-	WIDA	WIDB							

CBEAM	2	39	7	3	1.0	1.0	0.5		+A
+A		513			3.0				+B
+B	8	5							

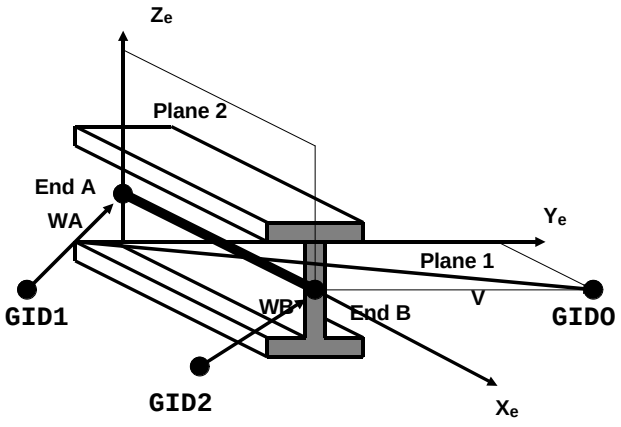
Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PBEAM** or **PBEAM1** property entry.	Integer>0	EID
GIDi	GRID point identification numbers of connection points.	Integer>0 **GID1≠GID2**	Required
Vi	Components of a vector, **V**, originating at End A. This vector, along with the vector from End A to End B, is used to determine the orientation of the element coordinate system. These vectors define the X-Y plane (also called Plane 1) of the element coordinate system. **Vi** must be specified in the output coordinate system for **GID1**.	Real	[2]
GID0	GRID point identification number used to define element orientation. Direction of orientation vector is End A to **GID0**.	Integer>0	[2]
PINA, PINB	Pin flags for BEAM Ends A and B respectively.	DOF Code	[7]
WAi, WBi	Components of offset vectors, measured in the displacement coordinate systems at GRID points **GID1** and **GID2**, from the GRID points to the end points of the axis of shear center.	Real	**0.0**
WIDA, WIDB	SCALAR or GRID point identification numbers for warping variables at Ends A and B.	Integer>0	[8]

Remarks:

1. Element identification numbers must be unique.

7 BULK

2. If Field 6 is an integer, then the GRID point **GID0** is used to define the element coordinate system orientation. If Field 6 is blank or real, then the vector **V** is used. Fields 6, 7 and 8 may also be defined using a **BEAMOR** Bulk Data entry. If all of the Fields 6, 7 and 8 are blank on the **CBEAM** entry, the corresponding data values are taken from the **BEAMOR** entry. The BEAM element coordinate system orientation is illustrated in the figure below.
3. **GID0** must be distinct from both End A and End B.
4. If there are no pin flags or offsets or warping variables, both continuation entries may be omitted.
5. The first continuation entry must be included, even if all fields are blank, if the second continuation entry is used.
6. If the second continuation entry is omitted, torsional stiffness due to warping of the cross-section will not be considered.
7. The pin flags are used to remove connections between the GRID point and selected degrees of freedom of the BEAM. The degrees of freedom are defined in the element coordinate system and the pin flags are applied at the offset ends of the BEAM. The BEAM must have stiffness associated with the pin flag. For example, if **PINA=4**, the **PBEAM** entry must have a nonzero value for **J**, the torsional stiffness.
8. If warping is specified, then **WIDA** and **WIDB** must be defined with **SPOINT** or **GRID** Bulk Data entries. If GRID data are used, the warping degree of freedom is attached to the first component, **T1** and the other 5 degrees of freedom must be constrained.
9. The BEAM element coordinate system orientation is shown in the figure below.



7 BULK

Bulk Data Entry CBUSH

Defines a BUSHing element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CBUSH	EID	PID	GID1	GID2	V1 GID0	V2	V3	CID	

CBUSH	101	1001	1	2					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PBUSH** property entry.	Integer>0	EID
GID1 **GID2**	GRID point identification numbers of connection points.	Integer>0 **GID1≠GID2**	Required
Vi	Components of a vector, **V**, originating at **GID1** that defines the orientation of the element. **Vi** must be specified in the output coordinate system for **GID1**.	Real	[2]
GID0	GRID point identification number used to define element orientation.	Integer>0	[2]
CID	Identification number of a coordinate system which defines the element orientation.	Integer>0	[2]

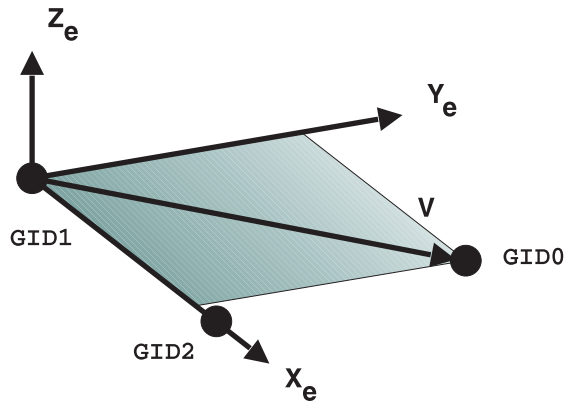
Remarks:

1. Element identification numbers must be unique.
2. The BUSH element orientation can be defined in several ways that depend on the location of the GRID points defining the element.

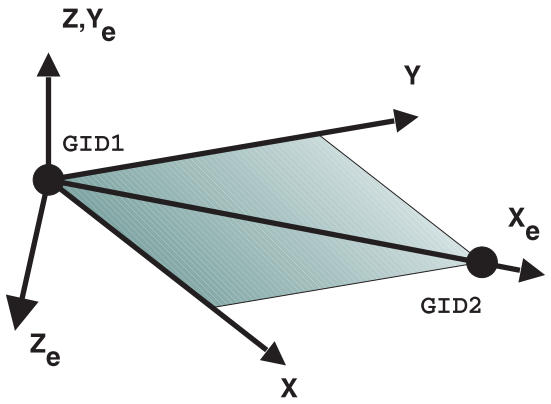
   When **GID1** and **GID2** are coincident, the element orientation and coordinate system may be defined in either of two ways. First, if no **CID** is specified, the Basic Coordinate System is used as the element coordinate system. If **CID** is specified, then coordinate system **CID** is used to define the element coordinate system. With either of these two cases, the element x-, y- and z-axis are the same as the x-, y- and z-axis of the **CID** (or Basic) coordinate system.

   If **GID1** and **GID2** are not coincident, then the element coordinate system is defined in the same manner as is done for the BAR and BEAM. This procedure requires definition of a vector, **V**, either by (1) specification of its components (**V1, V2, V3**) relative to **GID1** or by (2) defining the vector from **GID1** to **GID0**, as shown in Figure (a) on the following page.

   If you do not specify an orientation vector, **GID0**, or **CID**, then a default coordinate system is established as shown in Figure (b) on the next page. The element x-axis is taken to be the vector from **GID1** to **GID2**. The y-axis is then constructed perpendicularly to the x-axis in the plane of the x-axis and the Basic Coordinate system direction closest to normal to the element x-axis. The z-axis is then defined as the cross product of the x-axis and y-axis.



a. BUSH Element Orientation



b. Default Element Orientation

3. The BUSH element supports frequency dependent stiffness and damping properties in direct freqnency response analyses and nonlinear force-deflection relationship in Nonlinear Static analysis.
4. If you specify frequency dependent stiffness and damping properties for the element, all of the element degrees of freedom must be in the d-set.

Bulk Data Entry CCONEAX

Defines an axisymmetric conical shell element, CONEAX, for a structural model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CCONEAX	EID	PID	RID1	RID2					

CCONEAX	101	1002	101	102					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PCONEAX** property entry.	Integer>0	EID
RID1, RID2	Axisymmetric ring identification numbers. [2]	Integer>0 **RID1≠RID2**	Required

Remarks:

1. Element identification numbers must be unique.
2. Axisymmetric rings are defined with **RINGAX** Bulk Data entries.
3. Component degrees of freedom 4 and 6 are singular if transverse shear flexibility is not included on the **PCONEAX** entry.
4. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.
5. This element may not be used in heat transfer analysis; it may only be used in an axisymmetric harmonic structural analysis.

Bulk Data Entry CDAMP1

Defines a damping element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CDAMP1	EID	PID	GID1	DOF1	GID2	DOF2			
CDAMP1	19	6	22	2	23	2			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PDAMP** property entry.	Integer>0	EID
GIDi	GRID or SCALAR point identification numbers. [2,3]	Integer>0 **GID1≠GID2**	Required
DOFi	Single degree of freedom. [2,3]	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. The two connection points, (**GID1,DOF1**) and (**GID2,DOF2**), must be distinct.
3. If SCALAR points are used, the component number must be blank. If either pair (**GID1,DOF1**) or (**GID2,DOF2**) is blank, the damping element is placed between the indicated component and ground.
4. The additions to damping matrix BGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	GID1,DOF1	GID2,DOF2
GID1,DOF1	PID Value	-PID Value
GID2,DOF2	-PID Value	PID Value

Matrix Topology
One Connection Point

	GID1,DOF1	
GID1,DOF1	PID Value	

7 BULK

Bulk Data Entry CDAMP2

Defines a damping element and its damping value.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CDAMP2	EID	B	GID1	DOF1	GID2	DOF2			
CDAMP2	16	2.98	32	1					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
B	The damping value. [2]	Real	**0.0**
GIDi	GRID or SCALAR point identification numbers. [3,4]	Integer>0 **GID1≠GID2**	Required
DOFi	Single degree of freedom. [3,4]	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. This single entry completely defines the element since no material or geometric properties are required.
3. The two connection points, (**GID1,DOF1**) and (**GID2,DOF2**), must be distinct.
4. If SCALAR points are used, the component number must be blank. If either pair (**GID1,DOF1**) or (**GID2,DOF2**) is blank, the damping element is placed between the indicated component and ground. The additions to damping matrix BGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	GID1,DOF1	**GID2,DOF2**
GID1,DOF1	**B**	**-B**
GID2,DOF2	**-B**	**B**

Matrix Topology
One Connection Point

	GID1,DOF1	
GID1,DOF1	**B**	

7 BULK

Bulk Data Entry CDAMP3

Defines a damping element which is connected only to SCALAR points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CDAMP3	EID	PID	SID1	SID2					
CDAMP3	16	978	24	36					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PDAMP** property entry.	Integer>0	EID
SIDi	SCALAR point identification numbers. [2]	Integer>0 **SID1≠SID2**	Required

Remarks:

1. Element identification numbers must be unique.
2. **SID1** or **SID2** may be blank or zero indicating a connection to ground.
3. Additions to damping matrix BGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	SID1	SID2
SID1	PID Value	-PID Value
SID2	-PID Value	PID Value

Matrix Topology
One Connection Point

	SID1	
SID1	PID Value	

7 BULK

Bulk Data Entry CDAMP4

Defines a damping element which is connected only to SCALAR points and its property.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CDAMP4	EID	B	SID1	SID2					
CDAMP4	16	2.6	4	9					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
B	The damping value. [2]	Real	**0.0**
SIDi	SCALAR point identification numbers. [3]	Integer>0 **SID1≠SID2**	Required

Remarks:

1. Element identification numbers must be unique.
2. This entry completely defines the element since no material or geometric properties are required.
3. **SID1** or **SID2**, but not both, may be blank or zero indicating a connection to ground.
4. Additions to damping matrix BGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	SID1	SID2
SID1	B	-B
SID2	-B	B

Matrix Topology
One Connection Point

	SID1	
SID1	B	

7 BULK

Bulk Data Entry CELAS1

Defines a spring element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CELAS1	EID	PID	GID1	DOF1	GID2	DOF2			
CELAS1	2	6	8	1					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PELAS** property entry.	Integer>0	EID
GIDi	GRID or SCALAR point identification numbers. [2]	Integer>0 **GID1≠GID2**	Required
DOFi	Single degree of freedom. [2]	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. If SCALAR points are used, the component number must be blank. If either pair (**GID1,DOF1**) or (**GID2,DOF2**) is blank, the spring element is placed between the indicated component and ground.
3. Additions to stiffness matrix KGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	GID1,DOF1	GID2,DOF2
GID1,DOF1	PID Value	-PID Value
GID2,DOF2	-PID Value	PID Value

Matrix Topology
One Connection Point

	GID1,DOF1	
GID1,DOF1	PID Value	

7 BULK

Bulk Data Entry CELAS2

Defines a spring element and the spring value.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CELAS2	EID	K	GID1	DOF1	GID2	DOF2	GE	S	

CELAS2	28	6.2+5	32		19	4			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
K	The spring stiffness value.	Real	**0.0**
GIDi	GRID or SCALAR point identification numbers. [2]	Integer>0 **GID1≠GID2**	Required
DOFi	Single degree of freedom. [2]	DOF Code	Required
GE	Damping coefficient.	Real	**0.0**
S	Stress recovery coefficient. [3]	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. If SCALAR points are used, the component number must be blank. If either pair (**GID1,DOF1**) or (**GID2,DOF2**) is blank, the spring element is placed between the indicated component and ground.
3. The element stress is computed by multiplying the stress coefficient into the recovered element force.
4. This single entry completely defines the element since no material or geometric properties are required.
5. Additions to stiffness matrix KGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	GID1,DOF1	GID2,DOF2
GID1,DOF1	K	-K
GID2,DOF2	-K	K

Matrix Topology
One Connection Point

	GID1,DOF1	
GID1,DOF1	K	

7 BULK

Bulk Data Entry CELAS3

Defines a spring element which is connected only to SCALAR points.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CELAS3	EID	PID	SID1	SID2					

CELAS3	16	978	24	36					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PELAS** property entry.	Integer>0	EID
SIDi	SCALAR point identification numbers.	Integer>0 **SID1≠SID2**	[2]

Remarks:

1. Element identification numbers must be unique.
2. **SID1** or **SID2** may be blank or zero indicating a connection to ground.
3. Additions to stiffness matrix KGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	SID1	SID2
SID1	PID Value	-PID Value
SID2	-PID Value	PID Value

Matrix Topology
One Connection Point

	SID1	
SID1	PID Value	

7 BULK

Bulk Data Entry CELAS4

Defines a spring element and its stiffness value which is connected only to SCALAR points.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CELAS4	EID	K	SID1	SID2					

CELAS4	16	2.6	4	9					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
K	The spring stiffness value.	Real	**0.0**
SIDi	SCALAR point identification numbers.	Integer>0 **SID1≠SID2**	[2]

Remarks:

1. Element identification numbers must be unique.
2. **SID1** or **SID2**, but not both, may be blank or zero indicating a connection to ground.
3. This entry completely defines the element since no material or geometric properties are required.
4. Additions to stiffness matrix KGG are illustrated in the table below.

Matrix Topology
Two Connection Points

	SID1	SID2
SID1	K	-K
SID2	-K	K

Matrix Topology
One Connection Point

	SID1	
SID1	K	

7 BULK

Bulk Data Entry CELASNL

Defines a nonlinear spring element for a structural model.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CELASNL	EID	TNL K	GID1	DOF1	GID2	DOF2		S	

Referencing a Force-Deflection Curve:

CELASNL	32	12	1	1	2	1		0.02	

Using a Constant Stiffness:

CELASNL	32	1.5+5	1	1	2	1		0.02	

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
TNL	Table identification number that defines the element force-deflection curve.	Integer>0	[2,3]
K	Constant stiffness value.	Real	[2,4]
GIDi	GRID or SCALAR point identification numbers. [5]	Integer>0 **GID1≠GID2**	Required
DOFi	Single degree of freedom. [5]	DOF Code	Required
S	Stress recovery coefficient.	Real	**0.0**

Remarks:

1. All element identification numbers must be unique.
2. Either a force-deflection table identification number or a constant stiffness value must be entered.
3. The element force-deflection curves are specified with **TABLENL** Bulk Data entries. Stiffness is determined from these curves. The internal element force is computed as:

   $$f_{internal} = \mathbf{K}(\mathbf{TNL}) \cdot ((\mathbf{GID1,DOF1}) - (\mathbf{GID2,DOF2}))$$

   where **K**(**TNL**) is the stiffness value in field 3 or the slope of the curve defined by the **TABLENL** with **ID** specified in Field 3. (**GID1,DOF1**) is the displacement value of component **DOF1** at GRID **GID1** and similarly for **GID2**.
4. If a constant stiffness value is used, the element behaves linearly.
5. If SCALAR points are used, the component number must be blank. If either pair (**GID1,DOF1**) or (**GID2,DOF2**) is blank, the nonlinear spring element is placed between the indicated component and ground.

7 BULK

Bulk Data Entry CGAP

Defines a gap, cable, and/or friction element, GAP, of a structural model.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CGAP	EID	PID	GID1	GID2	V1	V2	V3	CID	
					GID0				

Using an Orientation Vector:

CGAP	101	200	1	2	0.707	0.707	0.0		

Using a Reference GRID point:

CGAP	101	200	1	2	9000				

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number.	Integer>0	Required
PID	Identification number of a **PGAP** property entry.	Integer>0	EID
GIDi	GRID point identification numbers of connection points.	Integer>0 **GID1≠GID2**	Required
Vi	Components of a vector **V**, originating at GRID point **GID1** that defines the element orientation.	Real	[3]
GID0	GRID point identification number used to define the element orientation.	Integer>0	[3]
CID	Identification number of a coordinate system that defines the element orientation.	Integer>0	[3]

Remarks:

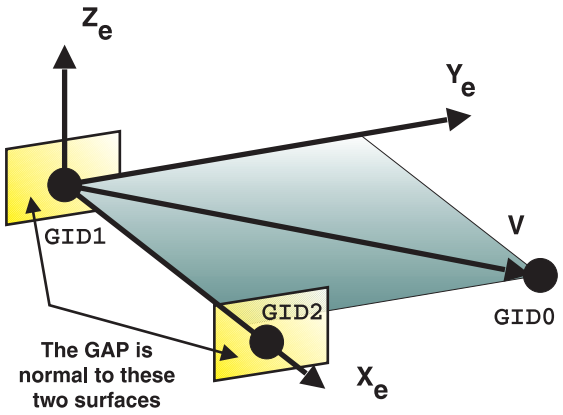
1. Element identification numbers must be unique.
2. This element may be used to simulate a gap and carry load only when closed, or it may be used to simulate a cable and carry only *tension* loads. This element may also slide freely or exhibit friction in the y-z plane of the element.
3. The GAP element orientation can be defined in several ways that depend on the location of the GRID points. These methods are described next and illustrated on the following page.

   When **GID1** and **GID2** are coincident (i.e. the distance between the GRID points within 10^{-4}), the element orientation and coordinate system may be defined in either of two ways. First, if no **CID** is specified, the Basic Coordinate System is used as the element coordinate system. If **CID** is specified, then coordinate system **CID** is used to define the element coordinate system. With either of these two cases, the element x-, y- and z-axis are the same as the x-, y- and z-axis of the **CID** (or Basic) coordinate system.

If **GID1** and **GID2** are not coincident and **CID** is not specified, then the element coordinate system is defined in the same manner as is done for the BAR and BEAM. This procedure requires definition of a vector, **V**, either by (1) specification of its components (**V1, V2, V3**) relative to **GID1** or by (2) defining the vector from **GID1** to **GID0**, as shown in the adjacent figure.

If **CID** is specified, then it is always used to define the element coordinate system. In this case, any values given for **Vi** or **GID0** are ignored.

The coordinate system must be defined such that the positive x-axis passes through the GAP ***element as shown in the figure.***



4. The element coordinate system does not rotate as a result of deflection.
5. The GAP element exhibits friction effects and has a bilinear force-deflection characteristic with **APPROACH NONLINEAR**.
6. GAP elements use the linear properties specified about the initial condition of the GAP with **APPROACH DISP** in linear static and dynamic analyses.
7. Initial GAP element openings are defined on the **PGAP** Bulk Data entry and not by the physical separation between **GID1** and **GID2**.
8. There are no element stresses, but element forces may be requested with the Case Control commands **FORCE** or **NLFORCE**. These are output in the element coordiante system, and positive values of F_x indicate compression.

Bulk Data Entry CHACAB

Defines a three-dimensional acoustic absorber, HACAB, element for fluid-structure interaction analyses.

Format and Example:

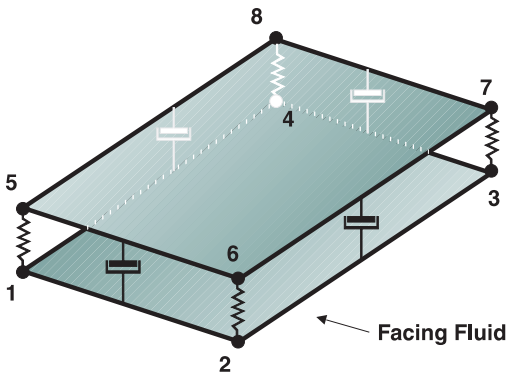
1	2	3	4	5	6	7	8	9	10
CHACAB	EID	PID	GID1	GID2	GID3	GID4	GID5	GID6	-cont-
-cont-	GID7	GID8	GID9	GID10	GID11	GID12			-cont-
-cont-			GID17	GID18	GID19	GID20			

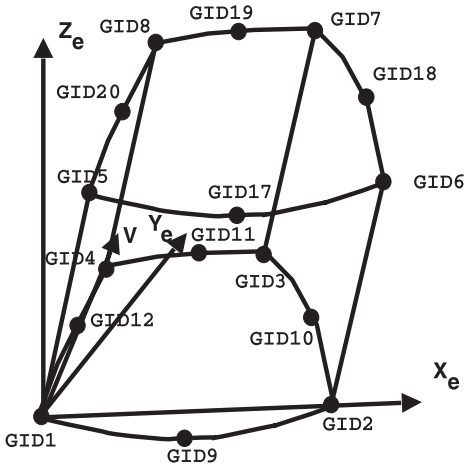
CHACAB	95	12	1	2	5	7	8	9	+A
+A	24	23							

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PACABS** property entry. [2]	Integer>0	EID
GIDi	GRID point identification numbers defining the element.	Integer>0 **GIDi≠GIDj**	[3,4]

Remarks:

1. Element identification numbers must be unique.
2. The CHACAB element geometry, coordinate system and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis is normal to X_e in the plane defined by X_e and the vector **V** from **GID1** to **GID4**. Finally, the element z-axis, Z_e, is computed as the vector product of X_e and Y_e.
3. The order of node sequencing may be clockwise (opposite of what is shown here), but each edge must consist of the nodes as defined in the figure. Note that if a midside node is omitted, then the midside node opposite must also be omitted.
4. The element face defined by **GID1** through **GID4** is assumed to be in contact with the fluid as shown in the figure below.





7 BULK

Bulk Data Entry CHACBR

Defines a three-dimensional acoustic barrier, HACBR, element for fluid-structure interaction analyses.

Format and Example:

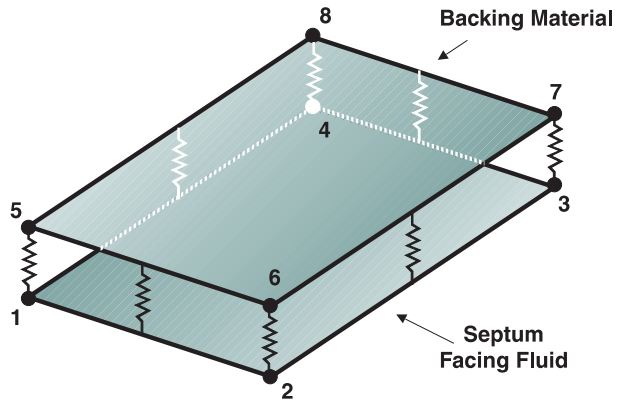
1	2	3	4	5	6	7	8	9	10
CHACBR	EID	PID	GID1	GID2	GID3	GID4	GID5	GID6	-cont-
-cont-	GID7	GID8	GID9	GID10	GID11	GID12			-cont-
-cont-			GID17	GID18	GID19	GID20			

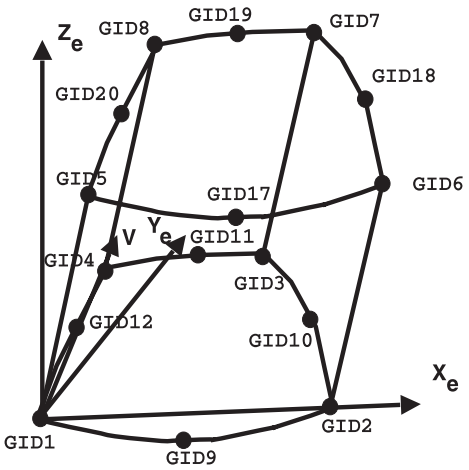
CHACBR	1001	101	1	2	3	4	11	12	+A
+A	13	14							

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PACBAR** property entry. [2]	Integer>0	EID
GIDi	GRID point identification numbers defining the element.	Integer>0 **GIDi≠GIDj**	[3,4]

Remarks:

1. Element identification numbers must be unique.
2. The CHACBR element geometry, coordinate system and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis is normal to X_e in the plane defined by X_e and the vector **V** from **GID1** to **GID4**. Finally, the element z-axis, Z_e, is computed as the vector product of X_e and Y_e.
3. The order of node sequencing may be clockwise (opposite of what is shown here), but each edge must consist of the nodes as defined in the figure. Note that if a midside node is omitted, then the midside node opposite must also be omitted.
4. The element face defined by **GID1** through **GID4** is assumed to be in contact with the fluid, and the element face defined by **GID5** through **GID8** is assumed to be the backing material.





Bulk Data Entry CHBDY

Defines a boundary element, HBDY, which is used for heat flux, thermal vector flux, convection and / or radiation in a heat transfer model.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CHBDY	EID	PID	TYPE	GID1	GID2	GID3	GID4	VIEWID	-cont-
-cont-	GIDA1	GIDA2	GIDA3	GIDA4	V1	V2	V3		

CHBDY	721	9001	LINE	202	199			20	+A
+A	203	204			0.0	1.0	0.0		

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PHBDY** entry.	Integer>0	Required
TYPE	Boundary area shape.	Character { **POINT** **LINE** **REV** **AREA3** **AREA4** **ELCYL** }	Required
GIDi	GRID point identification numbers of primary connected points.	Integer>0	**0**
GIDAi	GRID or SCALAR point identification numbers of associated ambient points.	Integer>0	**0**
Vi	Vector in the Basic Coordinate System used for element orientation.	Real	**0.0**
VIEWID	Identification number of a **VIEW** entry which will be used to control radiation effects.	Integer or Character	[3]

Remarks:

1. Element identification numbers must be unique.
2. The continuation entry is not required.

7 BULK

3. The value of **VIEWID** may be an integer, a character string, or blank depending on the type of radiation effects that you are modeling.

   If **VIEWID=0** or blank, then this HBDY element will not participate in radiation exchange with other HBDY elements.

   If **VIEWID=SPACE**, this element radiates to a black body, and no **VIEW** Bulk Data entry is referenced or required.

   If **VIEWID>0**, this element will participate in radiation exchange with other HBDY elements. The **VIEW** entry with identification number of **VIEWID** controls the radiation view factor calculations for this element.

   If **VIEWID<0**, this is the same for **VIEWID>0** except that the active side of the element types **AREA3** and **AREA4** is reversed. The absolute value of **VIEWID** is the **VIEW** identification number. For a description of the available **TYPE**s, see the ***User's Guide***.

4. A property entry, **PHBDY**, is used to define the associated area factors, the emissivity, the absorptivity, and the principal radii of an elliptic cylinder. The material coefficients used for convection and thermal capacity are also referenced by this entry.

5. The associated points, **GIDAi**, may be either GRID or SCALAR points, and are used to define the ambient temperature for a convection field. These points correspond to the primary points **GIDi**, and the number of them depends on the **TYPE** option, but they need not be unique. Their values may be set in static analysis with SPC data, or they may be connected to other elements. If any field is blank, the ambient temperature associated with that GRID point is assumed to be zero.

6. Heat flux may be applied to this element with **QBDY1** or **QBDY2** Bulk Data entries.

7. Thermal vector flux from a directional source may be applied to this element with **QVECT** Bulk Data entries.

8. **VIEWID** is not available for **TYPE**=**ELCYL**.

Bulk Data Entry CHEXA

Defines a three-dimensional isoparametric hexahedron, HEXA, solid or fluid finite element.

Format and Example:

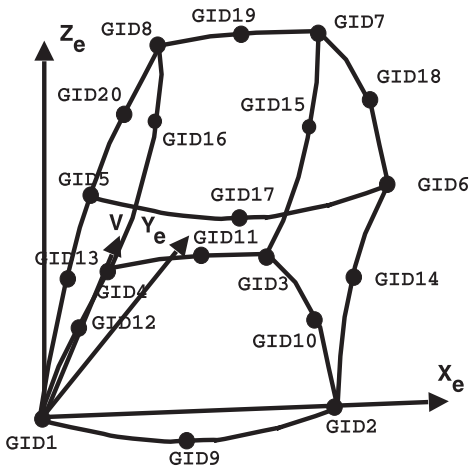
1	2	3	4	5	6	7	8	9	10
CHEXA	EID	PID	GID1	GID2	GID3	GID4	GID5	GID6	-cont-
-cont-	GID7	GID8	GID9	GID10	GID11	GID12	GID13	GID14	-cont-
-cont-	GID15	GID16	GID17	GID18	GID19	GID20			

CHEXA	1001	101	21	25	35	46	19	87	+A
+A	129	203	421				872	100	+B
+B	1265	1456			1732				

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSOLID** property entry. [2]	Integer>0	EID
GIDi	GRID point identification numbers defining the element.	Integer>0 **GIDi≠GIDj**	[3,4]

Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the **PSOLID** entry.
3. The midside GRID points are optional and any or all may be left blank. The second continuation entry is not required if the midside GRID points are omitted.
4. It is recommended that the midside GRID points be located within the middle third of an edge. If the midside GRID is located precisely at the quarter point, a numerical instability will occur.
5. The HEXA element geometry, coordinate system and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis is normal to X_e in the plane defined by X_e and the vector **V** from **GID1** to **GID4**. Finally, the element z-axis, Z_e, is computed as the vector product of X_e and Y_e.
6. The rotation of node sequencing may be clockwise (opposite of what is shown here), but each edge must consist of the nodes as defined in the figure.
7. The HEXA element may be used as a fluid element, and in this case only the 8 corner nodes may be defined.



7 BULK

Bulk Data Entry CMASS1

Defines a mass element for a structural model.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CMASS1	EID	PID	GID1	DOF1	GID2	DOF2			
CMASS1	101	1001	1201	1					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PMASS** property entry.	Integer>0	EID
GIDi	GRID or SCALAR point identification numbers.	Integer>0 **GID1≠GID2**	[2,3]
DOFi	Single degree of freedom. [2,3]	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. The two connection points, (**GID1,DOF1**) and (**GID2,DOF2**), must be distinct.
3. If SCALAR points are used, the component number must be blank. If either pair (**GID1,DOF1**) or (**GID2,DOF2**) is blank, the mass element is placed between the indicated component and ground.
4. The additions to mass matrix MGG are illustrated in the table below. Note, unlike the case of damping or stiffness, mass element additions are normally performed with only one connection point.

Matrix Topology
Two Connection Points

	GID1,DOF1	GID2,DOF2
GID1,DOF1	PID Value	-PID Value
GID2,DOF2	-PID Value	PID Value

Matrix Topology
One Connection Point

	GID1,DOF1	
GID1,DOF1	PID Value	

7 BULK

Bulk Data Entry CMASS2

Defines a mass element and its value for a structural model.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CMASS2	EID	M	GID1	DOF1	GID2	DOF2			
CMASS2	901	6.75	201	2					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
M	The value of the mass.	Real	**0.0**
GIDi	GRID or SCALAR point identification numbers.	Integer>0 **GID1≠GID2**	[2,3]
DOF1 **DOF2**	Single degree of freedom. [2,3]	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. The two connection points, (**GID1,DOF1**) and (**GID2,DOF2**), must be distinct.
3. If SCALAR points are used, the component number must be blank. If either pair (**GID1,DOF1**) or (**GID2,DOF2**) is blank, the mass element is placed between the indicated component and ground.
4. This single entry completely defines the element since no material or geometric properties are required.
5. The additions to mass matrix MGG are illustrated in the table below. Note, unlike the case of damping or stiffness, mass element additions are normally performed with only one connection point.

Matrix Topology
Two Connection Points

	GID1,DOF1	GID2,DOF2
GID1,DOF1	M	-M
GID2,DOF2	-M	M

Matrix Topology
One Connection Point

	GID1,DOF1	
GID1,DOF1	M	

7 BULK

Bulk Data Entry CMASS3

Defines a mass element for a structural model which is connected only to SCALAR points.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CMASS3	EID	PID	SID1	SID2					

CMASS3	123	555	101	1					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number.	Integer>0	Required
PID	Identification number of a **PMASS** property entry.	Integer>0	EID
SIDi	SCALAR point identification numbers.	Integer>0 **SID1≠SID2**	[2]

Remarks:

1. Element identification numbers must be unique.
2. **SID1** or **SID2** may be blank or zero indicating a connection to ground.
3. The additions to mass matrix MGG are illustrated in the table below. Note, unlike the case of damping or stiffness, mass element additions are normally performed with only one connection point.

Matrix Topology
Two Connection Points

	SID1	SID2
SID1	PID Value	-PID Value
SID2	-PID Value	PID Value

Matrix Topology
One Connection Point

	SID1	
SID1	PID Value	

7 BULK

Bulk Data Entry CMASS4

Defines for a structural model a mass element and its value which is connected only to SCALAR points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CMASS4	EID	M	SID1	SID2					
CMASS4	104	13.6	101	102					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
M	The mass value.	Real	**0.0**
SIDi	SCALAR point identification numbers. [2]	Integer>0 **SID1≠SID2**	**0**

Remarks:

1. Element identification numbers must be unique.
2. **SID1** or **SID2**, but not both, may be blank or zero indicating a connection to ground.
3. This entry completely defines the element since no material or geometric properties are required.
4. The additions to mass matrix MGG are illustrated in the table below. Note, unlike the case of damping or stiffness, mass element additions are normally performed with only one connection point.

Matrix Topology
Two Connection Points

	SID1	SID2
SID1	M	-M
SID2	-M	M

Matrix Topology
One Connection Point

	SID1	
SID1	M	

7 BULK

Bulk Data Entry CONCT

Defines the GRID point and degree of freedom connectivities between two substructures for a manual COMBINE operation.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CONCT	SID	DOF	SNAME1	SNAME2					-cont-
-cont-	GID11	GID21	GID12	GID22	*CONTINUES IN GROUPS OF 2*				-cont-

CONCT	307	1246	WING	FUSLG					+A
+A	201	207	957	214	917	216			

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of connectivity set. [1,2]	Integer>0	Required
DOF	List of degrees of freedom. [3]	DOF Code	Required
SNAME1, SNAME2	Names of Basic Substructures being connected. [4]	Name	Required
GID1i, GID2i	GRID or SCALAR point identification numbers where **GID1i** from **SNAME1** connects to **GID2i** from **SNAME2** at the degrees of freedom specified by **DOF**.	Integer>0	Required

Remarks:

1. The connectivity set is in the Substructure Control command Packet with the **CONNECT** subcommand of the **COMBINE** command:

```
COMBINE SUBA,SUBB
   CONNECT = 101
```

2. Any number of **CONCT** and **CONCT1** Bulk Data entries may use the same value of **SID**.
3. Component **DOF** specified with **CONCT** data are overridden by **RELES** Bulk Data entries.
4. Each **SNAMEi** must be a component Basic Substructure of one of the substructures being combined as specified on the substructure **COMBINE** command.
5. At least one continuation entry must be present.
6. See also the **CONCT1** Bulk Data entry.
7. The **CONCT** and **CONCT1** Bulk Data entries define an explicit substructure connectivity mapping. GRID points entered in the corresponding field of a substructure name specify the manner in which the substructures are connected.

7 BULK

Bulk Data Entry CONCT1

Defines the GRID point and degree of freedom connectivities between two or more substructures for a manual COMBINE operation.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CONCT1	SID	SNAME1	SNAME2	SNAME3	SNAME4	SNAME5	SNAME6	SNAME7	-cont-
-cont-	DOF1	GID11	GID12	GID13	GID14	GID15	GID16	GID17	-cont-
-cont-	DOF2	GID21	GID22	GID23	GID24	GID25	GID26	GID27	-cont-

CONCT1	805	WINGR	FUSLG	MIDWG	POD				+A
+A	123	528	17	32	106				+B
+B	46		518		108				

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of connectivity set. [1,2]	Integer>0	Required
SNAMEi	Basic Substructure names. [3]	Name	Required
DOFi	List of degrees of freedom. [4]	DOF Code	Required
GIDij	GRID or SCALAR point identification numbers in **SNAMEj** which are connected at **DOFi**.	Integer>0	Required

Remarks:

1. The connectivity set is selected in the Substructure Control Packet with the **CONNECT** subcommand of the **COMBINE** command:

```
COMBINE SUBA,SUBB
   CONNECT = 101
```

2. Any number of **CONCT** and **CONCT1** Bulk Data entries may use the same value of **SID**.
3. Each **SNAMEi** must be a component Basic Substructure of one of the substructures being combined as specified on the substructure **COMBINE** command.
4. Component **DOF** specified with **CONCT1** data are not overridden by **RELES** Bulk Data entries.
5. At least one continuation entry must be present.
6. See also the **CONCT** Bulk Data entry.
7. The **CONCT** and **CONCT1** Bulk Data entries define an explicit substructure connectivity mapping. GRID points entered in the corresponding field of a substructure name specify the manner in which the substructures are connected.

7 BULK

Bulk Data Entry CONM1

Defines a 6x6 symmetric mass matrix at a GRID point for a structural model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CONM1	EID	GID	CID	M11	M21	M22	M31	M32	-cont-
-cont-	M33	M41	M42	M43	M44	M51	M52	M53	-cont-
-cont-	M54	M55	M61	M62	M63	M64	M65	M66	

CONM1	103	1	0	3.4		3.4			+A
+A	3.4				23.6				+B
+B		23.6						23.6	

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GID	GRID point identification number.	Integer>0	Required
CID	Coordinate system identification number for the mass matrix.	Integer>0	**0**
Mij	Mass matrix values. [2]	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. The form of the mass matrix defined at GRID point **GID** by this entry is:

$$M = \begin{bmatrix} \mathbf{M11} & & & & & \\ \mathbf{M21} & \mathbf{M22} & & & & \\ \mathbf{M31} & \mathbf{M32} & \mathbf{M33} & & -Sym- & \\ \mathbf{M41} & \mathbf{M42} & \mathbf{M43} & \mathbf{M44} & & \\ \mathbf{M51} & \mathbf{M52} & \mathbf{M53} & \mathbf{M54} & \mathbf{M55} & \\ \mathbf{M61} & \mathbf{M62} & \mathbf{M63} & \mathbf{M64} & \mathbf{M65} & \mathbf{M66} \end{bmatrix}$$

The mass terms are defined in coordinate system **CID**.

3. See also the **CONM2** Bulk Data entry.

7 BULK

Bulk Data Entry CONM2

Defines a concentrated mass at a GRID point of a structural model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CONM2	EID	GID	CID	M	V1	V2	V3		-cont-
-cont-	I11	I12	I22	I13	I23	I33			

CONM2	103	116	0	.45					+A
+A	18.4		12.6			6.5			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GID	GRID point identification number.	Integer>0	Required
CID	Coordinate system identification number. [2]	Integer>0	**0**
M	Mass Value.	Real	**0.0**
Vi	Offset distances to the center of gravity of the mass from **GID** in the coordinate system **CID**.	Real	**0.0**
Iij	Mass moments of inertia measured at the center of gravity of the mass, in coordinate system **CID**. [2]	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. The mass matrix is defined in coordinate system **CID**.
3. The form of the mass matrix generated at GRID point **GID** is:

$$M = \begin{bmatrix} M & & & 0 & V3 \cdot M & -V2 \cdot M \\ & M & & -V1 \cdot M & 0 & V1 \cdot M \\ & & M & V2 \cdot M & -V1 \cdot M & 0 \\ & & & I11+(V2^2+V3^2) \cdot M & -I12-M \cdot V1 \cdot V2 & -I13-M \cdot V1 \cdot V3 \\ & & & & I22+(V1^2+V3^2) \cdot M & -I32-M \cdot V2 \cdot V3 \\ & & -Sym- & & & I33+(V1^2+V2^2) \cdot M \end{bmatrix}$$

4. See also the **CONM1** Bulk Data entry.

7 BULK

Bulk Data Entry CONROD

Defines a rod element and its properties.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CONROD	EID	GID1	GID2	MID	A	J	C	NSM	
CONROD	103	117	128	500	9.82				

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GIDi	GRID point identification numbers of connection points.	Integer>0 **GID1≠GID2**	Required
MID	Material identification number. [2]	Integer>0	Required
A	Area of rod.	Real>0.0	Required
J	Torsional constant.	Real	**0.0**
C	Coefficient for torsional stress determination.	Real	**0.0**
NSM	Nonstructural mass per unit length.	Real	**0.0**

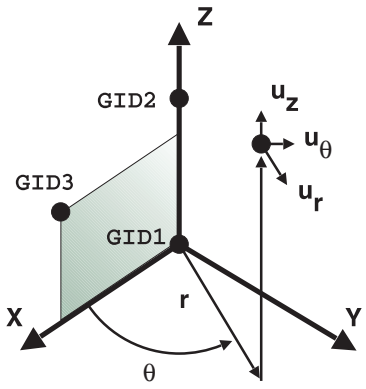
Remarks:

1. Element identification numbers must be unique.
2. For structural problems, **CONROD** entries may only reference **MAT1** material data, and for heat transfer problems, **CONROD** entries may only reference **MAT4** or **MAT5** material data.

7 BULK

Bulk Data Entry CORD1C

Defines a cylindrical coordinate system by reference to three GRID points. These points must be defined in coordinate systems whose definition does not involve the coordinate system being defined. As shown in the figure, the first point defines the origin of the new system. The second point defines the direction of the z-axis of the new system. The third point locates the plane of the aximuthal origin of the new system. The reference coordinate system must be independently defined.



Format and Example:

1	2	3	4	5	6	7	8	9	10
CORD1C	CID	GID1	GID2	GID3					

CORD1C	101	51	52	53					

Field	*Contents*	*Data Range*	*Default*
CID	Coordinate system identification number. [1]	Integer>0	Required
GIDi	GRID point identification numbers. [2]	Integer>0 **GID1≠GID2≠GID3**	Required

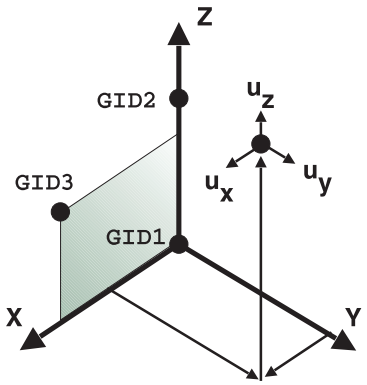
Remarks:

1. Coordinate system identification numbers on all **CORD1R**, **CORD1C**, **CORD1S**, **CORD2R**, **CORD2C**, and **CORD2S** entries must all be unique.
2. The three GRID points must be noncollinear.
3. The location of a GRID point P in this coordinate system is given by (r, θ, z) where is θ measured in degrees.
4. The displacement coordinate directions at P are dependent on the location of P as shown above by (u_r, u_θ, u_z)
5. Points on the z-axis may not have their displacement directions defined in this coordinate system since an ambiguity results.

7 BULK

Bulk Data Entry CORD1R

Defines a rectangular coordinate system by reference to three GRID points. These points must be defined in coordinate systems whose definition does not involve the coordinate system being defined. As shown in the figure, the first point defines the origin of the new system. The second point defines the direction of the z-axis of the new system. The third point locates the x-y plane of the new system. The reference coordinate system must be independently defined.



Format and Example:

1	2	3	4	5	6	7	8	9	10
CORD1R	CID	GID1	GID2	GID3					
CORD1R	101	1001	1002	1003					

Field	*Contents*	*Data Range*	*Default*
CID	Coordinate system identification number. [1]	Integer>0	Required
GIDi	GRID point identification numbers. [2]	Integer>0 **GID1≠GID2≠GID3**	Required

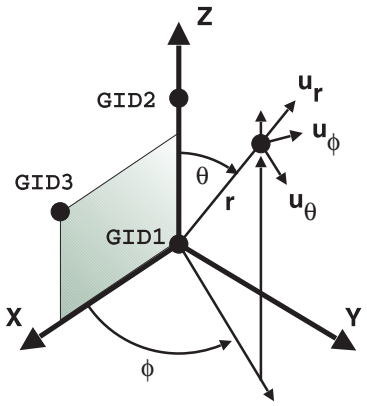
Remarks:

1. Coordinate system identification numbers on all **CORD1R**, **CORD1C**, **CORD1S**, **CORD2R**, **CORD2C**, and **CORD2S** entries must all be unique.
2. The three GRID points must be noncollinear.
3. The location of a GRID point P in this coordinate system is given by (x, y, z).
4. The displacement coordinate directions at P are shown above by (u_x, u_y, u_z).

7 BULK

Bulk Data Entry CORD1S

Defines a spherical coordinate system by reference to three GRID points. These points must be defined in coordinate systems whose definition does not involve the coordinate system being defined. As shown in the figure, the first point defines the origin of the new system. The second point defines the direction of the z-axis of the new system. The third point locates the plane of the aximuthal origin of the new system. The reference coordinate system must be independently defined.



Format and Example:

1	2	3	4	5	6	7	8	9	10
CORD1S	CID	GID1	GID2	GID3					
CORD1S	101	101	102	103					

Field	*Contents*	*Data Range*	*Default*
CID	Coordinate system identification number. [1]	Integer>0	Required
GIDi	GRID point identification numbers. [2]	Integer>0 **GID1≠GID2≠GID3**	Required

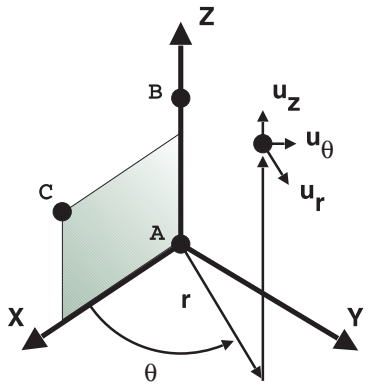
Remarks:

1. Coordinate system identification numbers on all **CORD1R**, **CORD1C**, **CORD1S**, **CORD2R**, **CORD2C**, and **CORD2S** entries must all be unique.
2. The three GRID points must be noncollinear.
3. The location of a GRID point P in this coordinate system is given by (r, θ, ϕ) where θ and ϕ are measured in degrees.
4. The displacement coordinate directions at P are dependent on the location of P as shown above by (u_r, u_θ, u_ϕ).
5. Points on the polar axis may not have their displacement directions defined in this coordinate system since an ambiguity results.

7 BULK

Bulk Data Entry CORD2C

Defines a cylindrical coordinate system using the coordinates of three points located in an independently defined reference coordinate system. As shown in the figure, the first point, A, locates the origin of the new coordinate system, the second point, B, defines the direction of the z-axis of the new system, and the third point, C, locates the plane of the azimuthal origin. The reference coordinate system must be independently defined.



Format and Examples:

1	2	3	4	5	6	7	8	9	10
CORD2C	CID	CIDREF	A1	A2	A3	B1	B2	B3	-cont-
-cont-	C1	C2	C3						

CORD2C	199	101	-2.9	1.0	0.0	3.6	0.0	1.0	+A
+A	5.2	1.0	-2.9						

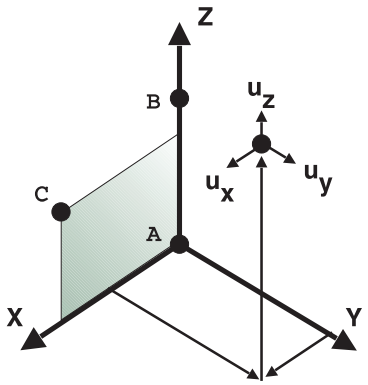
Field	*Contents*	*Data Range*	*Default*
CID	Coordinate system identification number. [1]	Integer>0	Required
CIDREF	Independently defined reference coordinate system which is used to locate and orient the new coordinate system.	Integer>0	**0**
Ai,Bi,Ci	Coordinates of three points in coordinate system **CIDREF**. [2]	Real	**0.0**

Remarks:

1. Coordinate system identification numbers must be unique.
2. The three points A, B and C must be unique and noncollinear.
3. The location of GRID point P is given by (r, θ, z) where θ is measured in degrees, and the displacement coordinate directions (u_r, u_θ, u_z) at P are dependent on the location of P.
4. Points on the z-axis may not have their displacement direction defined in this coordinate system since an ambiguity results.

Bulk Data Entry CORD2R

Defines a rectangular coordinate system using the coordinates of three points located in an independently defined reference coordinate system. As shown in the figure, the first point, A, locates the origin of the new coordinate system, the second point, B, defines the direction of the z-axis of the new system, and the third point, C, locates the x-z plane of the new system. The reference coordinate system must be independently defined.



Format and Examples:

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

CORD2R	10		1.0	1.0	0.0	3.0	0.0	1.0	+A
+A	1.5	1.5	-2.0						

Field	*Contents*	*Data Range*	*Default*
CID	Coordinate system identification number. [1]	Integer>0	Required
CIDREF	Independently defined reference coordinate system which is used to locate and orient the new coordinate system.	Integer>0	**0**
Ai,Bi,Ci	Coordinates of three points in coordinate system **CIDREF**. [2]	Real	**0.0**

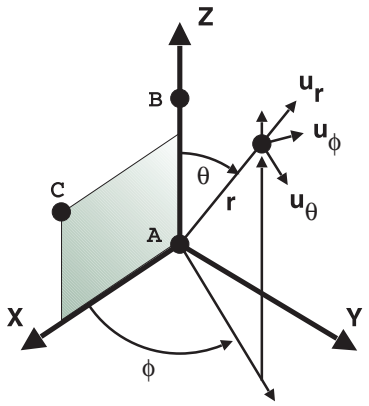
Remarks:

1. Coordinate system identification numbers must be unique. Continuation entry must be present.
2. The three points A, B and C must be unique and noncollinear.
3. The location of a GRID point P is given by (x, y, z).
4. The displacement coordinate directions (u_r, u_y, u_z) at P are dependent on the location of P.
5. The continuation entry is always required.

7 BULK

Bulk Data Entry CORD2S

Defines a spherical coordinate system using the coordinates of three points located in an independently defined reference coordinate system. As shown in the figure, the first point, A, locates the origin of the new coordinate system, the second point, B, defines the direction of the z-axis of the new system, and the third point, C, locates the aximuthal plane of the new system. The reference coordinate system must be independently defined.



Format and Examples:

1	2	3	4	5	6	7	8	9	10
CORD2S	CID	CIDREF	A1	A2	A3	B1	B2	B3	-cont-
-cont-	C1	C2	C3						

CORD2R	10		1.0	1.0	0.0	3.0	0.0	1.0	+A
+A	1.5	1.5	-2.0						

Field	*Contents*	*Data Range*	*Default*
CID	Coordinate system identification number. [1]	Integer>0	Required
CIDREF	Independently defined reference coordinate system which is used to locate and orient the new coordinate system.	Integer>0	**0**
Ai,Bi,Ci	Coordinates of three points in coordinate system **CIDREF**. [2]	Real	**0.0**

Remarks:

1. Coordinate system identification numbers must be unique.
2. The three points A, B and C must be unique and noncollinear.
3. The location of GRID point P is given by (r, θ, ϕ) where θ and ϕ are measured in degrees, and the displacement coordinate directions (u_r, u_θ, u_ϕ) at P are dependent on the location of P.
4. Points on the polar axis may not have their displacement direction defined in this coordinate system since an ambiguity results.

7 BULK

Bulk Data Entry CPENTA

Defines a three-dimensional isoparametric pentahedron, PENTA, solid or fluid element.

Format and Example:

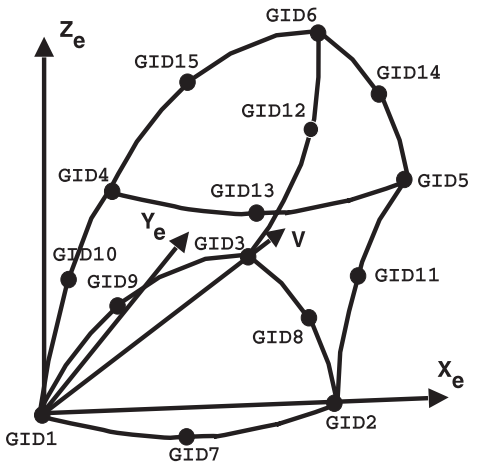
1	2	3	4	5	6	7	8	9	10
CPENTA	EID	PID	GID1	GID2	GID3	GID4	GID5	GID6	-cont-
-cont-	GID7	GID8	GID9	GID10	GID11	GID12	GID13	GID14	-cont-
-cont-	GID15								

CPENTA	1001	101	21	25	35	46	19	87	+A
+A	129	203	421				872	100	+B
+B	1265								

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSOLID** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	[3,4]

Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the **PSOLID** entry.
3. The corner GRID points are required. The midside GRID points are optional and any or all may be left blank. The continuation entry is not required if the midside GRIDs are omitted.
4. It is recommended that the midside GRID points be located within the middle third of an edge. If the midside GRID is located precisely at the quarter point, a numerical instability will occur.
5. The PENTA element geometry, coordinate system and numbering are shown in the adjoining figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the plane defined by X_e and the vector V from **GID1** to **GID3**. Finally, the element z-axis, Z_e, is computed as the vector product of X_e and Y_e.
6. The rotation of node sequencing may be clockwise (opposite of what is shown here), but each edge must consist of the nodes as defined in the figure.
7. The PENTA element may be used as a fluid element, and in this case only the 6 corner nodes may be defined.



7 BULK

Bulk Data Entry CPILE

Defines a one-dimensional line element PILE.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CPILE	EID	PID	PSOIL	GID1	GID2	V1 GIDO	V2	V3	-cont-
-cont-	PINA	PINB	ZA1	ZA2	ZA3	ZB1	ZB2	ZB3	

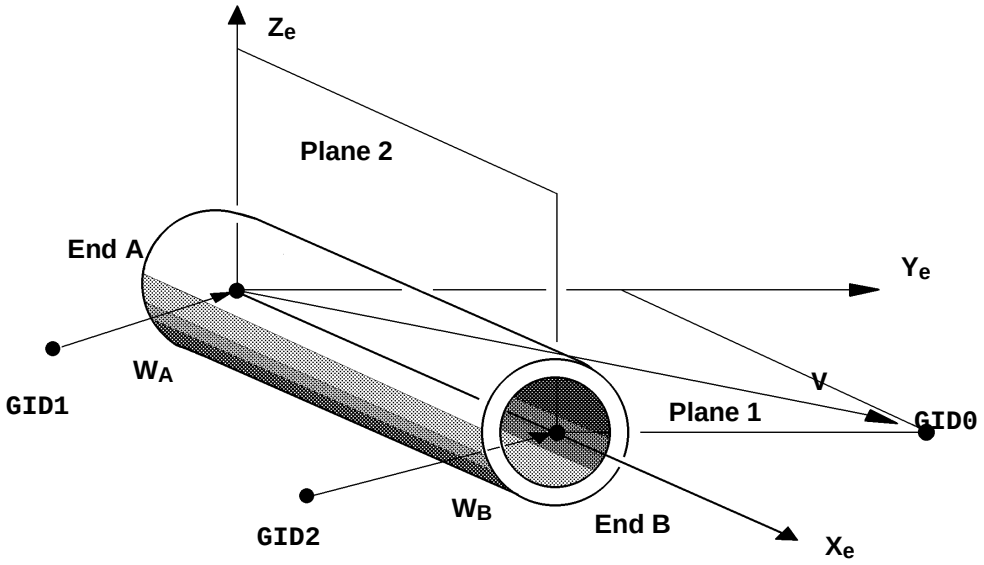
CPILE	100	10	200	3	4	0.0	1.0	0.0	+A
+A									

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PPILE** or **PPILE1** property entry.	Integer>0	EID
PSOIL	Identification number of a **PSOIL** property entry.	Integer>0	[2]
GIDi	Identification numbers of GRID or SCALAR points defining the element.	Integer>0 **GID1≠GID2**	Required
Vi	Components of a vector, **V**, originating at End A. This vector, along with the vector from End A to End B, is used to determine the orientation of the element coordinate system. These vectors define the X-Y plane (also called Plane 1) of the element coordinate system. **Vi** must be specified in the output coordinate system for **GID1**.	Real	[3]
GIDO	GRID point identification number used to define the element coordinate system orientation. The irection of the orientation vector is End A to **GIDO**.	Integer>0	[3]
PINA, PINB	Pin flags for PILE Ends A and B, respectively.	DOF Code	[4,5]
ZAi, ZBi	Components of offset vectors, measured in the displacement coordinate systems at GRID points **GID1** and **GID2**, from the GRID points to the end points of the axis of the shear center. [5]	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. If **PIDS** is blank or zero, there is no soil attached to the PILE element.
3. If Field 7 is an integer, then the GRID point **GIDO** is used to define the element coordinate system orientation as shown in the figure on the following page. If Field 7 is blank or real, then the vector **V** is used. Fields 7, 8 and 9 may also be defined using a **PILEOR** Bulk Data entry. Any of these fields which are blank on the **CPILE** entry take the value from the **PILEOR** entry.

4. The pin flags are used to move connections between the GRID point and selected degrees of freedom of the PILE. The degrees of freedom are defined in the element's coordinate system and the pin flags are applied at the offset ends of the beam. The PILE must have stiffness associated with the pin flag. For example, if **PINA=4**, the **PPILE1** entry must have a non-zero value for **J**, the torsional stiffness.
5. If there are no pin flags or offsets, the continuation entry may be omitted.
6. The PILE element coordinate system orientation and offsets are shown in the figure.



Bulk Data Entry CPIPE

Defines a one-dimensional curved element PIPE.

Format and Example:

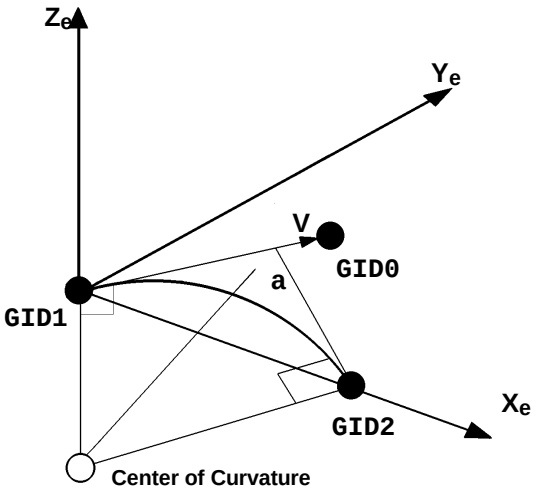
1	2	3	4	5	6	7	8	9	10
CPIPE	EID	PID	GID1	GID2	V1	V2	V3		-cont-
					GID0				
-cont-	PINA	PINB							

CPIPE	56	45	121	122	1.0	0.0	0.0		

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number.	Integer>0	Required
PID	Identification number of a **PPIPE** property entry.	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GID1≠GID2**	Required
GIDO	GRID point identification number used to define the tangent vector.	Integer>0	[2,3]
Vi	Components of tangent vector in the Basic Coordinate System.	Real	[2,3]
PINA, PINB	Pin flags for PIPE Ends A and B, respectively, that are used to ensure that the PIPE cannot resist a force or moment corresponding to the pin flag at that respective end.	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. The center of curvature is calculated as the intersection of a line perpendicular to the tangent at **GID1**, with the normal to the line from **GID1** to **GID2** at its mid-point. GRID point **GID0** may typically be input as the intersection of the tangents at the two ends of the element.
3. The angle α between the tangents at **GID1** and **GID2** must satisfy $0°<\alpha<180°$.



Bulk Data Entry CQUAD4

Defines a two-dimensional linear quadrilateral shell finite element QUAD4.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CQUAD4	EID	PID	GID1	GID2	GID3	GID4	θm MCSID	ZOFF	-cont-
-cont-			T1	T2	T3	T4			

CQUAD4	55	1	210	211	311	310	35.	0.05	+A
+A			0.01	0.01	0.03	0.03			

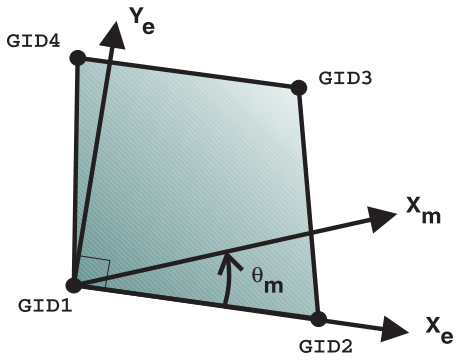
Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSHELL** or **PCOMPi** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	Required
ZOFF	Offset of the element reference plane from the plane of GRID points. A positive value means in the $+Z_e$ direction. [3,5]	Real	**0.0**
θm	Material property orientation angle in degrees. [3]	Real	**0.0**
MCSID	Material coordinate system identification number. The material x-axis of the element is in the direction of the projection of the x-axis of this system on the surface of the element.	Integer>0	[3]
Ti	Membrane thickness of the element at GRID point **GIDi**.	Real>0.0	[4]

Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the property entry.
3. The material coordinate system **MCSID** and the offset **ZOFF** may also be provided on the **PSHELL** entry. The **PSHELL** data will be used if the corresponding field on the **CQUAD4** entry is blank.
4. The **Ti** are optional, if not supplied they will be set to the value of **T** specified on the **PSHELL** entry. In such cases, the continuation entry is not required.

5. The QUAD4 element geometry, coordinate systems and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the plane defined by X_e and **GID4**. The material angle, θ_m, defines the material x-axis, X_m. The element reference plane is located at the mid-thickness of the element parallel to the element mean plane.

6. All interior angles must be less than 180°.



Bulk Data Entry CQUAD8

Defines a quadratic isoparametric curved quadrilateral shell finite element QUAD8.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CQUAD8	EID	PID	GID1	GID2	GID3	GID4	GID5	GID6	-cont-
-cont-	GID7	GID8	T1	T2	T3	T4	θm / MCSID	ZOFF	-cont-
-cont-			T5	T6	T7	T8			

CQUAD8	88	101	101	102	103	104	105	106	+A
+A	107	108	0.03	0.125	0.05	0.125	101	0.25	+B
+B			0.03	0.125	0.05	0.125			

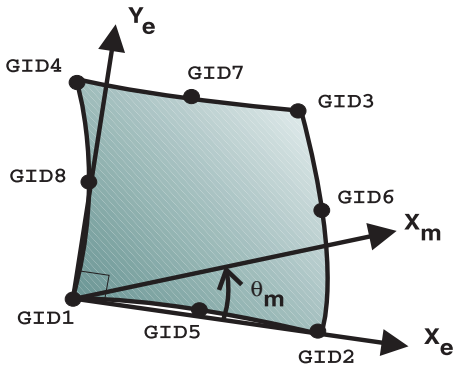
Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSHELL** or **PCOMPi** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element. [3]	Integer>0 **GIDi≠GIDj**	Required
ZOFF	Offset of the element reference plane from the plane of GRID points. A positive value means in the $+Z_e$ direction. [4,6]	Real	**0.0**
θm	Material property orientation angle in degrees. [4]	Real	**0.0**
MCSID	Material coordinate system identification number. The material x-axis of the element is in the direction of the projection of the x-axis of this system on the surface of the element.	Integer>0	[4]
Ti	Membrane thickness of the element at GRID point **GIDi**.	Real>0.0	[5]

Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the property entry.
3. The midside GRID points are optional and any or all may be left blank. It is recommended that the midside GRID points be located within the middle third of an edge. If the midside GRID is located precisely at the quarter point, a numerical instability will occur.
4. The material coordinate system **MCSID** and the offset **ZOFF** may also be provided on the **PSHELL** entry. The **PSHELL** data will be used if the corresponding field on the **CQUAD8** entry is blank.
5. The $\mathbf{T_i}$ are optional, if not supplied they will be set to the value of **T** specified on the **PSHELL** entry. In such cases, the two continuation entries may not be required.

6. The QUAD8 element geometry, coordinate systems and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the in the plane defined by X_e and **GID4**. The material angle, θ_m, defines the material x-axis, X_m. The element reference plane is located at the mid-thickness of the element parallel to the element mean plane.

7. All corner point interior angles must be less than 180°.



Bulk Data Entry CQUADR

Defines a two-dimensional linear quadrilateral shell finite element with drilling degrees of freedom, the QUADR.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CQUADR	EID	PID	GID1	GID2	GID3	GID4	θm MCSID	ZOFF	-cont-
-cont-			T1	T2	T3	T4			

CQUADR	55	1	210	211	311	310	35.	0.05	+A
+A			0.01	0.01	0.03	0.03			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSHELL** or **PCOMPi** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	Required
ZOFF	Offset of the element reference plane from the plane of GRID points. A positive value means in the $+Z_e$ direction. [3,5]	Real	**0.0**
θm	Material property orientation angle in degrees. [3]	Real	**0.0**
MCSID	Material coordinate system identification number. The material x-axis of the element is in the direction of the projection of the x-axis of this system on the surface of the element.	Integer>0	[3]
Ti	Membrane thickness of the element at GRID point **GIDi**.	Real>0.0	[4]

Remarks:

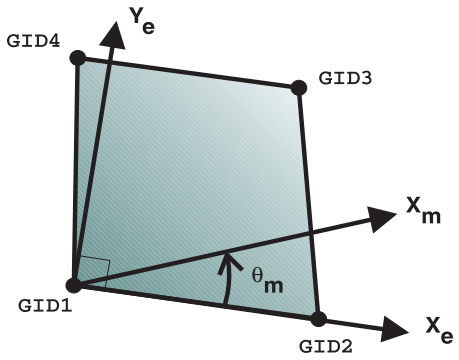
1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the property entry.
3. The material coordinate system **MCSID** and the offset **ZOFF** may also be provided on the **PSHELL** entry. The **PSHELL** data will be used if the corresponding field on the **CQUADR** entry is blank.
4. The **Ti** are optional, if not supplied they will be set to the value of **T** specified on the **PSHELL** entry. In such cases, the continuation entry is not required.

7 BULK

5. The QUADR element geometry, coordinate systems and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the plane defined by X_e and **GID4**. The material angle, θ_m, defines the material x-axis, X_m. The element reference plane is located at the mid-thickness of the element parallel to the element mean plane.

6. All interior angles must be less than 180°.

7. Because the QUADR has "drilling" degrees of freedom, it is not necessary to SPC the normal rotation of the GRID points.



Bulk Data Entry CROD

Defines a one-dimensional tension-compression-torsion element.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CROD	EID	PID	GID1	GID2					

CROD	113	114	122	124					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number.	Integer>0	Required
PID	Identification number of a **PROD** property entry.	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GID1≠GID2**	Required

Remarks:

1. Element identification numbers must be unique.
2. See also the **CONROD** Bulk Data entry.

7 BULK

Bulk Data Entry CSHEAR

Defines a two-dimensional shear panel finite element.

Format and Example:

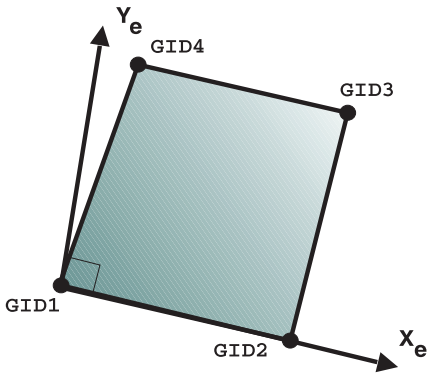
1	2	3	4	5	6	7	8	9	10
CSHEAR	EID	PID	GID1	GID2	GID3	GID4			

CSHEAR	104	107	102	106	104	108			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSHEAR** property entry.	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0	Required

Remarks:

1. Element identification numbers must be unique.
2. The SHEAR element geometry, coordinate system, and GRID point numbering are shown in the figure. All interior angles must be less than 180°.



7 BULK

Bulk Data Entry CTETRA

Defines a three-dimensional isoparametric tetrahedron, TETRA, solid or fluid element.

Format and Example:

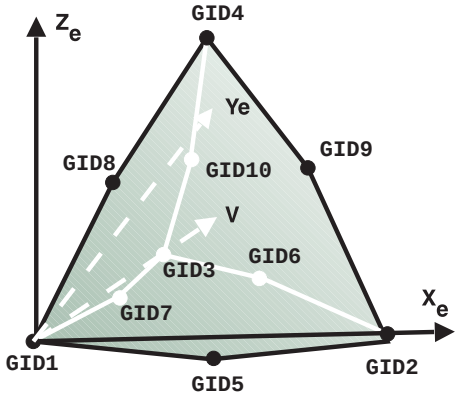
1	2	3	4	5	6	7	8	9	10
CTETRA	EID	PID	GID1	GID2	GID3	GID4	GID5	GID6	-cont-
-cont-	GID7	GID8	GID9	GID10					

CTETRA	1001	101	21	25	35	46	19	87	+A
+A	129	203	421						

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSOLID** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element. [3,4]	Integer>0 **GIDi≠GIDj**	Required

Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the **PSOLID** entry.
3. The midside GRID points are optional and any or all may be left blank. The continuation entry is not required if the last four midside GRID points are omitted.
4. It is recommended that the midside GRID points be located within the middle third of an edge. If the midside GRID point is located precisely at the quarter point, a numerical instability will occur.
5. The TETRA geometry, coordinate system and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the plane defined by X_e and the vector V from **GID1** to **GID3**. Finally, the element z-axis, Z_e, is computed as the vector product of X_e and Y_e.
6. The rotation of node sequencing may be clockwise (opposite of what is shown here), but each edge must consist of the nodes as defined in the figure.
7. The TETRA element may be used as a fluid element, and in this case only the 4 corner nodes may be defined.



7 BULK

Bulk Data Entry CTORDRG

Defines an axisymmetric toroidal cross-section ring (shell of revolution) element.

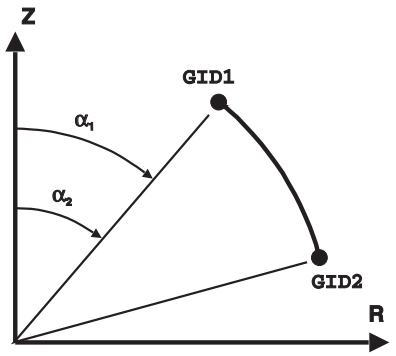
Format and Example:

1	2	3	4	5	6	7	8	9	10
CTORDRG	EID	PID	GID1	GID2	$\alpha 1$	$\alpha 2$			
CTORDRG	1059	601	1006	1012	32.0	65.2			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Property identification number.	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element. [2]	Integer>0 **GID1≠GID2**	Required
$\alpha 1,\alpha 2$	Angles of curvature at **GID1** and **GID2**, in degrees. [3]	$0° \le \alpha \le 180°$ $\alpha 2 \ge \alpha 1$	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. GRID points **GID1** and **GID2** must lie in the x-z plane of the Basic Coordinate System and to the right of the axis of symmetry (the z-axis).
3. To model a shell cap, you specify $\alpha 1=0.0$.



7 BULK

Bulk Data Entry CTRAPAX

Defines an axisymmetric trapezoidal cross-section ring (solid of revolution) element with nonaxisymmetric loading and deflection capability.

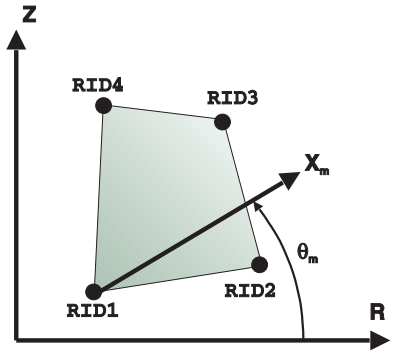
Format and Example:

1	2	3	4	5	6	7	8	9	10
CTRAPAX	EID	PID	RID1	RID2	RID3	RID4	θm		
CTRAPAX	115	105	110	111	112	113	39.6		

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number.	Integer>0	Required
PID	Identification number of a **PTRAPAX** entry.	Integer>0	EID
RIDi	Axisymmetric ring identification numbers. [2]	Integer>0 **RIDi≠RIDj**	Required
θm	Material property orientation angle in degrees.	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. Axisymmetric rings are defined with **RINGAX** Bulk Data entries.
3. The TRAPAX geometry, material coordinate system and RINGAX sequence numbering are shown in the figure. This element cannot be modeled with any point on the axis of symmetry.
4. The RINGAX points must be located using positive values of **R**. All interior angles must be less than 180 degrees. The material property orientation angle is measured from the radial direction as illustrated in the figure. There are no other restrictions on GRID point locations.
5. This element may not be used in heat transfer analysis; it may only be used in an axisymmetric harmonic structural analysis.



7 BULK

Bulk Data Entry CTRAPRG

Defines an axisymmetric trapezoidal cross-section ring (solid of revolution) element.

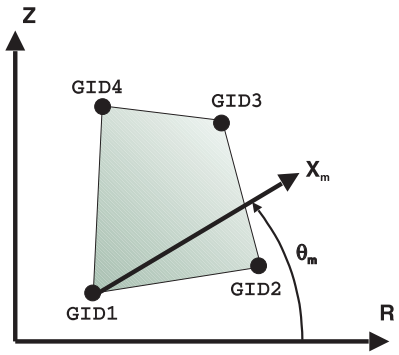
Format and Example:

1	2	3	4	5	6	7	8	9	10
CTRAPRG	EID	GID1	GID2	GID3	GID4	θm	MID		
CTRAPRG	72	13	14	15	16	29.2	100		

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	Required
θm	Material property orientation angle in degrees.	Real	**0.0**
MID	Material property identification number. [2]	Integer>0	Required

Remarks:

1. Element identification numbers must be unique.
2. For structural analyses, **CTRAPRG** entries may only reference **MAT1** or **MAT3** material data, and for heat transfer analyses, they may only reference **MAT4** or **MAT5** material data.
3. The TRAPRG geometry, coordinate systems and numbering is shown in the figure. The four GRID points must lie in the r-z plane of both the Basic and any local coordinate systems. The radial coordinates of all GRID points must be positive. The line connecting GRID points **GID1** and **GID2** and the line connecting GRID points **GID3** and **GID4** must both be normal to the z-axis. This restriction does not apply to heat transfer models.



7 BULK

Bulk Data Entry CTRIA3

Defines a two-dimensional linear triangular shell finite element TRIA3.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CTRIA3	EID	PID	GID1	GID2	GID3	θm MCSID	ZOFF		-cont-
-cont-			T1	T2	T3				

CTRIA3	101	17	1001	1005	1010	35.	0.05		+A
+A			0.03	0.125	0.05				

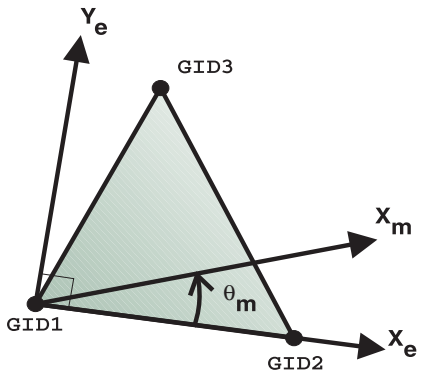
Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSHELL** or **PCOMPi** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	Required
ZOFF	Offset of the element reference plane from the plane of GRID points. A positive value means in the $+Z_e$ direction. [3,5]	Real	[3]
θm	Material property orientation angle in degrees. [3]	Real	**0.0**
MCSID	Material coordinate system identification number. The material x-axis of the element is in the direction of the projection of the x-axis of this system on the surface of the element.	Integer>0	[3]
Ti	Membrane thickness of the element at GRID point **GIDi**.	Real>0.0	[4]

Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the property entry.
3. The material coordinate system **MCSID** and the offset **ZOFF** may also be provided on the **PSHELL** entry. The **PSHELL** data will be used if the corresponding field on the **CTRIA3** entry is blank.
4. The **Ti** are optional, if not supplied they will be set to the value of **T** specified on the **PSHELL** entry. In such cases, the continuation entry is not required.

7 BULK

5. The TRIA3 geometry, coordinate systems and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the plane of the element. The material angle, θ_m, defines the material x-axis, X_m. The element reference plane is located at the mid-thickness of the element parallel to the element mean plane.



Bulk Data Entry CTRIA6

Defines an isoparametric curved triangular shell finite element, TRIA6.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CTRIA6	EID	PID	GID1	GID2	GID3	GID4	GID5	GID6	-cont-
-cont-	θm MCSID	ZOFF	T1	T2	T3	T4	T5	T6	

CTRIA6	300	107	101	105	110	113	116	127	+A
+A	101	0.1	0.03	0.125	0.05	0.125			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSHELL** or **PCOMPi** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element. [3,4]	Integer>0 **GIDi≠GIDj**	
ZOFF	Offset of the element reference plane from the plane of GRID points. A positive value means in the $+Z_e$ direction. [5,7]	Real	[3]
θm	Material property orientation angle in degrees. [5]	Real	**0.0**
MCSID	Material coordinate system identification number. The material x-axis of the element is in the direction of the projection of the x-axis of this system on the surface of the element.	Integer>0	[5]
Ti	Membrane thickness of the element at GRID point **GIDi**.	Real>0.0	[6]

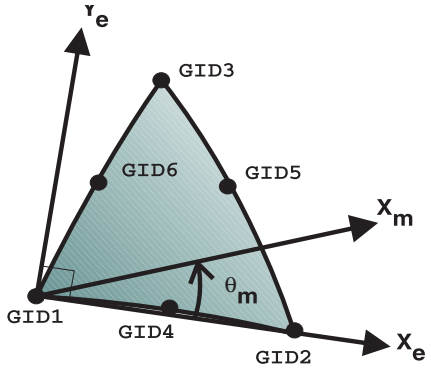
Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the property entry.
3. The midside GRID points are optional and any or all may be left blank.
4. It is recommended that the midside GRID points be located within the middle third of an edge. If the midside GRID is located precisely at the quarter point, a numerical instability will occur.
5. The material coordinate system **MCSID** and the offset **ZOFF** may also be provided on the **PSHELL** entry. The **PSHELL** data will be used if the corresponding field on the **CTRIA6** entry is blank.
6. The **Ti** are optional, if not supplied they will be set to the value of **T** specified on the **PSHELL** entry. In such cases, the continuation entry may not be required.

7 BULK

7. The TRIA6 geometry, coordinate systems and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the plane defined by X_e and the vector from **GID1** to **GID3**. The material angle, θ_m, defines the material x-axis, X_m. The element reference plane is located at the mid-thickness of the element parallel to the element mean plane.

8. All corner point interior angles must be less than 180°.



Bulk Data Entry: CTRIAR

Defines a two-dimensional linear triangular shell finite element with "drilling" degrees of freedom, the TRIA3.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CTRIAR	EID	PID	GID1	GID2	GID3	θm MCSID	ZOFF		-cont-
-cont-			T1	T2	T3				

CTRIAR	101	17	1001	1005	1010	35.	0.05		+A
+A			0.03	0.125	0.05				

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PSHELL** or **PCOMPi** property entry. [2]	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	Required
ZOFF	Offset of the element reference plane from the plane of GRID points. A positive value means in the $+Z_e$ direction. [3,5]	Real	[3]
θm	Material property orientation angle in degrees. [3]	Real	**0.0**
MCSID	Material coordinate system identification number. The material x-axis of the element is in the direction of the projection of the x-axis of this system on the surface of the element.	Integer>0	[3]
Ti	Membrane thickness of the element at GRID point **GIDi**.	Real>0.0	[4]

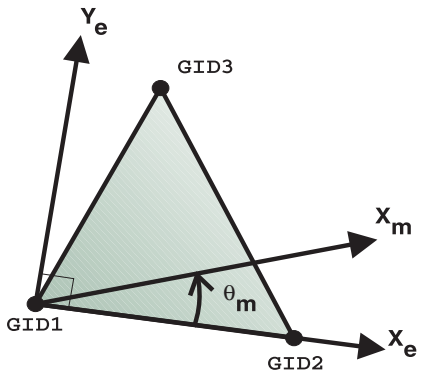
Remarks:

1. Element identification numbers must be unique.
2. Components of stress are output in the stress coordinate system defined by the property entry.
3. The material coordinate system **MCSID** and the offset **ZOFF** may also be provided on the **PSHELL** entry. The **PSHELL** data will be used if the corresponding field on the **CTRIAR** entry is blank.
4. The **Ti** are optional, if not supplied they will be set to the value of **T** specified on the **PSHELL** entry. In such cases, the continuation entry is not required.

7 BULK

5. The TRIAR geometry, coordinate systems and numbering are shown in the figure. The element x-axis, X_e, is defined by the vector from **GID1** to **GID2**. The element y-axis, Y_e, is normal to X_e in the plane of the element. The material angle, θ_m, defines the material x-axis, X_m. The element reference plane is located at the mid-thickness of the element parallel to the element mean plane.

6. Because the TRIAR has "drilling" degrees of freedom, it is not necessary to SPC the normal rotation of the GRID points.



Bulk Data Entry CTRIAAX

Defines an axisymmetric triangular cross-section ring (solid of revolution) element with nonaxisymmetric loading capability.

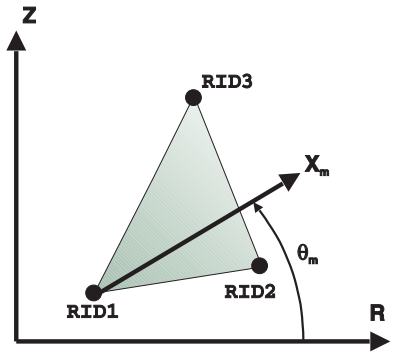
Format and Example:

1	2	3	4	5	6	7	8	9	10
CTRIAAX	EID	PID	RID1	RID2	RID3	θm			
CTRIAAX	121	116	143	162	203	42.0			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PTRIAAX** property entry.	Integer>0	EID
RIDi	Axisymmetric ring identification numbers.	Integer>0 **RIDi≠RIDj**	Required
qm	Material property orientation angle in degrees.	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. Axisymmetric rings are defined with **RINGAX** Bulk Data entries.
3. The **TRIAAX** geometry, coordinate system and numbering are shown in the figure. The GRID points must lie in the r-z plane of both the Basic and any local coordinate systems. The radial coordinates of all GRID points must be positive.
4. This element may not be used in heat transfer analysis; it may only be used in an axisymmetric harmonic structural analysis.



7 BULK

Bulk Data Entry CTRIARG

Defines an axisymmetric triangular cross-section ring (solid of revolution) element and its properties.

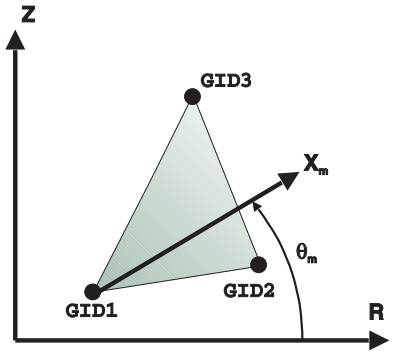
Format and Example:

1	2	3	4	5	6	7	8	9	10
CTRIARG	EID	GID1	GID2	GID3	θm	MID			
CTRIARG	117	1001	1002	1003	14.6	400			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	Required
θm	Material property orientation angle in degrees.	Real	**0.0**
MID	Material identification number. [2]	Integer>0	Required

Remarks:

1. Element identification numbers must be unique.
2. For structural analyses, **CTRIARG** entries may only reference **MAT1** or **MAT3** material data, and for heat transfer analyses, they may only reference **MAT4** or **MAT5** material data.
3. The TRIARG geometry, coordinate system and numbering are shown in the figure. The GRID points must lie in the r-z plane of both the Basic and any local coordinate systems. The radial coordinates of all GRID points must be positive.



7 BULK

Bulk Data Entry CTUBE

Defines a one-dimensional tension-compression-torsion element, TUBE.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CTUBE	EID	PID	GID1	GID2					

CTUBE	51	101	561	562					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number.	Integer>0	Required
PID	Identification number of a **PTUBE** property entry.	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GID1≠GID2**	Required

Remarks:

1. Element identification numbers must be unique.

7 BULK

Bulk Data Entry CTWIST

Defines a two-dimensional twist panel finite element, TWIST.

Format and Example:

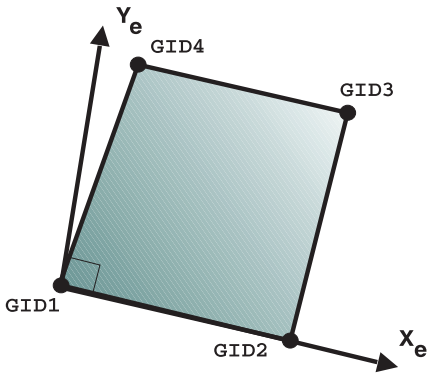
1	2	3	4	5	6	7	8	9	10
CTWIST	EID	PID	GID1	GID2	GID3	GID4			

CTWIST	55	201	101	102	103	104			

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of a **PTWIST** property entry.	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GIDi≠GIDj**	Required

Remarks:

1. Element identification numbers must be unique.
2. The TWIST geometry, coordinate system and numbering are shown in the figure.



7 BULK

Bulk Data Entry CVISC

Defines a viscous damper element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CVISC	EID	PID	GID1	GID2					
CVISC	121	1002	29	31					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
PID	Identification number of **PVISC** property entry.	Integer>0	EID
GIDi	Identification numbers of GRID points defining the element.	Integer>0 **GID1≠GID2**	Required

Remarks:

1. Element identification numbers must be unique.

7 BULK

Bulk Data Entry CYAX

Defines GRID points that lie on the axis of symmetry in cyclic symmetry analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CYAX	*GRID ID LIST*								-cont-
CYAX	1	5	THRU	25	30	31			

Field	*Contents*	*Data Range*	*Default*
GRID ID LIST	List of one or more GRID point identification numbers that are located on the axis of symmetry.	Integer>0 **GIDi≠GIDj**	Required

Remarks:

1. The coordinate system for a GRID point lying on the axis of symmetry must be a rectangular system with the z-component of motion aligned with the axis of symmetry.
2. If the dihedral symmetry option, **DIH** is selected on the **CYSYM** entry, the y-axis must be perpendicular to Side 1.
3. GRID points lying on the axis of symmetry may be constrained by **SPC**s but not by **MPC**s. If the number of segments is greater than or equal to 3, **SPC**s must be applied to both components 1 and 2 or to neither, and **SPC**s must be applied to both components 4 and 5 or to neither, in order to satisfy symmetry. In addition, the degrees of freedom (not constrained by **SPC**s) at these GRID points must be in the analysis set (a-set).
4. GRID points lying on the axis of symmetry must not be defined on Side 1 or Side 2 by means of a **CYJOIN** entry.
5. A range of GRID point identifiers may be specified with the **THRU** option. However, the **THRU** may not appear in Fields 2 or 9.

7 BULK

Bulk Data Entry CYJOIN

Defines the boundary points of a segment in cyclic symmetry analyses.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
CYJOIN	SIDE	NORM	*GRID ID LIST*						-cont-
CYJOIN	1	T2	101	451	475	489	502	679	+A
+A	734	891							

Field	*Contents*	*Data Range*	*Default*
SIDE	Side identification number. [1]	Integer {1, 2}	Required
NORM	Direction normal to symmetry boundary at **GIDi**.	Character {T1, T2, T3}	[2]
GRID ID LIST	List of one or more GRID or SCALAR point identification numbers.	Integer>0	**0**

Remarks:

1. For rotationally symmetric models there must be one logical entry for **SIDE=1** and one for **SIDE=2.** The two lists specify the GRID or SCALAR points to be connected. For models with dihedral symmetry, **SIDE=1** refers to the boundary between segments and **SIDE=2** refers to the middle of a segment.
2. The GRID point degree-of-freedom which is normal to the boundary must be specified as **NORM**. For SCALAR points with one degree-of-freedom, **NORM** is left blank if they are to have the same sign, and **NORM=T1** if they are opposite in sign. The directions are defined in the output coordinate system.
3. All points defined by the range must exist.
4. All components of displacement at boundary points are connected to adjacent segments except those constrained by **SPC**, **MPC** or **OMIT** Bulk Data entries.
5. **CYJOIN** Bulk Data entries are used only for cyclic symmetry analyses.

7 BULK

Bulk Data Entry CYSYM

Selects parameters for cyclic symmetry analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
CYSYM	NSEG	STYPE							
CYSYM	8	ROT							

Field	*Contents*	*Data Range*	*Default*
NSEG	Number of segments.	Integer>0	Required
STYPE	Symmetry type selector. [1]	Character { **ROT** } { **DIH** }	Required

Remarks:

1. The symmetry types are rotational symmetry, **ROT**, and dihedral symmetry, **DIH**.

7 BULK

Bulk Data Entry DAREA

Specifies the location of a dynamic load or enforced motion and a scale factor.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DAREA	SID	GID1	DOF1	A1	GID2	DOF2	A2		
DAREA	99	6	1	12.2	19	5	4.7		

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of **DAREA** set. [1]	Integer>0	Required
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	Single degree of freedom.	DOF Code	Required
Ai	Scale factor for the designated degree of freedom.	Real	Required

Remarks:

1. The **DAREA** data are selected by referencing the set identification number, **SID**, on either a **RLOAD1** or **RLOAD2** Bulk Data entry (frequency response) or **TLOAD1** or **TLOAD2** entry (transient response).

7 BULK

Bulk Data Entry DAREAS

Specifies the location of a dynamic load or enforced motion and a scale factor in reference to a Basic Substructure.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DAREAS	SID	SNAME	GID1	DOF1	A1	GID2	DOF2	A2	
DAREAS	101	WING	1001	2	23.8	1002	2	9.06	

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of **DAREA** set. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	Single degree of freedom.	DOF Code	Required
Ai	Scale factor for the designated degree of freedom.	Real	Required

Remarks:

1. The **DAREAS** data are selected by referencing the set identification number, **SID**, on either a **RLOAD1** or **RLOAD2** Bulk Data entry (frequency response) or **TLOAD1** or **TLOAD2** entry (transient response).
2. The **DAREAS** data are used only in the substructuring SOLVE operation.
3. Points referenced must exist in the substructure to be SOLVEd.

7 BULK

Bulk Data Entry DCDYNRG

Defines a dynamic response constraint, either structural, fluid or acoustic, on the combined amplitudes of one or more degrees of freedom in the model. Either the average amplitude over a range of frequencies or the amplitude at each analysis frequency may be constrained.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DCDYNRG	DCSID	DCNAME	TYPE	FORM	LLIM LLTAB	ULIM ULTAB	LRANGE	URANGE	-cont-
-cont-	GID1	COMP1	VAL1	GID2	COMP2	VAL2			-cont-
-cont-	GID3	COMP3	VAL3	*CONTINUES IN GROUPS OF 3*					-cont-

DCDYNRG	100	FREQR	VELO	PEAK	1.02		0.0	100.0	+A
+A		1001	T1	1.0	1002	T1	2.5		

Field	*Contents*	*Data Range*	*Default*
DCSID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labeling. [2]	Character	Blank
TYPE	Specifies the response to be constrained. [3]	Character { **DISP** **VELO** **ACCEL** **PRES** **DB** **DBA** }	Blank
FORM	Form of the constraint. May be **AVG** to constrain the average amplitude across the entire frequency range, or **PEAK** to request that the response be constrained at each frequency. [4]	Character { **AVG** **PEAK** }	Required
LLIM	Specifies the lower limit of the constraint value. [4]	Real≥0.0	Required
LLTAB	Specifies the identification number of a **TABLEDi** Bulk Data entry defining a tabular lower limit. [5]	Integer>0	Blank
ULIM	Specifies the upper limit of the constraint value. [4]	Real>0.0	Required
ULTAB	Specifies the identification number of a **TABLEDi** Bulk Data entry defining a tabular upper limit. [5]	Integer>0	Blank
LRANGE	Specifies the lower frequency limit (Hz) for which the constraint will be evaluated. [4]	Real	Required
URANGE	Specifies the upper frequency limit (Hz) for which the constraint will be evaluated. [4]	Real	Required
GIDi	GRID point identification number.	Integer>0	Required
COMPi	Single component at which the response will be constrained. [4,6]	Character	Required
VALi	Combinatorial factor to assemble the constrained response. [4]	Real	Required

Remarks:

1. Design constraint sets must be selected in the Case Control packet with the command:

   DESCON = SID

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. The responses **DISP**, **VELO**, and **ACCE** may only be specified for structural points, and the response **PRES**sure, **DB**, and **DBA** may only apply to fluid points. In the case of fluids points, the responses may be on a free surface if the fluid is incompressible.

4. The **DCDYNRG** entry constrains a response amplitude, $R(\omega)$, over a range of frequencies. The response is built from the displacement, velocity, or acceleration responses at one or more degrees of freedom using:

$$R(\omega) = \mathbf{VALi}\left[\sum_{i=1}^{ng} mag \begin{Bmatrix} u_i \\ i\,\omega\,u_i \\ -\omega^2 u_i \end{Bmatrix}\right]$$

   When **AVG** is selected, the constraints formed are:

$$\text{for upper bounds:}\quad g = \frac{1}{nf \cdot \mathbf{ULIM}}\left[\sum_{j=1}^{nf} R(\omega_j)\right] - 1.0 \le 0.0$$

$$\text{for lower bounds:}\quad g = 1.0 - \frac{1}{nf \cdot \mathbf{LLIM}}\left[\sum_{j=1}^{nf} R(\omega_j)\right] \le 0.0$$

   and, if used as the objective, the form is:

$$F = \frac{1}{nf}\left[\sum_{j=1}^{nf} R(\omega_j)\right]$$

   and when **PEAK** is selected, the *nf* constraints are formed using:

$$\text{for upper bounds:}\quad g_j = \frac{R(\omega_j)}{\mathbf{ULIM}} - 1.0 \le 0.0 \qquad j = 1,\ldots, nf$$

$$\text{for lower bounds:}\quad g_j = 1.0 - \frac{R(\omega_j)}{\mathbf{LLIM}} \le 0.0 \qquad j = 1,\ldots, nf$$

   In each case, *nf* is the number of frequencies specified by **FREQi** Bulk Data entries in the range:

   LRANGE $\le f \le$ **URANGE**

   and, if used as the objective, the form is:

$$F = R(\omega_j)$$

7 BULK

5. The table must specify the lower or upper limit as a function of frequency. The x-axis is the frequency (Hz) and the y-axis is then the response limit.
6. You may select a displacement component from the three translations, **T1**, **T2**, or **T3**, or from the three rotations, **R1**, **R2**, or **R3**.
7. **DCDYNRG** data are used only when performing Frequency Response Analyses in Multidisciplinary Design Optimization.

Bulk Data Entry **DCELEM**

Defines an element response constraint, R_e, on stress, R_σ, strain, R_ε or force, R_p.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DCELEM	SID	DCNAME	RTYPE	COMP	LLIM	ULIM		EID1	-cont-
-cont-	*ELEMENT ID LIST*								-cont-

DCELEM	100	SIG101	STRESS	SIGX		7.3+4		101	+A
+A	102	105	THRU	120					

Field	*Contents*	*Data Range*	*Default*
SID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labelling. [2]	Character	Blank
RTYPE	Indicates the response type being constrained.	Keyword { **STRESS** **STRAIN** **FORCE** }	Required
COMP	Response component subject to the constraint. [3,4]	Character	Required
LLIM	The value of a lower limit response. [4,5]	Real	[5]
ULIM	The value of an upper limit response. [4,5]	Real	[5]
ELEMENT ID LIST	List of one or more element identication numbers to which the constraint applies. [5,6]	Integer>0	EID1 is Required

Remarks:

1. Design constraint sets must be selected in the Case Control packet with the command:

   DESCON = SID

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. You may select any element response quantity shown in the table on the following page.

4. This Bulk Data entry specifies bounds on specific element responses. The form of the actual design constraints, and the objective function if a **DCELEM** value is used for the objective, are shown below:

CONSTRAINT SPECIFICATION	FORM OF DESIGN CONSTRAINT	FORM AS OBJECTIVE FUNCTION
$R_e = \left\{ \begin{matrix} R_\sigma \\ R_\varepsilon \\ R_p \end{matrix} \right\} \geq R_{llim}$	$g = \dfrac{R_{llim} - R_e}{\lvert R_{llim} \rvert} \leq 0.0$	$F = R_e$
$R_e = \left\{ \begin{matrix} R_\sigma \\ R_\varepsilon \\ R_p \end{matrix} \right\} \leq R_{ulim}$	$g = \dfrac{R_e - R_{ulim}}{\lvert R_{ulim} \rvert} \leq 0.0$	

7 BULK

RESPONSE TYPE	RESPONSE SYMBOL	ELEMENT CLASS												
		SCLR	ONE-D				TWO-D					THREE-D		
		ELASi	BAR	BEAM	PILE	ROD	QUAD4	TRIA3	QUAD8	TRIA6	SHEAR	TETRA	PENTA	HEXA
ELEMENT STRESS	SIGX	■	■	■	■	■	■	■	■	■		■	■	■
	SIGY						■	■	■	■		■	■	■
	SIGZ											■	■	■
	TAUXY						■	■	■	■		■	■	■
	TAUYZ											■	■	■
	TAUZX											■	■	■
	SIG1		■	■	■	■	■	■	■	■		■	■	■
	SIG2						■	■	■	■		■	■	■
	SIG3											■	■	■
	MEANSTRS											■	■	■
	MAXSHEAR					■	■	■	■	■	■	■	■	■
	VONMISES		■	■	■	■	■	■	■	■		■	■	■
ELEMENT STRAIN	EPSX						■	■	■	■				
	EPSY						■	■	■	■				
	EPSXY						■	■	■	■				
	EPS1						■	■	■	■				
	EPS2						■	■	■	■				
	MAXSHEAR						■	■	■	■	■			
	VONMISES						■	■	■	■				
ELEMENT FORCE	AXIAL	■	■	■	■	■	■	■	■	■				
	SHEAR		■	■	■		■	■	■	■	■			
	BENDING		■	■	■		■	■	■	■				
	TWIST		■	■	■	■	■	■	■	■				

5. One of the limits **LLIM** or **ULIM** is required.
6. The **EIDi** list may include **THRU** or **THRU/BY** groups.
7. **DCELEM** data are used only when performing Sensitivity analyses or Multidisciplinary Design Optimization.

Bulk Data Entry **DCFREQ**

Defines a design constraint on a cyclic frequency, *f*.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DCFREQ	DCSID	DCNAME	UPLOW	LIM	MODEID				
DCFREQ	100	F1	UPPER	9.5	1				

Field	*Contents*	*Data Range*	*Default*
DCSID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labelling. [2]	Character	Blank
UPLOW	Selects either an **UPPER** or **LOWER** bound for the specified **MODE**.	Character { **UPPER** } { **LOWER** }	None
LIM	The value of selected frequency limit (Hz). [3]	Real	**0.0**
MODEID	Mode number to which the constraint applies. [4]	Integer>0	Required

Remarks:

1. Design constraint sets must be selected in the Case Control packet with the command:

   DESCON = SID

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. This Bulk Data entry specifies bounds on a specific specific frequency. The form of the actual design constraints, and the objective function if a **DCFREQ** value is used for the objective, are shown below:

CONSTRAINT SPECIFICATION	FORM OF DESIGN CONSTRAINT	FORM AS OBJECTIVE FUNCTION
$f \le$ **LIM** if **UPLOW** = **UPPER**	$g = \frac{\lambda}{4\pi^2 \mathbf{LIM}^2} - 1 \le 0.0$	$F = \lambda$
$f \ge$ **LIM** if **UPLOW** = **LOWER**	$g = 1 - \frac{\lambda}{4\pi^2 \mathbf{LIM}^2} \le 0.0$	

   Note that in both cases the ***eigenvalue*** is used in constraint calculations even though the ***frequency*** is specified on the Bulk Data entry.

4. The **MODEID** refers to the extraction number of a mode during the baseline analysis. In subsequent iterations of MDO, the modes are automatically tracked and the constraint is placed on the correct mode shape regardless of its extraction order. To deselect mode tracking you may use **PARAM,NOTRACK**,-1. This is not recommended since your constraint will be applied to the wrong mode in the event of switching.

5. **DCFREQ** data are used only when performing Sensitivity analyses or Multidisciplinary Design Optimization.

Bulk Data Entry DCGRID

Defines a GRID point response constraint, R_g.

Formats and Example:

1	2	3	4	5	6	7	8	9	10
DCGRID	DCSID	DCNAME	"DISP"	COMP	LLIM	ULIM		GID1	-cont-
-cont-	*GRID ID LIST*								-cont-

DCGRID	100	TIPD	DISP	T3		-0.1		101	+A
+A	102	210	THRU	220					

Field	*Contents*	*Data Range*	*Default*
DCSID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labelling. [2]	Character	Blank
"DISP"	Indicates that you are defining a displacement constraint.	Keyword	Required
COMP	Response component subject to the constraint. [3,4]	Character	Required
LLIM	The value of a lower limit response. [4,5]	Real	[5]
ULIM	The value of an upper limit response. [4,5]	Real	[5]
GRID ID LIST	List of one or more GRID point identification numbers to which the constraint applies. [6]	Integer>0	GID1 is Required

Remarks:

1. Design constraint sets must be selected in the Case Control packet with the command:

   DESCON = SID

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. You may select a displacement component from the three translations, **T1**, **T2**, or **T3**, or from the three rotations, **R1**, **R2**, or **R3**.

4. This Bulk Data entry specifies bounds on specific element responses. The form of the actual design constraints, and the objective function if a **DCGRID** value is used for the objective, are shown below:

CONSTRAINT SPECIFICATION	FORM OF DESIGN CONSTRAINT	FORM AS OBJECTIVE FUNCTION
$R_g \geq R_{llim}$	$g = \frac{R_{llim} - R_g}{\lvert R_{llim} \rvert} \leq 0$	$F = R_g$
$R_g \leq R_{ulim}$	$g = \frac{R_g - R_{ulim}}{\lvert R_{ulim} \rvert} \leq 0$	

7 BULK

5. One of the limits **LLIM** or **ULIM** is required. If the other is left blank, no limit is applied.
6. The **GIDi** list may include **THRU** and **THRU/BY**groups.
7. **DCGRID** data are used only when performing Sensitivity analyses or Multidisciplinary Design Optimization.

Bulk Data Entry DCGRIDM

Defines a response constraint, $R_{g.}$, for a linear combination of GRID point response components.

Formats and Example:

1	2	3	4	5	6	7	8	9	10
DCGRIDM	DCSID	DCNAME	"DISP"		LLIM	ULIM			-cont-
-cont-	GID1	COMP1	A1	GID2	COMP2	A2			-cont-
-cont-	GID3	COMP3	A3	*CONTINUES IN GROUPS OF 3*					-cont-

DCGRIDM	100	RELD	DISP			-0.1			+A
+A	102	T1	1.0	202	T1	-1.0			

Field	*Contents*	*Data Range*	*Default*
DCSID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labelling. [2]	Character	Blank
"DISP"	Indicates that you are defining a displacement constraint.	Keyword	Required
LLIM	The value of a lower limit response. [3,4]	Real	[3]
ULIM	The value of an upper limit response. [3,4]	Real	[3]
GIDi	GRID point identification numbers to which the constraint applies.	Integer>0	GID1 is Required
COMPi	Single constrained response component. [5]	Character	Required
Ai	Coefficient of constrained response component. [4]	Real	Required

Remarks:

1. Design constraint sets must be selected in the Case Control packet with the command:

   DESCON = SID

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. One of the limits **LLIM** or **ULIM** is required. If the other left blank, no limit is applied.

7 BULK

4. This Bulk Data entry specifies bounds on specific element responses. The form of the actual design constraints, and the objective function if a **DCGRIDM** value is used for the objective, are shown below:

CONSTRAINT SPECIFICATION	FORM OF DESIGN CONSTRAINT	FORM AS OBJECTIVE FUNCTION
$R_g = \sum_i \mathbf{Ai}\, u_i \geq R_{llim}$	$g = \frac{R_{llim} - R_g}{\lvert R_{llim} \rvert}$	$F = R_g$
$R_g = \sum_i \mathbf{Ai}\, u_i \leq R_{ulim}$	$g = \frac{R_g - R_{ulim}}{\lvert R_{ulim} \rvert}$	

5. You may select a displacement component from the three translations, **T1**, **T2**, or **T3**, or from the three rotations, **R1**, **R2**, or **R3**.

6. **DCGRIDM** data are used only when performing Sensitivity analyses or Multidisciplinary Design Optimization.

Bulk Data Entry DCMODE

Requests that eigenvector sensitivity be computed for selected modes.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DCMODE	DCSID	DCNAME	*MODEID LIST*						-cont-
DCMODE	100	SHAPES	1	THRU	10				

Field	*Contents*	*Data Range*	*Default*
DCSID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labelling. [2]	Character	Blank
MODEID LIST	List of one or more mode identification numbers.	Integer>0	Required

Remarks:

1. Design constraint sets must be selected in the Case Control packet with the command:

   DESCON = SID

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. **DCMODE** data are used only when performing Sensitivity analyses.

Bulk Data Entry DCMODEL

Defines a discipline- and CASE-independent model constraint in Multidisciplinary Design Optimization.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DCMODEL	DCSID	DCNAME	TYPE		LLIM	ULIM			
DCMODEL	100	WEIGHT	WEIGHT		1000.0	1100.0			

Field	*Contents*	*Data Range*	*Default*
DCSID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labelling. [2]	Character	Blank
TYPE	Specifies the model characteristic to be constrained. [3]	Character { **WEIGHT** **MASS** **VOLUME** }	Required
LLIM	Specifies the lower limit of the model constraint value. [4]	Real	Required
ULIM	Specifies the upper limit of the model constraint value. [4]	Real	Required

Remarks:

1. An overall model design constraint must be selected in the Case Control packet with the command:

   MODDESCON = SID

   This command must appear before any analysis **CASE**s.

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. **DCMODEL** data are used only when performing Multidisciplinary Design Optimization.

7 BULK

Bulk Data Entry DCMODR

Defines an eigenvector constraint by specifying the required shape as coefficients at some number of GRID points and components in the model. Either the RMS error or each specified modal displacement may be constrained.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DCMODR	DCSID	DCNAME	MODEID	FORM	LIM	NORM	GIDN	COMPN	-cont-
-cont-		GID1	COMP1	DISP1	GID2	COMP2	DISP2		-cont-
-cont-		GID3	COMP3	DISP3	*CONTINUES IN GROUPS OF 3*				-cont-

DCMODR	100	MSHAPE	1	RMS	0.01				

Field	*Contents*	*Data Range*	*Default*
DCSID	Design constraint set identification number. [1]	Integer>0	Required
DCNAME	Constraint name for output labelling. [2]	Character	Blank
MODEID	Mode identification number. [3]	Integer>0	Required
FORM	Form of the constraint. May be **RMS** to constrain the **RMS** error between the computed and input mode shape, or **COMP** to request that each component in the input mode shape be constrained. [4]	Character { **RMS** **COMP** }	Required
LIMIT	Specifies the maximum error between the computed and input mode shape. [4]	Real	Required
NORM	Normalization in which the required eigenvector terms (**VALi**) are input. [5]	Character { **MASS** **MAX** **POINT** }	MAX
GIDN	GRID or SCALAR point identification number used if **NORM** is **POINT**.	Integer>0	[5]
COMPN	Single degree of freedom of **GIDN** if **NORM** is **POINT**. [6]	Character	[5]
GIDi	GRID point identification number.	Integer>0	Required
COMPi	Single degree of freedom at which the desired mode shape is measured. [4,6]	Character	Required
DISPi	The modal deformation at (**GIDi,COMPi**) for the desired mode shape. [4]	Real	Required

Remarks:

1. Design constraint sets must be selected in the Case Control packet with the command:

   DESCON = SID

7 BULK

2. Although the constraint name is not required, it is strongly recommended to improve the readability of your solution results.

3. The **MODEID** refers to the extraction number of a mode during the baseline analysis. In subsequent iterations of MDO, the modes are automatically tracked and the constraint is placed on the correct mode shape regardless of its extraction order. To deselect mode tracking you may use **PARAM,NOTRACK,-1**. This is not recommended since your constraint will be applied to the wrong mode in the event of switching.

4. The mode shape constraint input allows you to specify a desired mode shape for the model as normalized displacements at some number of degrees of freedom. Either the average (**RMS**) error can be constrained, or the error at each degree of freedom (**COMP**) may be constrained.

   When **RMS** is selected, the constraint formed is:

$$g = \frac{1}{nc}\left[\sum_{j=1}^{nc}\left[\left(u_i - \mathbf{VALi}\right)^2\right]\right]^{1/2} - \mathbf{LIM} \le 0.0$$

   and, if used as the objective, the form is:

$$F = \frac{1}{nc}\left[\sum_{j=1}^{nc}\left[\left(u_i - \mathbf{VALi}\right)^2\right]\right]^{1/2}$$

   and when **COMP** is selected, the *nc* constraints are formed using:

$$g_i = \left[\left(u_i - \mathbf{VALi}\right)^2\right]^{1/2} - \mathbf{LIM} \le 0.0 \quad i = 1, \ldots, nc$$

   and the objective form is:

$$F = \left[\left(u_i - \mathbf{VALi}\right)^2\right]^{1/2}$$

   In each case, *nc* is the number of components, **COMP**, specified for the mode shape.

5. The **NORM** is independent of the eigenextraction method used for the modal analysis. It is simply the manner in which the ***input*** modal deformations were normalized. See the **EIGR** Bulk Data entry for a detailed description of these analysis normalization options.

6. You may select a displacement component from the three translations, **T1**, **T2**, or **T3**, or from the three rotations, **R1**, **R2**, or **R3**.

7. **DCMODR** data are used only when performing Sensitivity analyses or Multidisciplinary Design Optimization.

7 BULK

Bulk Data Entry DEFORM

Defines enforced axial deformation for one-dimensional elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DEFORM	SID	EID1	D1	EID2	D2	EID3	D3		

DEFORM	102	865	0.05	866	-.08				

Field	*Contents*	*Data Range*	*Default*
SID	Deformation set identification number. [1]	Integer>0	Required
EIDi	Element identification number.	Integer>0	Required
Di	Deformation. [2]	Real	Required

Remarks:

1. Deformation sets must be selected in the Case Control packet with the command:

   DEFORM = SID

2. A positive value of **Di** means extension of the element.
3. The referenced element must be one-dimensional and selected from ROD, CONROD, TUBE, BAR, or BEAM.
4. Axial deformations may only be used in statics, differential stiffness, and buckling.

7 BULK

Bulk Data Entry DEFUSET

Defines names for user sets.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DEFUSET	UNAME	NNAME	*CONTINUES IN GROUPS OF 2*						
DEFUSET	U1	MYU							

Field	*Contents*	*Data Range*	*Default*
UNAME	User set name (**U1-U8**). [1]	Character [1]	Required
NNAME	New name to be used for **UNAME**. [2]	Character[1]	Required

Remarks:

1. Both **UNAME** and **NNAME** must be four or less characters in length.
2. The **NNAME**s must be unique.
3. The user set names may be referenced on **USET** and **USET1** Bulk Data entries and are used in some DMAP modules such as **VEC**, **UPARTN**, and **UMERGE**. Renaming of the user sets is not required, you may use the names **U1-U8**.

7 BULK

Bulk Data Entry DELAY

Defines the time delay for frequency or time dependent dynamic loads.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DELAY	SID	GID1	DOF1	T1	GID2	DOF2	T2		
DELAY	101	122	6	6.05	126	5	2.7		

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of **DELAY** set. [1]	Integer>0	Required
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom.	DOF Code	Required
Ti	Time delay for designated coordinate.	Real	**0.0**

Remarks:

1. The **DELAY** data are selected by referencing the set identification number, **SID**, on either a **RLOAD1** or **RLOAD2** Bulk Data entry (frequency response) or **TLOAD1** or **TLOAD2** entry (transient response).

7 BULK

Bulk Data Entry DELAYS

Defines the time delay for frequency or time dependent loads in reference to a Basic Substructure.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DELAYS	SID	SNAME	GID1	DOF1	T1	GID2	DOF2	T2	

DELAYS	4	SKIN	21	6	2.1	8	6	7.2	

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of **DELAY** set. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom.	DOF Code	Required
Ti	Time delay for designated component.	Real	**0.0**

Remarks:

1. The **DELAYS** data are selected by referencing the set identification number, **SID**, on either a **RLOAD1** or **RLOAD2** Bulk Data entry (frequency response) or **TLOAD1** or **TLOAD2** entry (transient response).
2. **DELAYS** data are used in the substructuring SOLVE operation.
3. Points referenced must exist in the substructure to be SOLVEd.

7 BULK

Bulk Data Entry DLOAD

Requests dynamic loads and enforced motions for frequency response and transient response problems as a linear combination of load sets of the form:

$$P = \mathbf{S} \sum_{i} \mathbf{S_i} P_{\mathbf{Li}}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
DLOAD	LID	S	S1	L1	S2	L2	S3	L3	-cont-
-cont-	S4	L4	S5	L5	*CONTINUES IN GROUPS OF 2*				-cont-

DLOAD	106	1.0	1.5	108	26.0	121	12.5	137	+A
+A	6.7	194							

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
S	Overall scale factor for load set **LID**.	Real	**0.0**
Si	Scale factor for load set **Li**.	Real	**0.0**
Li	Load set identification numbers. [2,3,4]	Integer>0	[5]

Remarks:

1. Dynamic load sets must be selected in the Case Control packet with the command:

   DLOAD = LID

2. The **LID** and **Li** must be unique.
3. Nonlinear transient loads are not selected with this command; they are selected separately in the Case Control packet using the **NONLINEAR** command.
4. A **DLOAD** entry may not reference a load set identification number defined by another **DLOAD** or **DLOAD1** entry.
5. The **Li** may reference only **TLOAD1** and **TLOAD2** data for transient response analyses; **RLOAD1** and **RLOAD2** data for frequency and random response analyses; or **ACSRCE** data for frequency response in fluid-structure interaction.
6. **TLOAD1** and **TLOAD2** loads may be combined only using **DLOAD** or **DLOAD1** entries.
7. **RLOAD1**, **RLOAD2**, and **ACSRCE** loads may be combined only using **DLOAD** or **DLOAD1** entries.
8. **RLOAD1** and **RLOAD2** loads may be combined only with the **DLOAD** entry.

7 BULK

Bulk Data Entry DLOAD1

Requests dynamic loads and enforced motions for frequency response and transient response problems as a linear combination of load sets of the form:

$$P = \mathbf{S} \sum_{i} P_{\mathbf{Li}}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
DLOAD1	LID	S	*LOAD ID LIST*						-cont-

DLOAD1	200	1.0	102	108	109	121	125	137	+A
+A	191	194	200	THRU	220				

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
S	Overall scale factor for load set **LID**.	Real	**0.0**
LOAD ID LIST	Load set identification numbers. [2,3,4]	Integer>0	[5]

Remarks:

1. Dynamic load sets must be selected in the Case Control packet with the command:

   DLOAD = LID

2. The **LID** and ***LOAD ID LIST*** identification numbers must be unique.
3. Nonlinear transient loads are not selected with this command; they are selected separately in the Case Control packet using the **NONLINEAR** command.
4. A **DLOAD1** entry may not reference a load set identification number defined by another **DLOAD1** or **DLOAD** entry.
5. The ***LOAD ID LIST*** may reference only **TLOAD1** and **TLOAD2** data for transient response analyses; **RLOAD1** and **RLOAD2** data for frequency and random response analyses; or **ACSRCE** data for frequency response in fluid-structure interaction.
6. **TLOAD1** and **TLOAD2** loads may be combined only using **DLOAD** or **DLOAD1** entries.
7. **RLOAD1**, **RLOAD2**, and **ACSRCE** loads may be combined only using **DLOAD** or **DLOAD1** entries.

7 BULK

Bulk Data Entry **DMI**

Defines a matrix data block which may then be directly referenced in DMAP.

Format and Example:

Each matrix definition is composed of a matrix header entry and one or more column entries.

Matrix Header Entry:

1	2	3	4	5	6	7	8	9	10
DMI	NAME	"0"	FORM	TYPIN	TYPOUT		NROW	NCOL	

Column Entries:

DMI	NAME	COL	ROW1	TERM	TERM	*CONTINUES*	ROW2	TERM	-cont-
-cont-	TERM	*CONTINUES*	ROWm	TERM	"THRU"	ROWn	*CONTINUES*		-cont-

DMI	MYMAT	0	2	1	1		10	4	
DMI	MYMAT	1	1	1.5	2.5	...	8	12.3	+A
+A	10.3	...	5	2.7	THRU	8			

Field	*Contents*	*Data Range*	*Default*
NAME	Name of matrix. [1]	Character	Required
FORM	Form of matrix. [2]	Integer>0	Required
TYPIN	Type of matrix being entered. [2]	Integer>0	Required
TYPOUT	Type of matrix to be created. [2]	Integer>0	Required
NROW	Number of rows in the matrix.	Integer>0	Required
NCOL	Number of columns in the matrix.	Integer>0	Required
COL	Column number of data being entered.	Integer>0	Required
ROWi	Row number of data being entered.	Integer>0	Required
TERM	Matrix terms. [3,4]	Real	**0.0**

Remarks:

1. To use **DMI** data, either Rigid Format ALTERs or a DMAP sequence must be used. DMI matrices are immediately available for use at any location in the Rigid Format. For example, to print your DMI matrix the following DMAP ALTER form, in Executive Control, may be used:

```
ALTER statement_number
MATPRN dmi_name,,,, // $
ENDALTER
```

   where *dmi_name* is the matrix name you assigned on the **DMI** Bulk Data entries, and *statement_number* is the location in the Rigid Format where you wish the matrix print to occur.

7 BULK

2. Each DMI matrix requires a header entry and a column entry for each nonnull column. The header entry specifies the form and data type of the input and output matrix as specified in the following table:

FORM Form of Matrix		TYPIN Type of Input Data		TYPOUT Type of Output Matrix	
1	Square, unsymmetric	**1**	Real, single precision	**1**	Real, single precision
2	Rectangular	**2**	Real, double precision	**2**	Real, double precision
6	Symmetric	**3**	Complex, single precision	**3**	Complex, single precision
		4	Complex, double precision	**4**	Complex, double precision

3. It is necessary to enter only the nonzero matrix terms. Symmetric matrices require that you enter all data. You may enter only the upper or lower and diagonal terms and use the following DMAP ALTER packet to create a complete, symmetric matrix:

```
ALTER statement_number
$
$ Extract the diagonal terms
$
DIAGONAL dmi_name / DIAGTERM / C,Y,OPT=SQUARE $
$
$ Transpose the original input matrix
$
TRNSP dmi_name / TRANPMAT $
$
$ Add the original matrix, the transposed matrix, and
$ subtract the extra diagonal terms
$
ADD5 dmi_name,TRANPMAT,DIAGTERM,, / FINALMAT $
SWITCH FINALMAT,dmi_name // -1 $
ENDALTER
```

   Matrix *dmi_name* is now the desired symmetric matrix.

4. The **THRU** option allows a long sequence of identical rows to be entered quickly.

5. Complex matrix terms are entered in two consective fields defining the real and imaginary parts.

6. A blank field is not equivalent to a zero. If a zero is desired, it must be entered explicitly.

7. There is no required relationship between input data types, i.e. single precision versus double precision, and the standard precision versus high precision Bulk Data entry format. Generally, a high precision Bulk Data format is required for meaningful double precision input data.

Bulk Data Entry DMIAX

Defines an axisymmetric related direct input matrix which is selected for use with a Case Control command.

Format and Example:

Each matrix definition is composed of a matrix header entry and one or more entries defining the matrix.

Matrix Header Entry:

1	2	3	4	5	6	7	8	9	10
DMIAX	NAME	"0"	FORM	TYPIN	TYPOUT				

Column Entries:

DMIAX	NAME	GIDJ	DOFJ	HIDJ					-cont-
-cont-	GIDI1	DOFI1	HIDI1	XIJ1	YIJ1				-cont-
-cont-	GIDI2	DOFI2	HIDI2	XIJ2	YIJ2				-cont-
-cont-	*CONTINUES WITH ONE TERM PER RECORD*								-cont-

DMIAX	K2PP	0	1	3	4				
DMIAX	K2PP	103							+A
+A	542	4		4.6+6	3.4+4				+B
+B	543	4		1.1+7	8.5+3				

Field	*Contents*	*Data Range*	*Default*
NAME	Name of matrix. [2]	Character	Required
FORM	Form of matrix. [3,4]	Integer	Required
TYPIN	Type of matrix data being entered. [4,5,6]	Integer	Required
TYPOUT	Type of matrix to be created. [4,6]	Integer	Required
GIDJ, GIDIi	GRID, SCALAR, RINGFL, POINTAX, PRESPT pressure point, FREEPT free surface displacement, or EXTRA point identification number. [7]	Integer>0	Required
DOFJ, DOFIi	Single degree of freedom for **GIDJ** and **GIDIi** if they are GRID points. [7]	DOF Code	Required
HIDJ, HIDIi	Harmonic identification number of RINGFL point.	Integer	[7]
XIJi,YIJi	Real and imaginary parts of the matrix term. [8]	Real	**0.0**

7 BULK

Remarks:

1. This entry is allowed only if an **AXIF** entry is also present in the Bulk Data packet.
2. Matrices defined by this entry may be used in dynamics by using the Case Control commands by **K2PP=NAME**, **B2PP=NAME**, or **M2PP=NAME**.
3. For symmetric matrices, the entire matrix must be input.
4. Each DMIAX matrix requires a header entry and a column entry for each nonnull column. The header entry specifies the form and data type of the input and output matrix as specified in the following table:

FORM Form of Matrix		TYPIN Type of Input Data		TYPOUT Type of Output Matrix	
1	Square, unsymmetric	**1**	Real, single precision	**1**	Real, single precision
2	Rectangular	**2**	Real, double precision	**2**	Real, double precision
6	Symmetric	**3**	Complex, single precision	**3**	Complex, single precision
		4	Complex, double precision	**4**	Complex, double precision

5. If **TYPIN = 1** or **2**, **YIJi** must be blank.
6. There is no required relationship between input data types, i.e. single precision versus double precision, and the standard precision versus high precision Bulk Data entry format. Generally, a high precision Bulk Data format is required for meaningful double precision input data.
7. Enter a positive harmonic number for COSINE series RINGFL terms, the negative of the harmonic number for SINE series RINGFL terms, and leave blank for other degrees of freedom from GRID, SCALAR or POINTAX points.
8. Only nonzero terms need be entered.

Bulk Data Entry DMIG

Defines a direct input matrix in terms of GRID point degrees-of-freedom.

Format and Example:

Each matrix definition is composed of a matrix header entry and one or more entries defining each non-null column of the matrix.

Matrix Header Entry:

1	2	3	4	5	6	7	8	9	10
DMIG	NAME	"0"	FORM	TYPIN	TYPOUT	CFORM	SYMTOL	NCOL	

Column Entries:

DMIG	NAME	GIDJ	DOFJ		GIDI1	DOFI1	XIJ1	YIJ1	-cont-
-cont-	GIDI2	DOFI2	XIJ2	YIJ2	GIDI3	DOFI3	XIJ3	YIJ3	-cont-
-cont-	GIDI4	DOFI4	XIJ4	YIJ4	*CONTINUES IN GROUPS OF 4*				-cont-

Field	*Contents*	*Data Range*	*Default*
NAME	Name of matrix. [1,2]	Character	Required
FORM	Form of Matrix. [3,5,6]	Integer	Required
TYPIN	Type of matrix data being entered. [7]	Integer	Required
TYPOUT	Type of matrix to be created.	Integer	Required
CFORM	Specifies the format of data entry for complex matrices. [8]	Character	REAL
SYMTOL	Symmetric matrix tolerance check. [5,9]	Real≥0.0	**0.0**
NCOL	Number of columns in the matrix, only used if **FORM=9**. [6]	Integer>0	[10]
GIDJ, GIDIi	GRID, SCALAR or EXTRA point identification numbers. [10]	Integer>0	Required
DOFJ, DOFIi	Single degree-of-freedom. [10]	DOF Code	Required
XIJi,YIJi	Real and Imaginary parts of matrix element at **GIDIi** and **DOFIi**. [8,10]	Real	**0.0**

Remarks:

1. Matrices defined by this entry may be used in any Rigid Format by using the Case Control commands:

   K2GG = NAME or B2GG = NAME or M2GG = NAME

   Additionally, in dynamics Rigid Formats, you may also use:

   K2PP = NAME or B2PP = NAME or M2PP = NAME

7 BULK

2. To input more general matrices, a DMAP ALTER packet is required as described next. For these cases there are no Case Control commands to be used. Matrices of the following dimensions may be input:

   - g-set rows by *g-set* columns
   - g-set rows by any arbitrary number of columns
   - p-set rows by *p-set* columns
   - p-set rows by any arbitrary number of columns

   For all matrices with *g-set* number of rows, place the following DMAP statement in an appropriate location in the Rigid Format, anywhere after module GP1:

   ```
   MTRXIN,   ,MATPOOL,EQEXIN,,/ NAME1,NAME2,NAME3 / LUSET / S,N,NOMAT1
   /
             S,N,NOMAT2 / S,N,NOMAT3 $
   ```

   where **NAMEi** are names used on the **DMIG** Bulk Data entries. Any of the **NAMEi** may be blank so between one and three matrices may be input with each **MTRXIN** DMAP statement. **MTRXIN** saves parameters **NOMATi** which indicate the status of matrices **NAMEi**. **NOMATi = 1** if the matrix is created, otherwise **NOMATi = -1**.

   Note, the comma immediately following **MTRXIN** as shown above is required.

   For all matrices with *p-set* number of rows, place the following DMAP statement in an appropriate location in a dynamics Rigid Format, anywhere after module DPD:

   ```
   MTRXIN,   ,MATPOOL,EQDYN,,/ NAME1,NAME2,NAME3 / LUSETD / S,N,NOMAT1
   /
             S,N,NOMAT2 / S,N,NOMAT3 $
   ```

   The other information presented above for the *g-set* case is also true for the *p-set* case.

3. Each **DMIG** matrix requires a header entry and a column entry for each nonnull column. The header entry specifies the form and data type of the output matrix and specifies the type of input data as described in the following table:

FORM Form of Matrix		TYPIN Type of Input Data		TYPOUT Type of Output Matrix	
1	Square, unsymmetric	**1**	Real, single precision	**1**	Real, single precision
2	Rectangular [6]	**2**	Real, double precision	**2**	Real, double precision
6	Symmetric [5]	**3**	Complex, single precision	**3**	Complex, single precision
7	Symmetric [5]	**4**	Complex, double precision	**4**	Complex, double precision
9	Rectangular [6]				

   When **FORM=7**, the off-diagonal terms of the symmetric matrix are averaged to guarantee symmetry.

4. There is no required relationship between input data types, i.e. single precision versus double precision, and the standard precision versus high precision Bulk Data entry format. Generally, a high precision Bulk Data format is required for meaningful double precision input data.

5. There are two methods of entering symmetric matrices. When using **FORM=6**, only the upper ***or*** lower triangle of the matrix is entered. If you define a term in both halves of the matrix, a fatal error will occur. If you use **FORM=7**, then you enter the full matrix. In the latter case, the **SYMTOL** field is used to specify a tolerance check for comparing terms in the upper and lower halves of the

matrix to insure that the matrix is, indeed, symmetric. If the values are not identical, then the average of the two values is used for each value in the final matrix.

6. There are three methods of entering rectangular matrices. The methods differ both in how the actual column numbers are determined, and how the the total number of columns is determined. For all methods, the matrices will have either *g-size* or *p-size* rows depending on the context of the **MTRXIN** call.

   When **FORM=9**, and when a value is entered for **NCOL**, the **GIDJ** value is directly used as the column number. The **DOFJ** field should be left blank in this case. Any columns that you do not enter are assumed to be null.

   When **FORM=9**, and the **NCOL** field is left blank, the **GIDJ/DOFJ** fields are only used to determine the order of the columns entered. The actual column numbers will be sequential from the first to the last column. Null columns can thus not be entered in this manner.

   When **FORM=2**, **NCOL** is not used. The **GIDJ/DOFJ** values are converted to the correct internal sequence numbers. These are then used as the column number. The number of columns in the resulting matrix corresponds to the largest internal sequence number specified by a **GIDJ/DOFJ** pair. All other columns are null.

7. If **TYPIN=1** or **2**, **YIJi** must be blank.

8. If **POLAR=REAL**, then the **(XIJi,YIJi)** pairs are entered as real and imaginary components. If **POLAR=POLAR**, they are entered as magnitude and phase. In the latter case, the phase angle is always entered in degrees.

9. **SYMTOL** specifies the tolerance used in verifying the symmetry of the input matrix when **FORM=7**. For real matrices, the off-diagonal terms must satisfy the relation:

$$\frac{|\ \mathbf{XIJ} - \mathbf{XJI}\ |}{\max(|\ \mathbf{XIJ}\ |\ ,\ |\ \mathbf{XJI}\ |)} < \mathbf{SYMTOL}$$

   For complex matrices, the relation is:

$$\left|\ \sqrt{\mathbf{XIJ}^2 - \mathbf{YIJ}^2}\ - \sqrt{\mathbf{XJI}^2 - \mathbf{YJI}^2}\ \right| < \mathbf{SYMTOL}$$

   A matrix of **FORM=7** which does not satisfy one of these relationships will result in a fatal error.

10. Only the nonzero terms of the matrix need be entered.

Bulk Data Entry DPHASE

Defines the phase lead term for frequency-dependent loads.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DPHASE	SID	GID	DOF	θ					
DPHASE	105	122	6	4.1					

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of **DPHASE** set. [1]	Integer>0	Required
GID	GRID or SCALAR point identification number.	Integer>0	Required
DOF	List of degrees of freedom.	DOF Code	Required
θ	Phase lead (in degrees) for designated degrees of freedom.	Real	**0.0**

Remarks:

1. The **DPHASE** data are selected by referencing the set identification number, **SID**, on either a **RLOAD1** or **RLOAD2** Bulk Data entry for frequency response analysis.

7 BULK

Bulk Data Entry DPHASES

Defines the phase lead term for frequency-dependent loads in reference to a Basic Substructure.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DPHASES	SID	SNAME	GID	DOF	θ				
DPHASES	4	SKIN	21	6	2.1				

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of **DPHASE** set. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GID	GRID or SCALAR point identification number.	Integer>0	Required
DOF	List of degrees of freedom.	DOF Code	Required
θ	Phase lead (in degrees) for the designated degree of freedom.	Real	**0.0**

Remarks:

1. The **DPHASES** data are selected by referencing the set identification number, **SID**, on either a **RLOAD1** or **RLOAD2** Bulk Data entry for frequency response analysis.
2. **DPHASES** data are used in the substructuring SOLVE operation.
3. Points referenced must exist in the substructure to be SOLVEd.

7 BULK

Bulk Data Entry DSFACT

Defines scale factors for applied loads and stiffness matrices for a Differential Stiffness analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DSFACT	SID	B1	B2	B3	B4	B5	B6	B7	-cont-
-cont-	B8	*CONTINUES WITH LIST OF VALUES*							-cont-

DSFACT	198	-1.5	-2.5	-5.0	-7.0	-9.0	-10.	-15.	+A
+A	-20.	-30.	-40.	-50.					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
Bi	Scale factor. [2]	Real	**0.0**

Remarks:

1. The Differential Stiffness load set factors must be selected in the Case Control packet with the command:

   DSCO = SID

2. At least one scale factor, **B1**, is required.

7 BULK

Bulk Data Entry DTI

Defines a table data block which may be directly used in DMAP.

Format and Example:

Each table definition is composed of a table header entry and one or more entries defining the records in the table.

Table Header Entry:

1	2	3	4	5	6	7	8	9	10
DTI	NAME	0	T1	T2	T3	T4	T5	T6	-cont-
-cont-	H3	H4	H5	*CONTINUES WITH LIST OF VALUES*					-cont-

Record Entries:

DTI	NAME	RECNO	VAL1	VAL2	VAL3	VAL4	VAL5	VAL6	-cont-
-cont-	VAL7	*CONTINUES WITH LIST OF VALUES*				"ENDREC"			

DTI	MYTAB	0	4	32	4096				
DTI	MYTAB	1	23	456	0	0	1	4	+A
+A	16	2	33	ENDREC					

Field	*Contents*	*Data Range*	*Default*
NAME	Name of table.	Character	Required
Ti	Trailer values. [1]	Integer>0	**0**
Hi	Additional table header values.	Any	[2]
RECNO	Record number. [3]	Integer>0	Required
VALi	Table value.	Any	Required
"ENDREC"	Indicates the end of data for the current record. [4]	Character	Required

Remarks:

1. If **T1** is zero or blank, then it will be reset to the largest **RECNO** in the table.
2. If additional header data are not required, then continuation entries for the header record are not necessary, as shown in the example.
3. All records must be entered and numbered consecutively from 1 to the maximum entered.
4. If your data record ends with one or more blank fields, then you **must** use the optional **ENDREC** marker. The reserved word **ENDREC** may not be entered into a table. This is also true for the header values **Hi**.
5. Each table requires a header entry and an entry for each record in the table.
6. You must write a DMAP program or DMAP ALTER packet to use the **DTI** feature. All of the rules governing the use of data blocks in DMAP sequences apply.

Bulk Data Entry DVGRID

Defines a physical design variable as a GRID point coordinate and specifies its upper and lower bounds during Multidisciplinary Design Optimization.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DVGRID	DVNAME	GID		LBOUND	UBOUND	COORD			
DVGRID	TOPX	101		100.0	200.0	X			

Field	*Contents*	*Data Range*	*Default*
DVNAME	Physical design variable name. [1]	Character	Required
GID	GRID point identification number.	Integer>0	Required
LBOUND	Specifies the lower bound on the GRID point coordinate.	Real	10^{-6}
UBOUND	Specifies the upper bound on the GRID point coordinate.	Real	10^{9}
COORD	Selects the coordinate of the GRID point which will be the physical design variable. [2,3]	Character	Required

Remarks:

1. Design variable names are used in output presentation. They must be unique.
2. The design variable is the coordinate value in the ***input*** coordinate system, not the ***global*** coordinate system.
3. The value of **COORD** depends on the input coordinate system of GRID point **GID**. The following keywords are used:

Type of Input Coordinate System	Values for COORD		
Rectangular	**X**	**Y**	**Z**
Cylindrical	**R**	**THETA**	**Z**
Spherical	**R**	**THETA**	**PHI**

7 BULK

Bulk Data Entry DVGRIDS

Defines a simple linking of physical design variables which are GRID point coordinates and specifies the upper and lower bounds on coordinates to be used during Multidisciplinary Design Optimization.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DVGRIDS	DVNAME	GID1		LB1	UB1	COORD1			-CONT-
-CONT-		GID2		LB2	UB2	COORD2			-CONT-
-CONT-		*CONTINUES WITH GRID POINT ENTRIES*							-CONT-

DVGRIDS	SHAPE	101		100.0	200.0	X			+A
+A		201		50.0	75.0	X			+B
+B		301		50.0	100.0	X			

Field	*Contents*	*Data Range*	*Default*
DVNAME	Mathematical design variable name. [1]	Character	Required
GIDi	GRID point identification numbers.	Integer>0	Required
LBi	Specifies the lower bound on the selected **GIDi** coordinate.	Real	10^{-6}
UBi	Specifies the upper bound on the selected **GIDi** coordinate.	Real	10^{9}
COORDi	Selects the coordinate of **GIDi** which will be the physical design variable. [2,3,4]	Character	Required

Remarks:

1. Design variable names are used in output presentation. They must be unique.
2. The design variable is the coordinate value in the ***input*** coordinate syste, not the ***global*** coordinate system.
3. The value of **COORD** depends on the input coordinate system of GRID point **GID**. The following keywords are used:

Type of Input Coordinate System	Values for COORD		
Rectangular	**X**	**Y**	**Z**
Cylindrical	**R**	**THETA**	**Z**
Spherical	**R**	**THETA**	**PHI**

7 BULK

4. This Bulk Data entry is a simple method for defining a design variable linking scheme. The relationship between the **COORDi** values specified on this entry define the shape that will be maintained during the resizing process. The initial values of **COORDi** are used as a column in the linking matrix:

$$\begin{Bmatrix} \text{COORD1} \\ \text{COORD2} \\ \cdots \\ \text{COORDn} \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \\ \cdots \\ 0 \end{Bmatrix} + \begin{bmatrix} \text{COORD1}_o \\ \text{COORD2}_o \\ \cdots \\ \text{COORDn}_o \end{bmatrix} \text{v}$$

where the COORDi_o values are the initial values of the specified coordinates as defined by **GRID** Bulk Data entries. The value of the mathematical design variable, **v**, is initially set to **1.0**.

Bulk Data Entry DVLINK

Defines a mathematical design variable, v_m, as a linear relationship among physical design variables, v_p, of the form:

$$v_p = v_p^{inv} + T\, v_m$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
DVLINK	DVNAME	VINIT	PINV1	C1	PNAM1	PINV2	C2	PNAM2	-cont-
-cont-			PINV3	C3	PNAM3	*CONTINUES IN GROUPS OF 3*			-cont-

DVLINK	TLINK	1.0	0.0	1.0	T1	0.0	0.667	T2	
			0.0	0.333	T3				

Field	*Contents*	*Data Range*	*Default*
DVNAME	Mathematical design variable name. [1]	Character	Required
VINIT	Initial value of the mathematical design variable. [2]	Real	Required
PINVi	Invariant portion of the **i**th physical design variable **PNAMi**. [2,3]	Real	**0.0**
Ci	Coefficient linking the **i**th physical design variable with the mathematical design variable. [2,3]	Real	Required
PNAMi	Name of the **i**th physical design variable being linked to the mathematical variable. [2,3]	Character	Required

Remarks:

1. Design variable names are used in output presentation. They must be unique.
2. The following figure illustrates how **DVLINK** input data are used to define the relationship between physical and mathematical design variables:

$$\begin{Bmatrix} \mathbf{PNAM1} \\ \mathbf{PNAM2} \\ \mathbf{PNAM3} \end{Bmatrix} = \begin{Bmatrix} \mathbf{PINV1} \\ \mathbf{PINV2} \\ \mathbf{PINV3} \end{Bmatrix} + \begin{bmatrix} \cdot & \mathbf{C1} & \cdot & \cdot \\ \cdot & \mathbf{C2} & \cdot & \cdot \\ \cdot & \mathbf{C3} & \cdot & \cdot \end{bmatrix} \begin{Bmatrix} \cdot \\ \mathbf{VINIT} \\ \cdot \\ \cdot \end{Bmatrix}$$

   In the above, **PNAMi** represents the actual value computed for the linked physical design variable with name **PNAMi** (as defined with **DVPROP** or **DVGRID** Bulk Data), and **VINIT** is the initial value for the mathematical design variable named **DVNAME**.
3. All data **PNAMi, PINIVi** and **Ci** associated with the mathematical design variable **DVNAME** must be defined on a single **DVLINK** Bulk Data entry.
4. By default, the lower and upper bounds of the mathematical variable are set to $\mathbf{-10^{20}}$ and $\mathbf{10^{20}}$, respectively. The **DVMATH** Bulk Data entry may be used to override these defaults.
5. See the **UAI/*NASTRAN User's Guide*** for a detailed description of design variable linking.

7 BULK

Bulk Data Entry **DVMATH**

Defines the initial value and bounds for a mathematical design variable defined by **DVLINK** data.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DVMATH	DVNAME	FNAME	VINIT	VMIN	VMAX				
DVMATH	MATH1	THICK	0.1	0.01					

Field	*Contents*	*Data Range*	*Default*
DVNAME	Mathematical design variable name. [1]	Character	Required
FNAME	Specifies the name of a mathematical variable defined by a **DVLINK** Bulk Data entry. [2]	Character	Required
VINIT	Initial value of the mathematical variable.	Real	None
VMIN	Specifies the lower bound for the mathematical variable.	Real	-10^{20}
VMAX	Specifies the upper bound for the mathematical variable.	Real	10^{20}

Remarks:

1. Design variable names are used in output presentation. They must be unique.
2. It is not necessary to use a **DVMATH** entry when using **DVLINK**.
3. The **VINIT** value entered here will override the value on a **DVLINK** entry.
4. The values of **VMIN** and **VMAX** specified will override the defaults specified by the **DVLINK** Bulk Data entry.

Bulk Data Entry DVPROP

Defines a physical design variable as an element property, element connection, material property, or modal damping value.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
DVPROP	DVNAME	PTYPE	PRPID	LBOUND	UBOUND	PSYM			

Selecting an Element Property Design Variable

DVPROP	THICK	PSHELL	99			T			

Selecting a Material Property Design Variable

DVPROP	YOUNG	MAT1	10			E			

Field	*Contents*	*Data Range*	*Default*
DVNAME	Physical design variable name. [1]	Character	Required
PTYPE	Selects an element property type, element connection type, or a material type from which the design variable will be selected. [2]	Character	Required
PRPID	Specifies the identification number of the connection, property, or material Bulk Data entry from which the design variable will be selected. [2]	Integer>0	Required
LBOUND	Specifies the lower bound on the selected property **PSYM**.	Real	10^{-6}
UBOUND	Specifies the upper bound on the selected property **PSYM**.	Real	10^{9}
PSYM	Specifies a symbol for the connection, property, material, or damping Bulk Data entry defining the design variable. [2]	Character	Required

Remarks:

1. Design variable names are used in output presentation. They must be unique.
2. The allowable property, element and material design variables are shown in the tables on the following pages.

7 BULK

Rev: V20.1

GROUP	PTYPE	PSYM	DESCRIPTION
PROPERTIES	PBAR	A	Cross Sectional Area
		I1,I2,I12	Area Moments of Inertia
		J	Torsional Constant
		NSM	Nonstructural Mass
		K1,K2	Shear Area Factors
	PBAR1	D1,D2,D3,D4,D5,D6	Cross Sectional Area Parameters
		NSM	Nonstructural Mass
	PBEAM	AA, A1,…,A9, AB	Cross Sectional Areas
		I1A,I11,…,I19,I1B	Area Moments Of Inertia
		I2A,I21,…,I29,I2B	Area Moments Of Inertia
		I12A,I121,…,I129,I12B	Area Products Of Inertia
		JA,J1,…,J9,JB	Torsional Stiffness Parameters
		NSMA, NSM1,…,NSM9, NSMB	Nonstructural Masses
		K1, K2	Shear Stiffness Factors
		S1, S2	Shear Relief Coefficients
		NSIA, NSIB	Moments of Inertia about Nonstructural Mass CG
		CWA, CWB	Warping Coefficients
		YMA, ZMA, YMB, ZMB	Coordinates of Nonstructural Mass CG
		YNA, ZNA, YNB, ZNB	Coordinates of Neutral Axis
	PBEAM1	D1A,D2A,D3A,D4A,D5A	Cross Section Shape Parameters
		D1B,D2B,D3B,D4B,D5B	Cross Section Shape Parameters
		NSMA, NSMB	Nonstructural Mass
	PBUSH	K1, K2,...,K6	Bushing Spring Stiffnesses
		B1, B2,...,B6	Bushing Dampers
	PDAMP	B	Damping Value
	PELAS	K	Stiffness
	PMASS	M	Mass
	PPILE	A,T,D	Cross Sectional Area Parameters
		NSM	Nonstructural Mass
	PPILE1	A	Cross Sectional Area
		I1,I2	Area Moments of Inertia
		J	Torsional Constant
		NSM	Nonstructural Mass
	PROD	A	Cross Sectional Area
		J	Torsional Constant
		NSM	Nonstructural Mass
	PSHEAR	T	Plate Thickness
		NSM	Nonstructural Mass
	PSHELL	T	Plate Thickness
		IFACT	Bending Stiffness Parameter
		TFACT	Transverse Shear Parameter
		NSM	Nonstructural Mass
		Z0	Offset from Reference Plane

CONTINUED ON NEXT PAGE

GROUP	PTYPE	PSYM	DESCRIPTION
ELEMENTS	CDAMP2,4	B	Damping Value
	CELAS2,4	K	Stiffness
	CMASS2,4	M	Mass
	CONM1	Mij	Mass
	CONM2	M Iij	Mass Moments of Inertia
MATERIALS	MAT1	E, G, NU RHO ALPHA GE	Young's and Shear Moduli and Poisson's Ratio Density Thermal Expansion Coefficient Structural Damping Coefficient
	MAT2	Gij RHO ALPHA1, ALPHA2, ALPHA12 GE	Material Property Matrix Density Thermal Expansion Coefficients Structural Damping Coefficient
	MAT8	E1,E2 NU12 Gij RHO ALPHA1, ALPHA2 GE	Young's Moduli Poisson's Ratio Shear Moduli Density Thermal Expansion Coefficients Structural Damping Coefficient
	MAT9	Gij RHO ALPHAi GE	Material Property Matrix Density Thermal Expansion Coefficients Structural Damping Coefficient
MODAL DAMPING	TABDMP1	d1,d2,...	Modal damping as a function of frequency
	TABDMP2	d1,d2,...	Modal damping at specific modes

Bulk Data Entry DVPROPS

Defines a simple linking relationship between physical design variables such as an element properties, element connections or material properties.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DVPROPS	DVNAME	PTYPE1	PRPID1	LB1	UB1	PSYM1			-cont-
-cont-		PTYPE2	PRPID2	LB2	UB2	PSYM2			-cont-
-cont-		*CONTINUES IN GROUPS OF 5*							-cont-

DVPROPS	TAPER	PSHELL	99	0.1	0.2	T			+A
+A		PSHELL	100	0.1	0.2				

Field	*Contents*	*Data Range*	*Default*
DVNAME	Mathematical design variable name. [1]	Character	Required
PTYPEi	Selects an element property type, element connection type, or a material type from which the design variable will be selected. [2]	Character	Required
PRPIDi	Specifies the identification number of the connection, property, or material Bulk Data entry from which the design variable will be selected. [2]	Integer>0	Required
LBi	Specifies the lower bound on the selected property **PSYMi**.	Real	Required
UBi	Specifies the upper bound on the selected property **PSYMi**.	Real	Required
PSYMi	Specifies a symbol for the connection, property, or material Bulk Data entry defining the physical design variable. [3]	Character	Required

Remarks:

1. Design variable names are used in output presentation. They must be unique.
2. The allowable property, element and material design variables are shown in the table following the **DVPROP** entry.
3. This Bulk Data entry is a simple method for defining a design variable linking scheme. The relationship between the **PSYMi** values specified on this entry define the shape that will be maintained during the resizing process. The initial values of **PSYMi** are used as a column in the linking matrix:

$$\begin{Bmatrix} \mathbf{PSYM1} \\ \mathbf{PSYM2} \\ \cdots \\ \mathbf{PSYMn} \end{Bmatrix} = \begin{Bmatrix} \mathbf{0} \\ \mathbf{0} \\ \cdots \\ \mathbf{0} \end{Bmatrix} + \begin{bmatrix} \mathbf{PSYM1_o} \\ \mathbf{PSYM2_o} \\ \cdots \\ \mathbf{PSYMn_o} \end{bmatrix} \mathbf{v}$$

where the $\mathbf{PSYMi_o}$ values are the initial values of the specified properties. The value of the mathematical design variable, **v**, is initially set to 1.0.

7 BULK

Bulk Data Entry DYNRED

Defines dynamic reduction control parameters.

Format and Example:

1	2	3	4	5	6	7	8	9	10
DYNRED	SID	FMAX				NVEC			
DYNRED	1	50.0							

Field	*Contents*	*Data Range*	*Default*
SID	Dynamic reduction set identification number. [1]	Integer>0	Required
FMAX	The highest frequency of interest (cycles / sec).	Real>0	[2,3]
NVEC	Number of desired eigenvectors.	Integer>0	[2]

Remarks:

1. Dynamic reduction must be selected in the Case Control packet with the command:

   DYNRED = SID

2. You must enter either **FMAX** or **NVEC**. If you enter both, the number of eigenvectors has precedence over the frequency value. The most efficient procedure is to enter only **FMAX**.

3. The dynamic reduction procedure will develop generalized coordinates and a transformation matrix between the physical degrees of freedom and these generalized coordinate coordinate degrees of freedom. The generalized coordinates will contain vibration shape properties of the model in the frequency range **0.0** to **FMAX**.

Bulk Data Entries ECHOON, ECHOOFF

Enables or disables the normal echo of Bulk Data entries. These entries are often used with **SKIPON** and **SKIPOFF**.

Formats:

1	2	3	4	5	6	7	8	9	10
ECHOON									

ECHOOFF									

Field	*Contents*	*Data Range*	*Default*

There is no additional input for these entries.

Remarks:

1. The ECHO of consecutive Bulk Data entries may be removed from the print file by insertion of the **ECHOOFF** entry into the Bulk Data packet. ECHO may be reactivated by use of the **ECHOON** entry.
2. As many pairs of **ECHOOFF**, **ECHOON** entries may be used as desired.
3. These entries work for both the sorted and unsorted Bulk Data ECHO listings.

Bulk Data Entry EIGB

Defines buckling analysis eigenvalue extraction control data.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
EIGB	SID	METHOD	FL	FU		NVEC		E	-cont-
-cont-	NORM	GID	DOF						

EIGB	101	LANCZOS	0.5	2.0					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Method of eigenvalue extraction. [2]	Character {**LANCZOS** / **SINV**}	Required
FL, FU	The lower and upper limits specifying the range of buckling load factors desired. [3]	Real **FL<FU**	[4]
NVEC	The maximum number of eigenvectors to be computed.	Integer≥0	[4]
E	Convergence criteria.	Real>0.0	10^{-6}
NORM	Method for normalizing buckling mode shapes. [5]	Character {**MAX** / **POINT**}	MAX
GID	GRID or SCALAR point identification number.	Integer>0	[5]
DOF	Single degree of freedom.	DOF Code	[5]

Remarks:

1. The buckling analysis eigenvalue extraction method set must be selected in the Case Control packet with the command:

   METHOD = SID

2. Buckling analyses of free bodies should not be attempted using the **LANCZOS** method; or, if this is necessary, the **FL** and **FU** ranges should be specified and should not span 0.0 in their range.

3. The quantities **FL** and **FU** specify a range in which the eigenvalues, which represent the dimensionless buckling load factors, are to be found. The load factors are values by which the prebuckling state of stress is multiplied to produce buckling.

4. For the **LANCZOS** method, **NVEC** may be used to request the lowest **NVEC** buckling roots instead of using the **FL** and **FU** data. With the **SINV** method, **FL** and **FU** are required, and **NVEC** is used to (possibly) truncate the number of vectors which are output.

5. If you select **NORM=MAX**, the buckling mode shapes are normalized with respect to the largest component value in the analysis set. Hence, components that are not in the analysis set may have values larger than unity. If you select **NORM=POINT**, the buckling mode shapes are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.

Bulk Data Entry **EIGC (Inverse Power Method)**

MORE EIGC DATA ☞

Specifies complex eigensolution control data for the Inverse Power method which is used to extract a few eigenvalues and eigenvectors in a specified complex frequency range.

Format and Example:

1	2	3	4	5	6	7	8	9	10
EIGC	SID	METHOD	NORM	GID	DOF	E		EPS	-cont-
-cont-	PA1	QA1	PB1	QB1	W1	NE1	ND1		-cont-
-cont-	PA2	QA2	PB2	QB2	W2	NE2	ND2		-cont-

EIGC	14	INV	POINT	27	1	1.-8		1.-6	+A
+A	2.0	5.6	2.0	-3.4	2.0	4	4		+B
+B	-5.5	-5.5	5.6	5.6	1.5	6	3		

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Selects the Inverse Power complex eigenextraction method. [2]	Character **INV**	Required
NORM	Method for normalizing eigenvectors. [3,4]	Character { **MAX** / **POINT** }	Required
GID	GRID or SCALAR point identification number.	Integer>0	[4]
DOF	Single degree of freedom of GRID point **GID**.	DOF Code	[4]
E	Convergence criterion.	Real	10^{-4}
PAi,QAi, PBi,QBi	Complex point pairs defining lines in the complex plane. [5,6,7,8]	Real	Required
Wi	Width of region **i** in complex plane. [5,6,7,8]	Real>0.0	Required
NEi	Estimated number of roots in each region.	Integer>0	Required
NDi	Desired number of roots in each region.	Integer>0	**3*NEi**
EPS	Zero threshhold value. [9]	Real	10^{-24}

Remarks:

1. The complex eigenvalue extraction method set must be selected in the Case Control packet with the command:

   CMETHOD = SID

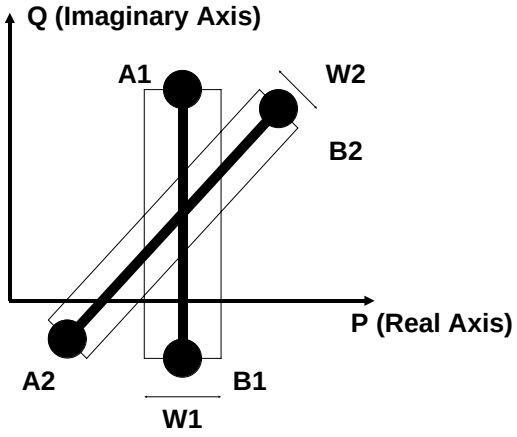
2. The Inverse Power method computes complex eigenvalues and eigenvectors in user specified ranges of the complex plane. This method is most efficient when a relatively small number of roots are required as compared to problem size. If a large number of roots are required, the Upper Hessenberg method may be more efficient.

3. If you select **NORM=MAX**, the eigenvectors are normalized with respect to the largest component value in the analysis set. Hence, components that are not in the analysis set may have values larger than unity.
4. If you select **NORM=POINT**, the eigenvectors are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.
5. Each continuation entry defines a rectangular search region which is used for locating eigenvalues. A maximum of 10 regions may be used and they may overlap. Roots in overlapping regions will not be extracted more than once.
6. The units of **P,Q** and **W** are radians per unit time.
7. A pair **P,Q** defines a complex eigenvalue. From this pair the following may be computed:

Undamped frequency: $f_N = \frac{1}{2\pi}\left[P^2 + Q^2\right]^{1/2}$

Damping coefficient: $\zeta = \frac{-P}{\left[P^2 + Q^2\right]^{1/2}}$

Damped frequency: $f_D = f_N\left[1 - \zeta^2\right]^{1/2}$



For lightly damped systems, **Q** is a measure of the radian frequency and **P** is a measure of the damping.

8. **Wi** should be kept greater than 5 percent of the segment length **Ai** to **Bi** for relatively efficient processing.
9. Eigenvalue components with an absolute value less than or equal to **EPS** are set to zero.

Bulk Data Entry EIGC (ISRR)

MORE EIGC DATA ☞

Specifies complex eigensolution control data for the Iterative Schur-Rayleigh-Ritz method. This method extracts the specified number of roots of the problem and computes the same number of eigenvectors.

Format and Example:

1	2	3	4	5	6	7	8	9	10
EIGC	SID	METHOD	NORM	GID	DOF		ND	EPS	
EIGC	14	ISSR	POINT	27	1		4	1.-6	

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Method of complex eigenvalue extraction. [2]	Character **ISSR**	Required
NORM	Method for normalizing eigenvectors. [3,4]	Character { **MAX** **POINT** }	Required
GID	GRID, SCALAR or EXTRA point identification number.	Integer>0	[4]
DOF	Single degree of freedom of GRID point **GID**.	DOF Code	[4]
ND	The desired number of eigenvalues and eigenvectors to be computed. [5]	Integer>0	Required
EPS	Zero threshhold value. [6]	Real	10^{-24}

Remarks:

1. The complex eigenvalue extraction method set must be selected in the Case Control packet with the command:

   CMETHOD = SID

2. The **ISRR** method is intended for large, sparse problems for which only a few roots are desired.
3. If you select **NORM=MAX**, the eigenvectors are normalized with respect to the largest component value in the analysis set. Hence, components that are not in the analysis set may have values larger than unity.
4. If you select **NORM=POINT**, the eigenvectors are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.
5. The roots are selected from the set of eigenvalues having the smallest moduli. More roots may be found than requested, but the number of eigenvectors extracted will always be less than or equal to the number requested.
6. Eigenvalue components with an absolute value less than or equal to **EPS** are set to zero.

Bulk Data Entry EIGC (Upper Hessenberg Method)

Specifies complex eigensolution control data for the Upper Hessenberg method. This method extracts all roots of the problem and computes a number of user specified eigenvectors.

Format and Example:

1	2	3	4	5	6	7	8	9	10
EIGC	SID	METHOD	NORM	GID	DOF		ND	EPS	

EIGC	14	HESS	POINT	27	1		4	1.-6	

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Method of complex eigenvalue extraction. [2]	Character **HESS**	Required
NORM	Method for normalizing eigenvectors. [3,4]	Character { **MAX** } { **POINT** }	Required
GID	GRID, SCALAR or EXTRA point identification number.	Integer>0	[4]
DOF	Single degree of freedom of GRID point **GID**.	DOF Code	[4]
ND	The desired number of eigenvectors to be computed.	Integer>0	Required
EPS	Zero threshhold value. [5]	Real	10^{-24}

Remarks:

1. The complex eigenvalue extraction method set must be selected in the Case Control packet with the command:

   CMETHOD = SID

2. For the Upper Hessenberg method, both the stiffness and mass matrices may be non-positive definite. This method is most efficient when a large proportion of the roots and vectors are required. If only a relatively few roots are required compared to the problem size, the Inverse Power method may be more efficient.

3. If you select **NORM=MAX**, the eigenvectors are normalized with respect to the largest component value in the analysis set. Hence, components that are not in the analysis set may have values larger than unity.

4. If you select **NORM=POINT**, the eigenvectors are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.

5. Eigenvalue components with an absolute value less than or equal to **EPS** are set to zero.

7 BULK

Bulk Data Entry **EIGR** (GIVENS and Modified GIVENS)

MORE EIGR DATA ☞

Specifies real eigensolution control data for the Givens methods which are used to extract all eigenvalues.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
EIGR	SID	METHOD	FL	FU		NVEC		E	-cont-
-cont-	NORM	GID	DOF						

Requesting Eigenvectors in a Frequency Range:

EIGR	13	GIV	.0	20.0					+A
+A	POINT	32	4						

Requesting a Specified Number of Eigenvectors:

EIGR	13	MGIV				10			+A
+A	POINT	32	4						

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Method of eigenvalue extraction. [2,3]	Character { **GIV** **MGIV** }	Required
FL, **FU**	Frequency range for eigenvector computations. (cycles / sec)	Real>0.0 **FL<FU**	[4]
NVEC	Number of eigenvectors to compute.	Integer>0	**1**
E	Mass orthogonality test parameter. A non-zero value requests a check of the mass orthogonality of the eigenvectors. [5]	Real>0.0	**0.0**
NORM	Method for eigenvectors normalization. [5,6]	Character { **MASS** **MAX** **POINT** }	MAX
GID	GRID or SCALAR point identification number.	Integer>0	[5]
DOF	Single degree of freedom of **GID**.	DOF Code	[5]

Remarks:

1. The real eigenvalue extraction method set must be selected in the Case Control packet with the command:

   METHOD = SID

7 BULK

Rev: V20.1

2. Both the **GIV** and **MGIV** methods are full-spectrum tridiagonalization procedures which compute all eigenvalues and a range of eigenvectors selected by the user. The **GIV** method requires that the a-set mass matrix be positive definite. The **MGIV** method uses an additional transformation to remove this requirement.

3. If **METHOD** is **GIV**, the mass matrix for the analysis set must be positive definite. This means that all degrees of freedom, including rotations, must have mass properties. The **AUTOOMIT** Case Control command or **OMIT** Bulk Data entries may be used to remove massless degrees of freedom. If neither are present, the program will automatically set the condition **AUTOOMIT=YES** for methods **GIV** and **MGIV**.

4. The number of eigenvalues which are computed depend on the values of **FL**, **FU**, and **NVEC**. The following table summarizes the options.

FL	FU	NVEC	Mode Shapes Computed
Blank	Blank	Blank	The lowest mode only.
Blank	Blank	***n_val***	The first *n_val* modes.
Blank	***hi_val***	Blank	All modes between $-\infty$ and ***hi_val***.
Blank	***hi_val***	***n_val***	First ***n_val*** modes in the range $-\infty$ and ***hi_val***.
low_val	Blank	Blank	First mode above ***low_val***.
low_val	Blank	***n_val***	First ***n_val*** modes above ***low_val***.
low_val	***hi_val***	Blank	All modes between ***low_val*** and ***hi_val***.
low_val	***hi_val***	***n_val***	First ***n_val*** modes between ***low_val*** and ***hi_val***.
If you are extracting rigid body modes you should leave the FL Field blank.			

5. You may provide default values for **E** and **NORM** in the Preference File. The defaults are found in the **<Solution Techniques>** group of the **[UAI/NASTRAN]** Configuration Section. You can print the contents of the Preference File with the command:

   PRINT PREFERENCES

   Your System Support Specialist can provide you with more information about this advanced feature.

6. If you select **NORM=MASS**, the eigenvectors are normalized to a unit value of the generalized mass. If you select **NORM=MAX**, the eigenvectors are normalized with respect to the largest component value in the *g-set*. When using the **MAX** normalization with Dynamic Reduction, the *g-set* degrees of freedom, excluding the dynamic reduction generalized coordinates, are used in the normalization process. Finally, if you select **NORM=POINT**, the eigenvectors are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.

Bulk Data Entry **EIGR (INVERSE POWER)**

MORE EIGR DATA ☞

Specifies real eigensolution control data for the Inverse Power method which is used to extract a few eigenvalues in a specified frequency range.

Format and Example:

1	2	3	4	5	6	7	8	9	10
EIGR	SID	METHOD	FL	FU	NEST	NVEC		E	-cont-
-cont-	NORM	GID	DOF						

EIGR	13	INV	1.9	15.6	10	12		1.-6	+A
+A	POINT	32	4						

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Method of eigenvalue extraction.	Character **SINV**	Required
FL, FU	Frequency range of interest (cycles / sec). [2]	Real **FL<FU**	Required
NEST	Estimated number of roots in the frequency range **FL** to **FU**. [2]	Integer>0	Required
NVEC	The number of eigenvectors to be computed. [2]	Integer>0	**3*NEST**
E	The mass orthogonality test and eigenvalue convergence parameter. A non-zero value requests a check of the mass orthogonality of the eigenvectors.	Real>0.0	10^{-10}
NORM	Method for normalizing eigenvectors. [3]	Character { **MASS** **MAX** **POINT** }	MAX
GID	GRID or SCALAR point identification number.	Integer>0	[3]
DOF	Single degree of freedom of **GID**.	DOF Code	[3]

Remarks:

1. The real eigenvalue extraction method set must be selected in the Case Control packet with the command:

   METHOD = SID

2. The number of eigenvalues and eigenvectors extracted depends on the **FL, FU** and **NVEC** values. A summary is given in the table found with entry **EIGR** (**Lanczos**).

3. If you select **NORM=MASS**, the eigenvectors are normalized to a unit value of the generalized mass. If you select **NORM=MAX**, the eigenvectors are normalized with respect to the largest component value in the *g-set*. When using the **MAX** normalization with Dynamic Reduction, the *g-set* degrees of freedom, excluding the dynamic reduction generalized coordinates, are used in the normalization process. Finally, if you select **NORM=POINT**, the eigenvectors are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.

Bulk Data Entry **EIGR (LANCZOS)**

MORE EIGR DATA ☞

Specifies real eigensolution control data for the Lanczos method of eigenvalue extraction.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
EIGR	SID	METHOD	FL	FU		NVEC		E	-cont-
-cont-	NORM	GID	DOF						
EIGR	1	LANCZOS	.0	20.0					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Method of eigenvalue extraction. [2]	Character { **LANCZOS** }	Required
FL, FU	Frequency range for eigenvector computations. (cycles / sec)	Real **FL<FU**	[3]
NVEC	Number of eigenvectors to compute.	Integer	[3]
E	Mass orthogonality test parameter. A non-zero value requests a check of the mass orthogonality of the eigenvectors. [4]	Real>0.0	**0.0**
NORM	Method for eigenvectors normalization. [4,5]	Character { **MASS** **MAX** **POINT** }	MAX
GID	GRID or SCALAR point identification number.	Integer>0	[4]
DOF	Single degree of freedom of **GID**.	DOF Code	[4]

Remarks:

1. The real eigenvalue extraction method set must be selected in the Case Control packet with the command:

   METHOD = SID

2. The Lanczos eigenvalue extraction technique is optimized for processing large, sparse matrices. It is not recommended to perform either Guyan reduction or Dynamic Reduction with the Lanczos technique.

7 BULK

Rev: V20.1

3. The number of eigenvalues and eigenvectors extracted depends on the **FL, FU** and **NVEC** values. A summary is given in the following table:

FL	FU	NVEC	Eigenvalues and Mode Shapes Computed
Blank	Blank	Blank	The lowest mode only.
Blank	Blank	***n_val***	The first *n_val* modes.
Blank	***hi_val***	Blank	All modes between $-\infty$ and ***hi_val***.
Blank	***hi_val***	***n_val***	First ***n_val*** modes in the range $-\infty$ and ***hi_val***.
low_val	Blank	Blank	First mode above ***low_val***.
low_val	Blank	***n_val***	First ***n_val*** modes above ***low_val***.
low_val	***hi_val***	Blank	All modes between ***low_val*** and ***hi_val***.
low_val	***hi_val***	***n_val***	First ***n_val*** modes between ***low_val*** and ***hi_val***.
If you are extracting rigid body modes you should leave the FL Field blank.			

4. You may provide default values for **E** and **NORM** in the Preference File. The defaults are found in the **<Solution Techniques>** group of the **[UAI/NASTRAN]** Configuration Section. You can print the contents of the Preference File with the command:

   PRINT PREFERENCES

   Your System Support Specialist can provide you with more information about this advanced feature.

5. If you select **NORM=MASS**, the eigenvectors are normalized to a unit value of the generalized mass. If you select **NORM=MAX**, the eigenvectors are normalized with respect to the largest component value in the *g-set*. When using the **MAX** normalization with Dynamic Reduction, the *g-set* degrees of freedom, excluding the dynamic reduction generalized coordinates, are used in the normalization process. Finally, if you select **NORM=POINT**, the eigenvectors are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.

6. The Lanczos method may also be selected using the **EIGRL** Bulk Data entry found in other versions of NASTRAN.

Bulk Data Entry EIGR (SUBSPACE ITERATION)

Specifies real eigensolution control data for the Subspace Iteration method of eigenvalue extraction.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
EIGR	SID	METHOD	FL	FU		NVEC	MAXDIM	E	-cont-
-cont-	NORM	GID	DOF						

EIGR	1	SUBS	.0	20.0					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
METHOD	Method of eigenvalue extraction. [2,3]	Character **SUBS**	Required
FL, **FU**	Frequency range for eigenvector computations. (cycles / sec)	Real **FL<FU**	[3]
NVEC	Number of eigenvectors to compute.	Integer	[3]
E	Iteration convergence test and mass orthogonality test parameter.	Real>0.0	$\mathbf{10}^{-5}$
MAXDIM	Maximum dimension of the subspace. [4]	Integer>0	**60**
NORM	Method for eigenvectors normalization. [5,6]	Character { **MASS** **MAX** **POINT** }	**MAX**
GID	GRID or SCALAR point identification number.	Integer>0	[5]
DOF	Single degree of freedom of **GID**.	DOF Code	[5]

Remarks:

1. The real eigenvalue extraction method set must be selected in the Case Control packet with the command:

   METHOD = SID

2. The Subspace Iteration eigenvalue extraction technique is optimized for unreduced Fluid-Structure Interaction (FSI) models. It is not recommended to perform either Guyan reduction or Dynamic Reduction with this method.

7 BULK

3. The number of eigenvalues and eigenvectors extracted depends on the **FL, FU** and **NVEC** values. A summary is given in the following table:

FL	FU	NVEC	Eigenvalues and Mode Shapes Computed
Blank	Blank	Blank	The lowest mode only.
Blank	Blank	***n_val***	The first *n_val* modes.
Blank	***hi_val***	Blank	All modes between $-\infty$ and ***hi_val***.
Blank	***hi_val***	***n_val***	First ***n_val*** modes in the range $-\infty$ and ***hi_val***.
low_val	Blank	Blank	First mode above ***low_val***.
low_val	Blank	***n_val***	First ***n_val*** modes above ***low_val***.
low_val	***hi_val***	Blank	All modes between ***low_val*** and ***hi_val***.
low_val	***hi_val***	***n_val***	First ***n_val*** modes between ***low_val*** and ***hi_val***.
If you are extracting rigid body modes you should leave the FL Field blank.			

4. The Subspace Iteration method extract eigenvalues one block at a time. The block size is determined by **FU**, **NVEC**, and **MAXDIM**. If the requested number of eigenvectors is greater than **MAXDIM**, then the block size is set to one half of **MAXDIM.**

5. If you select **NORM=MASS**, the eigenvectors are normalized to a unit value of the generalized mass. If you select **NORM=MAX**, the eigenvectors are normalized with respect to the largest component value in the *g-set*. When using the **MAX** normalization with Dynamic Reduction, the *g-set* degrees of freedom, excluding the dynamic reduction generalized coordinates, are used in the normalization process. Finally, if you select **NORM=POINT**, the eigenvectors are normalized with respect to the value of the component defined by **GID** and **DOF**. This component must be in the analysis set.

6. You may provide a default value for **NORM** in the Preference File. The defaults is found in the **<Solution Techniques>** group of the **[UAI/NASTRAN]** Configuration Section. You can print the contents of the Preference File with the command:

   PRINT PREFERENCES

   Your System Support Specialist can provide you with more information about this advanced feature.

Bulk Data Entry **EPOINT**

Defines EXTRA points as generalized coordinates for use in dynamics analyses.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
EPOINT	*EXTRA POINT ID LIST*								-cont-

EPOINT	108	1294	234	235	100	THRU	120		

Field	*Contents*	*Data Range*	*Default*
EXTRA POINT ID LIST	List of one or more EXTRA point identification numbers. [1,2]	Integer>0 [2]	Required

Remarks:

1. All EXTRA point identification numbers must be unique with respect to all other GRID, SCALAR and PRESSURE points.
2. The maximum value for the EXTRA Point identification number is one tenth of the largest integer that can be represented on the **UAI*/NASTRAN*** host computer.
3. The EXTRA points are the coordinates used in the definition of the transfer function on the **TF** Bulk Data entries.

Bulk Data Entry FLFREE

Defines a list of fluid GRID points on a free surface. These GRID points must be attached to fluid solid elements HEXA, PENTA or TETRA.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FLFREE	*GRID ID LIST*								-cont-
FLFREE	101	THRU	145	148	150				

Field	*Contents*	*Data Range*	*Default*
GRID ID LIST	List of one or more identification numbers of fluid GRID points on the free surface.	Integer>0	[1]

Remarks:

1. If free surface effects are desired in a fluid/structure interaction model, this entry must be used to define the GRID points on the surface, and the **FSIDATA** entry must be used to define a gravity vector.

7 BULK

Bulk Data Entry FLSOLVE

Specifies control data for the K and PK Flutter solutions.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FLSOLVE	FID	METHOD	VCUT	GCUT	GOFF	EPS			
FLSOLVE	11	LAGRANGE	1200.0	0.001	-0.05	0.0			

Field	*Comments*	*Data Range*	*Default*
FID	Flutter solution control identification Number [1].	Integer>0	Required
METHOD	Method of Generalized Aerodynamic Force interpolation.	Keyword { **CUBIC** / **LAGRANGE** }	**LAGRANGE**
VCUT	Velocity cutoff for Flutter Speed crossings search.[3]	Real>0.0	10^6
GCUT	Damping cutoff for Flutter Speed crossings search.[3]	Real≤0.0	**0.0**
GOFF	Damping offset for Flutter Speed crossings search.[3]	Real≥0.0	**0.0**
EPS	Convergence criteria for PK-ITER method of flutter solution. [4]	Real>0.0	**0.01**

Remarks

1. An **FLSOLVE** entry must be selected in the Case Control packet with the command:

   FLSOLVE = FID

2. There are two methods of interpolation, a cubic spline or a Langrangian polynomial interpolation.

3. The Flutter speed crossings search is limited to crossings below the cutoff speed **VCUT**, and for branches that produce a minimum stable damping below **GCUT**. The location of the Flutter speed crossing is defined as the speed at which the damping exceeds **GOFF**.

   For example, the user may wish all crossings above a limit speed by setting **VCUT**. To ignore a mode that shows a lightly unstable damping, **GOFF** can be used to define flutter where the damping exceeds **GOFF**. **GCUT** can be used to ignore modes that never generate significant stable damping. This can be used to ignore lightly damped modes while retaining the flutter crossing as defined by 0.0 damping.

4. **EPS** defines the convergence criteria of a root as the maximum difference between the reduced frequency of the aerodynamic frequency estimate and the structural eigenvalue of interest.

7 BULK

Bulk Data Entry FLSTR

Defines a list of structural elements which are in contact with a specific fluid element in fluid/structure interaction models.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FLSTR	EIDF	*ELEMENT ID LIST*							-cont-
FLSTR	101	201	221	THRU	281	BY	20		

Field	*Contents*	*Data Range*	*Default*
EIDF	Identification number of a fluid element in contact with one or more structural elements.	Integer>0	[1,2]
ELEMENT ID LIST	List of one or more identification numbers of ***structural*** elements in contact with the referenced fluid element.	Integer>0	[1,2]

Remarks:

1. If fluid/structure interaction effects are desired in a fluid/structure interaction model, this entry must be used to define the contacts between fluid and structural elements. Otherwise, fluid elements not completely surrounded by other fluid elements will be assumed placed against a rigid boundary.
2. This entry allows multiple structural elements to contact the same fluid element, thus allowing for dissimilar meshes between the structural and fluid elements. Normally, the fluid mesh may be coarser than the structural mesh for a given frequency range.

7 BULK

Bulk Data Entry FORCE

Defines a static load at a GRID point of the form:

$$\vec{f} = \mathbf{F} \cdot \vec{\mathbf{V}}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
FORCE	LID	GID	CID	F	V1	V2	V3		
FORCE	102	527	205	2.9	0.0	0.0	1.0		

Field	*Contents*		
LID	Load set identification number. [1]	Integer>0	Required
GID	GRID point identification number.	Integer>0	Required
CID	Coordinate system identification number used for definition of the force vector. [2]	Integer>0	**0**
F	Load scale factor.	Real	Required
Vi	Components of the load vector measured in coordinate system **CID**.	Real	[3]

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. When **CID** is not defined, the force vector will be applied in the Basic Coordinate System.

3. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.

7 BULK

Bulk Data Entry FORCE1

Defines a static load at a GRID point of the form:

$$\vec{f} = \mathbf{F} \cdot \frac{\mathbf{GID1}\vec{\ }\mathbf{GID2}}{|\ \mathbf{GID1}\vec{\ }\mathbf{GID2}\ |}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
FORCE1	LID	GID	F	GID1	GID2				
FORCE1	106	113	8.76	116	100				

Field	*Contents*		
LID	Load set identification number. [1]	Integer>0	Required
GID	GRID point identification number.	Integer>0	Required
F	Value of load.	Real	Required
GIDi	GRID point identification numbers defining the direction of the resulting force vector. [2]	Integer>0 **GID1≠GID2**	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. The direction of the resulting force vector is determined by the normalized vector from **GID1** to **GID2**.

7 BULK

Bulk Data Entry FORCE2

Defines a static load at a GRID point of the form:

$$\vec{f} = \mathbf{F} \cdot \frac{\mathbf{GID1}\vec{\ }\mathbf{GID2} \times \mathbf{GID3}\vec{\ }\mathbf{GID4}}{|\ \mathbf{GID1}\vec{\ }\mathbf{GID2} \times \mathbf{GID3}\vec{\ }\mathbf{GID4}\ |}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
FORCE2	LID	GID	F	GID1	GID2	GID3	GID4		
FORCE2	116	113	76.3	116	113	117	118		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
GID	GRID point identification number where load is applied.	Integer>0	Required
F	Value of load.	Real	Required
GIDi	GRID point identification numbers defining the direction of the resulting force vector. [2]	Integer>0 **GID1≠GID2** **GID3≠GID4**	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. The direction of the resulting force vector is determined by the normalized cross product of two vectors, defined from **GID1** to **GID2** and from **GID3** to **GID4**.

7 BULK

Bulk Data Entry FORCEAX

Defines a static load of the form:

$$\vec{f} = \mathbf{F} \cdot \vec{\mathbf{V}}$$

for use in an axisymmetric harmonic analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FORCEAX	LID	RID	HID	F	V1	V2	V3		
FORCEAX	102	103	3	1.5	0.0	1.5	0.0		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
RID	**RINGAX** identification number.	Integer>0	Required
HID	Harmonic identification number. [2]	Integer>0 or Harmonic Code	Required
F	Load scale factor.	Real	Required
Vi	Components of load vector in the cylindrical coordinate system.	Real	[3]

Remarks:

1. Axisymmetric loads must be selected in the Case Control packet with the command:

   LOAD = LID

2. A separate entry is needed to define the force associated with each harmonic.
3. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.
4. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

7 BULK

Bulk Data Entry FREQ

Defines an explicit set of frequencies to be used in the solution of frequency response problems.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FREQ	FREQID	F1	F2	F3	F4	F5	F6	F7	-cont-
-cont-	F8	F9	*CONTINUES WITH LIST OF VALUES*						-cont-

FREQ	101	11.5	14.7	19.6	23.3	28.6	33.3	39.6	+A
+A	32.6	29.1							

Field	*Contents*	*Data Range*	*Default*
FREQID	Frequency set identification number. [1,2]	Integer>0	Required
Fi	Frequency values. [3]	Real>0.0	Required

Remarks:

1. Frequency sets must be selected in the Case Control packet with the command:

   FREQ = FREQID

2. Any number of **FREQ**, **FREQ1**, and **FREQ2** entries may be combined by assigning the same frequency set identification number, **FREQID**, to them.

3. Frequencies are entered in cycles per unit time.

7 BULK

Bulk Data Entry FREQ1

Defines a set of frequencies to be used in the solution of frequency response problems by specification of a starting frequency, frequency increment and number of increments desired.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FREQ1	FREQID	FL	DF	NDF					
FREQ1	101	4.3	0.25	24					

Field	*Contents*	*Data Range*	*Default*
FREQID	Frequency set identification number. [1,2]	Integer>0	Required
FL	First and lowest frequency in set. [3,4]	Real≥0.0	Required
DF	Frequency increment. [3,4]	Real>0.0	Required
NDF	Number of frequency increments. [4]	Integer>0	Required

Remarks:

1. Frequency sets must be selected in the Case Control packet with the command:

   FREQ = FREQID

2. Any number of **FREQ**, **FREQ1**, and **FREQ2** entries may be combined by assigning the same frequency set identification number, **FREQID**, to them.
3. Frequencies are entered in cycles per unit time.
4. The frequencies defined by these data are given by the following recurrence relationship:

   $f_1 = \mathbf{FL}$

   $f_{i+1} = f_i + \mathbf{DF} \quad i = 1, \ldots, \mathbf{NDF}$

7 BULK

Bulk Data Entry FREQ2

Defines a set of frequencies to be used in the solution of frequency response problems by specification of a lower frequency, an upper frequency, and the number of logarithmic increments desired.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FREQ2	FREQID	FL	FU	NF					
FREQ2	101	1.0	1.E5	5					

Field	*Contents*	*Data Range*	*Default*
FREQID	Frequency set identification number. [1,2]	Integer>0	Required
FL, FU	Lower and upper frequencies. [3]	Real>0.0 **FL**<**FU**	Required
NF	Number of logarithmic increments. [4]	Integer>0	Required

Remarks:

1. Frequency sets must be selected in the Case Control packet with the command:

   FREQ = FREQID

2. Any number of **FREQ**, **FREQ1**, and **FREQ2** entries may be combined by assigning the same frequency set identification number, **FREQID**, to them.

3. Frequencies are entered in cycles per unit time. Because of the logarithmic function, frequencies must be greater than 0.

4. The frequencies defined by these data are given by:

   $$f_i = \mathbf{FL}\, e^{(i-1)d} \qquad i = 1, \ldots, \mathbf{NF}+1$$

   where: $$d = \frac{1}{\mathbf{NF}} \log_e \left(\frac{\mathbf{FU}}{\mathbf{FL}} \right)$$

   The frequencies may also be defined using the following equivalent form:

   $$f_i = \mathbf{FL} \left(\frac{\mathbf{FU}}{\mathbf{FL}} \right)^{\frac{(i-1)}{\mathbf{NF}}} \qquad i = 1, \ldots, \mathbf{NF}+1$$

   For the example shown, the list of frequencies will be 1.0, 10.0, 100.0, 1000.0, 10000.0, and 100000.0 cycles per unit time.

7 BULK

Bulk Data Entry FREQ3

Defines a set of frequencies, taken from the computed modal frequencies, to be used in the solution of modal frequency response problems. Frequencies are computed from a specified range of either frequencies or mode numbers that reference the eigenvalue analysis performed. Additional frequencies may be generated within the frequency range, with a provision for extrapolation beyond the computed normal modes, using an increment parameter and parameters controlling the type of spacing between increments.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FREQ3	FREQID	FL	FU	NINC	TYPE	FACTOR	ZERO	E	
		NL	NU						
FREQ3	101	1.0	100.						
FREQ3	102	1	20	8	LINEAR	0.5			

Field	*Contents*	*Data Range*	*Default*
FREQID	Frequency set identification number. [1,2]	Integer>0	Required
FL,FU	Lower and upper frequencies. [3,4,5,6]	Real≥0.0 **FL**<**FU**	Required
NL,NU	Lower and upper mode numbers. [4,6]	Integer>0 **NL**<**NU**	Required
NINC	Number of frequency increments between modal frequencies. [7]	Integer≥1	**1**
TYPE	Type of interpolation to use when computing incremental frequencies. [8]	Character { **LOG** / LINEAR }	LOG
FACTOR	Interpolation factor to use when computing incremental frequencies. [8]	Real>0.0	**0.5**
ZERO	Defines a non-zero floating point number used in the frequency interpolation algorithm for logarithmic increments. [9]	Real>0.0	10^{-4}
E	Tolerance used in identifying duplicate frequencies expressed as a fraction of the largest computed frequency increment. [10]	Real>0.0	10^{-4}

Remarks:

1. Frequency sets must be selected in the Case Control packet with the command:

   FREQ = FREQID

2. Any number of **FREQ**, **FREQ1**, and **FREQ2** entries may be combined by assigning the same frequency set identification number, **FREQID**, to them. However, only one **FREQ3** entry may appear in each SUBCASE.

7 BULK

3. Frequencies are entered in cycles per unit time.
4. The normal mode frequencies between **FL** and **FU**, as well as the values of **FL** and **FU** themselves, are added to the frequency set. Alternately, fields **NL** and **NU** may be entered as mode identification numbers. These two options may not be mixed; either two frequencies or two mode identificastion numbers must be used. When **NL** is specified as 0, it implies that the lower bound frequency value is 0.0.
5. When either of **FL** and **FU** is outside the range of the calculated modes, then extrapolation is performed to generate the requested frequencies.
6. If both **FL** and **FU** are outside the range of the calculated modes, then the solution is performed at only **FL** and **FU**, i.e. no additional frequencies are generated, and a warning message is issued.
7. When **NINC = 1**, the default value, the only frequencies generated by **FREQ3** are the actual modal frequencies. With **NINC = 2**, one new frequency between each modal frequency is generated, yielding 2 frequency increments between each natural frequency, etc.
8. The size of each frequency increment is determined by parameters **TYPE** and **FACTOR**. If **TYPE = LINEAR** and **FACTOR = 1.0**, all frequency increments between a pair of modal frequencies are the same size. If **TYPE = LOG**, the increment sizes are adjusted using a log function. If **FACTOR** is less than 1.0, then each frequency increment closer to a modal frequency is of a size **FACTOR** times the size of its adjacent increment. This causes the frequency increments near the modal frequencies to be smaller than the increments near the center of the frequency range between each modal frequency.
9. The value for **ZERO** replaces any floating point values of 0.0 for the purposes of logarithmic interpolation.
10. After the frequencies generated by all **FREQi** Bulk Data entries are merged, the maximum increment between consecutive frequencies, Δf, is computed. Consecutive frequencies separated by less than $E \cdot \Delta f$ are considered to be duplicates, all but the lowest is discarded, thus avoiding redundant solutions.

Bulk Data Entry FSIDATA

Defines fluid-structure interaction data.

Format and Example:

1	2	3	4	5	6	7	8	9	10
FSIDATA	GRAVID	TOLER	PREFDB	FSIEPS	DBOUT	CFREQ	NTANKS	GID	-cont-
-cont-	DOF								

FSIDATA	3		20.-6			YES			

Field	*Contents*	*Data Range*	*Default*
GRAVID	Identification number of a **GRAV** Bulk Data entry. [1]	Integer>0	Required
TOLER	Tolerance,, in units of length, used to determine points on the fluid-structure interface.	Real≥0.0	[2]
PREFDB	Reference sound pressure level. [3]	Real>0.0	**1.0**
FSIEPS	The fluid stiffness matrix inversion parameter. [4]	Real≥0.0	10^{-6}
DBOUT	Selects the sound pressure level output method. [3]	Character { PEAK / **RMS** }	**RMS**
CFREQ	Specifies that input frequencies are defined as the center frequencies of $1/n$-octave bands. [5]	Character { YES / **NO** }	**NO**
NTANKS	Specifies the number of of isolated fluid compatments, or *tanks*.	Integer≥0.0	[6]
GID	GRID point identification number. [7]	Integer>0	[7]
DOF	Single degree of freedom for **GID**. [7]	DOF Code	[7]

Remarks:

1. A gravity identification number must be input to define free surface effects, such as slosh, in fluid-structure interaction models.
2. The default value for **TOLER** is 0.3 times the minimum edge length of all fluid elements and structural elements which may be on the interface between the fluid and structure in the model. This includes the HEXA, PENTA,TETRA, QUAD4, and TRIA3 elements.
3. The sound pressure in dB is defined as: $dB = 20 \log (p/\text{PREFDB})$. The value taken for p depends on the **DBOUT** field. It may be selected as the **PEAK** or **RMS** value. Note that for air, **PREFDB** is 20×10^{-6} Pa.
4. **FSIEPS** is a parameter used to improve the conditioning of the fluid stiffness matrix during inversion. See Chapter 23 of the **UAI/*NASTRAN User's Guide*** for additional information.

7 BULK

Rev: V20.1

5. When performing Frequency Response analyses, **CFREQ=YES** indicates that center frequencies were specified in the input frequency lists defined by **FREQi** Bulk Data entries. In such cases, you should use a $1/n$-octave band frequency which results in more accurate combine $dB(A)$ output.
6. The default value is one tank. The program will automatically treat the constant pressure mode of a single fluid cavity. If you have multiple tanks that introduce constant pressure modes, then you must specify **NTANKS**. Note that a tank is a volume of fluid which is completely enclosed by structural elements. Do not include any compartments whose constant pressure mode has been removed by an **SPC** in your count of **NTANKS**.
7. By default, the free surface of a fluid in any FSI model is grounded in the direction of gravity. To solve a free-free system, you must specify **GID,DOF**. When this is done, the motions of the free surface are measured relative to **DOF**. The GRID point **GID** must be located on the line of the center of gravity of the free surface, and the direction of **DOF** must be aligned with the gravity vector.

Bulk Data Entry GENEL

Defines a general element of the structural model by a stiffness or flexibility matrix.

Format:

1	2	3	4	5	6	7	8	9	10
GENEL	EID		GIDI1	DOFI1	GIDI2	DOFI2	GIDI3	DOFI3	-cont-
-cont-	GIDI4	DOFI4	*CONTINUES IN GROUPS OF 2*						-cont-
-cont-	"UD"		GIDD1	DOFD1	GIDD2	DOFD2	GIDD3	DOFD3	-cont-
-cont-	GIDD4	DOFD4	*CONTINUES IN GROUPS OF 2*						-cont-
-cont-	"K"/"Z"	K11	K21	K31	...	K22	K32	K42	-cont-
-cont-	...	K33	K43	K53	*CONTINUES WITH LIST OF TERMS*				-cont-
-cont-	"S"	S11	S12	S13	S14	...	S21	S22	-cont-
-cont-	S23	...	S31	S32	*CONTINUES WITH LIST OF TERMS*				-cont-

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GIDIi	GRID or SCALAR point identification numbers of points in the **GI** list.	Integer>0	Required
DOFIi	Single degree of freedom corresponding to the points **GIDIi**.	DOF Code	Required
"UD"	Indicates that start of the **GD** degrees of freedom.	Character	Required
GIDDi	GRID or SCALAR point identification numbers of points in the **GD** list.	Integer>0	Required
DOFDi	Single degree of freedom corresponding to the points **GIDDi**.	DOF Code	Required
"K","Z"	Indicates the start of the element stiffness, **K**, or flexibility, **Z**, matrix.	Character	Required
Kij	Elements of the **K** or **Z** matrix. [2]	Real	**0.0**
"S"	Indicates the start of data defining the rigid body, **S**, matrix.	Character	Required
Sij	Elements of the **S** matrix. [3]	Real	**0.0**

Remarks:

1. Element identification numbers must be unique.
2. The **K** or **Z** matrices are entered as lower triangular matrices by columns. High precision input format may be used.
3. The **S** matrix is entered by rows.
4. There are four distinct sections of data to input; the **GI** list, the **UD** list, the **K** or **Z** matrix, and the **S** matrix.

7 BULK

Bulk Data Entry GPFIELD

Defines a two- or three-dimensional stress/strain field for GRID point stress or strain recovery.

Format and Example:

1	2	3	4	5	6	7	8	9	10
GPFIELD	FID	TYPE	CIDOUT	AXIS	NORM			TOLER	-cont-
-cont-	*ELEMENT ID LIST*								-cont-

GPFIELD	20	SHELL	5	X2					
	22	23	25	THRU	35				

Field	*Contents*	*Data Range*	*Default*
FID	Stress/strain field identification number. [1,2]	Integer>0	Required
TYPE	Stress/strain field type.	Character { **SHELL** } { **SOLID** }	Required
CIDOUT	Stress/strain field output coordinate system identification number. [3]	Integer≥0	[3]
AXIS	Nominal x-axis of **CIDOUT**. [4,5]	Character { **X1** } { **X2** } { **X3** }	X1
NORM	Nominal z-axis of **CIDOUT**. [4,5]	Character { **X1** or **NX1** } { **X2** or **NX2** } { **X3** or **NX3** }	X3
TOLER	Tolerance angle, in degrees, for interelement continuity tests. [6]	Real	**0.0**
EIDi	Identification numbers of elements comprising the stress field.	Integer>0	Required

Remarks:

1. Stress/strain field set identification numbers must be unique.
2. **GPFIELD** data may be referenced indirectly by the Case Control commands:

   ```
   GPSTRESS = FIELDSET fsid    or
   GPSTRAIN - FIELDSET fsid    where

   SET fsid = fid1, fid2, ....
   ```

   However, GRID point stresses and/or strains may be requested without defining a stress/strain field.
3. If **CIDOUT** is not specified, the output will be in the Basic Coordinate System.

7 BULK

4. The symbols used to represent the **AXIS** and **NORM** coordinate directions are:

SYMBOL	Rectangular Systems	Cylindrical Systems	Spherical Systems	SYMBOL	Rectangular Systems	Cylindrical Systems	Spherical Systems
X1	x	r	r	NX1	-x	-r	-r
X2	y	q	q	NX2	-y	-q	-q
X3	z	z	f	NX3	-z	-z	-f

5. The **AXIS** and **NORM** directions may not be coincident.

6. The **TOLER** field is ignored for **SOLID** stress/strain fields. A value of zero for **SHELL** stress/strain fields causes the interelement continiuity tests not to be performed.

7. GRID point stress and strain results may be obtained only for HEXA, PENTA, QUAD4, QUAD8, TETRA, TRIA3, and TRIA6 elements.

Bulk Data Entry GRAV

Defines a gravity loading on the structural model of the form:

$$\vec{g} = \mathbf{G} \cdot \vec{\mathbf{V}}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
GRAV	LID	CID	G	V1	V2	V3			
GRAV	101	100	32.2	0.0	0.0	-1.0			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
CID	Coordinate system identification number. [3]	Integer≥0	**0**
G	Gravity vector scale factor. [4]	Real	Required
Vi	Components of the gravity vector measured in coordinate system **CID**.	Real	[5]

Remarks:

1. Gravity loads may be combined with other loads, such as **FORCE** and **MOMENT** by using **LOAD** Bulk Data entries. The **LID** on the **GRAV** entry may not be the same as that of any other load data.
2. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

3. The gravity vector is defined in coordinate system **CID**. If **CID** is not specified, the gravity vector is defined in the Basic Coordinate System. Positive gravity loads are in the direction of the gravity vector.
4. The gravity vector scale factor must take into account the mass and length units of the model as well as the magnitude of the gravity vector.
5. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.

7 BULK

Bulk Data Entry GRDSET

Defines default options for coordinate systems and constraints for **GRID** Bulk Data entries.

Format and Example:

1	2	3	4	5	6	7	8	9	10
GRDSET		CIDIN				CIDOUT	PSPC		

GRDSET		115				267	1246		

Field	*Contents*	*Data Range*	*Default*
CIDIN	Identification number of coordinate system in which the location of GRID points not having a value in Field 3 of their **GRID** entries is defined.	Integer>0	[1]
CIDOUT	Identification number of coordinate system in which vector quantities are measured at GRID points not having a value in field 7 of their **GRID** entries.	Integer>0	[1]
PSPC	Permanent single-point constraints associated with GRID points not having a value in Field 8 of their **GRID** entries.	DOF Code	[1]

Remarks:

1. At least one of the entries **CIDIN**, **CIDOUT** or **PSPC** must be nonblank.
2. The contents of fields on this entry are used for any **GRID** entry whose corresponding fields are blank. The default may be overridden by placing a zero in the appropriate field of the **GRID** entry.
3. Only one **GRDSET** entry may appear in the Bulk Data packet.

7 BULK

Bulk Data Entry GRID

Defines the location of a geometric GRID point of the structural model, the direction of GRID point output quantities, and its permanent single-point constraints.

Format and Example:

1	2	3	4	5	6	7	8	9	10
GRID	GID	CIDIN	X1	X2	X3	CIDOUT	PSPC		
GRID	101	11	1.6	10.2	15.8		124		

Field	*Contents*	*Data Range*	*Default*
GID	GRID point identification number. [1,2]	Integer>0 [2]	Required
CIDIN	Identification number of coordinate system in which the location of the GRID point is defined. [3]	Integer≥0	**0**
Xi	Location of the GRID point in coordinate system **CIDIN**. [4]	Real	Required
CIDOUT	Identification number of coordinate system in which displacements, degrees of freedom, constraints, and solution vectors are defined at the GRID point. [5]	Integer≥0	**0**
PSPC	Permanent single-point constraints associated with GRID point.	DOF Code	**0**

Remarks:

1. GRID point identification numbers must be unique with respect to all other GRID, SCALAR, EXTRA and PRESSURE points.
2. The maximum value for the GRID Point identification number is one tenth of the largest integer that can be represented on the **UAI/*NASTRAN*** host computer.
3. The Basic Coordinate System is the default coordinate system.
4. The coordinates **X1**, **X2** and **X3** depend on the type of the coordinate system **CIDIN**. If the system is rectangular, they are (x,y,z); if it is cylindrical, they are (r,θ,z); and if it is spherical, they are (r,θ,ϕ).
5. The collection of all **CIDOUT** coordinate systems defined on all **GRID** entries is called the Global Coordinate System. All degrees of freedom, constraints and solution vectors are expressed in the Global Coordinate System.
6. The **GRDSET** entry can be used to define default options for coordinate systems and constraints.

7 BULK

Bulk Data Entry GTRAN

Defines the output coordinate system transformation to be applied to the displacements of a selected GRID point in a specified substructure.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
GTRAN	SID	SNAME	GID	TID					

GTRAN	44	GIMBAL	1067	45					

Field	*Contents*	*Data Range*	*Default*
SID	Identification number of the transformation set. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GID	GRID point identification number.	Integer>0	Required
TID	Identification number of a **TRANS** entry. [2,3]	Integer≥0	**0**

Remarks:

1. Transformation sets must be selected in the Substructure Control packet with the

   TRANS = SID

   subcommand of the substructure **COMBINE** command.

2. If **TID=SID**, the point will remain fixed to the substructure with no transformation being made.

3. If **TID=0**, the displacement set at the GRID point will be transformed to the Basic Coordinate System of the COMBINEd substructure.

7 BULK

Bulk Data Entry INCLUDE

Merges the contents of another file into the **UAI/*NASTRAN*** input file.

Format and Example:

```
   1       2       3       4       5       6       7       8       9       10
INCLUDE   ['] phys_name ['] [,params]
```

```
INCLUDE   'materials.lib', OLD
```

Field	*Contents*		
phys_name	Specifies the host computer dependent name of the physical file containing data that you wish to have included in your input data stream. [1,2]	Character	Required
params	Selects host computer dependent parameters. [3]	Character	None

Remarks:

1. Physical file name vary depending on your host computer. See Chapter 1 for details.
2. The character string that you specify for *phys_name* is always converted to upper case characters by **UAI/*NASTRAN***. If your host computer has case-sensitive file names, and if you wish to use lower case characters, then you must enclose *phys_name* in single quotation marks, sometimes called tics, as shown in the example above.
3. Other host computer dependent parameters may be available. See Chapter 1.
4. An **INCLUDE** file may not contain any other **INCLUDE** commands.
5. Any number of **INCLUDE** commands may appear anywhere in the Bulk Data packet.

Bulk Data Entry LOAD

Defines a static load as a linear combination of load sets of the form:

$$L = \mathbf{S}\sum_{i} \mathbf{Si}\, L_i$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
LOAD	LID	S	S1	LID1	S2	LID2	S3	LID3	-cont-
-cont-	S4	LID4	*CONTINUES IN GROUPS OF 2*						-cont-

LOAD	201	2.25	1.0	501	5.4	502			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
S	Scale factor applied to the combined load set **LID**.	Real	Required
Si	Scale factor for load set **LIDi**.	Real	Required
LIDi	Set identification numbers of loads being combined. [3,4]	Integer>0	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. The load set identification numbers, **LID** and **LIDi**, may not be the same as any other applied load or **SPCD** data entries.
3. The **LIDi** must be unique and may not reference any other **LOAD** entry.
4. This entry must be used if gravity loads (**GRAV** entries) are to be combined with any other load types.

7 BULK

Bulk Data Entry LOADC

Defines the static load for a Substructuring analysis as a linear combination of load sets defined for each Basic Substructure. The form of the load is:

$$L = \mathbf{S} \sum_{i} \mathbf{Si}\, L_i$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
LOADC	LID	S	SNAME1	LID1	S1	SNAME2	LID2	S2	-cont-
-cont-			SNAME3	LID3	S3	*CONTINUES IN GROUPS OF 3*			-cont-

LOADC	27	1.0	WINGR	5	0.5	FUSLG	966	2.5	+A
+A			WINGL	6	0.5				

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
S	Scale factor applied to the combined load.	Real	Required
SNAMEi	Basic Substructure name. [2]	Name	Required
LIDi	Identification number of a load set in substructure **SNAMEi**. [2,3,4]	Integer>0	Required
Si	Scale factor for load set **SIDi**.	Real	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. It is not necessary for the **SNAMEi** and **LIDi** to be unique.
3. The load set identification numbers **LIDi** reference the load sets selected in PHASE1 SUBCASEs. These Basic Substructure loads are combined and transformed as necessary for application to the solution substructure.
4. The **LOADC** entry is used to specify a static loading condition in a PHASE2 Substructure analysis. The **LIDi** may also reference temperature loads or element deformation loads defined in PHASE1.

7 BULK

Bulk Data Entry LOADCYH

Defines the harmonic coefficients of a static load for use in Cyclic Symmetry analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
LOADCYH	LID	S	HID	HTYPE	S1	LID1	S2	LID2	
LOADCYH	1	2.5	2	GRAV	2.	101			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
S	Scale factor applied to the combined load.	Real	Required
HID	Harmonic identification number. [2]	Integer>0	Required
HTYPE	Harmonic type. [2,3]	Character { **C**, **S**, **CSTAR**, **SSTAR**, **GRAV**, **RFORCE** }	Blank
Si	Scale factor applied to the specified load set. [2]	Real	Required
LIDi	Load set identification number. [2,3,4,5,6]	Integer>0	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. If **HTYPE** is **GRAV** or **RFORCE**, gravity or centrifugal force loading will be used. Harmonic loads for appropriate available harmonics will be generated automatically in these cases. Also, the entry in **HID** will be ignored and hence may be blank. **S2** and **LID2** must be blank for this case.
3. If **HTYPE** is blank, the load will be applied to all applicable types in the problems.
4. **LID1** and **LID2** may reference any static loading data, including **GRAV** and **RFORCE**. They may also reference a **LOAD** Bulk Data Entry.
5. If **LIDi** references an enforced displacement load, defined with **SPCD** Bulk Data, the same **LID** may not be used for any other load data.
6. The same load set may not be referenced by both **LOADCYN** and **LOADCYH** data.

7 BULK

Bulk Data Entry LOADCYN

Defines a physical static load for use in Cyclic Symmetry analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
LOADCYN	LID	S	SEGID	SEGTYP	S1	LID1	S2	LID2	
LOADCYN	101	1.0	2	R	2.0	102			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
S	Scale factor applied to the combined load.	Real	Required
SEGID	Segment identification number.	Integer>0	Required
SEGTYP	Segment type.	Character {**L** / **R**}	[2]
Si	Scale factor applied to the specified load set.	Real	Required
LIDi	Load set identification number. [3,4,5]	Integer>0	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. If **SEGTYP** is blank, both **R** and **L** segments will be used when symmetry is dihedral.
3. **LID1** and **LID2** may reference any static loading data, except **GRAV** and **RFORCE**. If they reference a **LOAD** Bulk Data entry, then that entry may not reference **GRAV** and **RFORCE**, either.
4. If **LIDi** references an enforced displacement load, defined with **SPCD** Bulk Data, the same **LID** may not be used for any other load data.
5. The same load set may not be referenced by both **LOADCYN** and **LOADCYH** data.

7 BULK

Bulk Data Entry MAT1

Defines the material properties for linear, temperature-independent, isotropic materials.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT1	MID	E	G	ν	ρ	α	T0	GE	-cont-
-cont-	ST	SC	SS						

MAT1	101	3.1+7	4.2+6		0.002	1.2-6	537.	0.32	+A
+A	25.+4	20.+4	1.2+5						

Field	*Contents*	*Data Range*	*Default*
MID	Material identification number. [1]	Integer>0	Required
E	Young's modulus.	Real>0.0	[2]
G	Shear modulus.	Real>0.0	[2]
ν	Poisson's ratio.	-1.0<Real<0.5	[2]
ρ	Mass density. [3,4]	Real>0.0	**0.0**
α	Thermal expansion coefficient.	Real>0.0	**0.0**
T0	Thermal expansion reference temperature.	Real	**0.0**
GE	Structural element damping coefficient. [5]	Real	**0.0**
ST,SC, SS	Stress/strain limits for tension, compression and shear.	Real	[6]

Remarks:

1. **MID** must be unique for all **MAT1**, **MAT2**, **MAT3**, **MAT8** and **MAT9** entries.

7 BULK

2. The table below indicates the relationship between the **E**, **G** and ν fields and the available options:

IF THE FOLLOWING VALUES ARE INPUT FOR:			THEN UAI/*NASTRAN* WILL USE THE VALUES		
E	**G**	ν	**E**	**G**	ν
E-val	**G**-val	ν-val	**E**-val	**G**-val	ν-val
		blank	**E**-val	**G**-val	$\nu = \frac{E}{2G} - 1$
	blank	ν-val	**E**-val	$G = \frac{E}{2(1+\nu)}$	ν-val
		blank	**E**-val	$G = \frac{E}{2}$	0.0
blank	**G**-val	ν-val	$E = 2G(1+\nu)$	**G**-val	ν-val
		blank	$E = 2G$	**G**-val	0.0
	blank	ν-val	Illegal, one of **E** or **G** must be entered.		
		blank			

3. The mass density, ρ, will be used to automatically compute mass for all structural elements.

4. If ρ is in weight units, the Bulk Data entry:

   PARAM,WTMASS,*conversion_factor*

   must be used to convert the weight data to mass units for dynamics analyses. For example, if ρ is entered as pounds per cubic inch, then the *conversion_factor* is 1./386. or .00259, the reciprocal of the acceleration of gravity in (in/sec/sec).

5. An element damping matrix (K4ee, the element partition of matrix K4GG) is computed as the element stiffness matrix times the damping factor (Kee * GE).

6. Stress/strain limits are required when composite materials are being used and failure index calculations are requested, and they are required for computing margins of safety with respect to either stress or strain allowables. The presence of these values acts as the request for margin of safety calculations for BAR, BEAM, and PILE elements.

7. **MAT1** materials may be made temperature dependent by use of the **MATT1** data.

Bulk Data Entry MAT1NL

Defines the nonlinear isotropic material properties for BAR, ROD, PILE, TRIA3, TRIAR, QUAD4, QUADR, TETRA, PENTA, andHEXA elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT1NL	MID	TNLID	TYPE	YC					
MAT1NL	2	101							

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1]	Integer>0	Required
TNLID	TABLENL identification number which defines the material stress as a function of strain.	Integer>0	Required
TYPE	Nonlinear material type. [2]	Character { **PLASTIC** **NLELAST** **EPCOMB** }	**PLASTIC**
YC	Yield criterion. [3]	Character { **VONM** **TRESCA** **MC** **DP** }	**VONM**

Remarks:

1. A **MAT1** Bulk Data entry with the same **MID** must be present to define any required properties other than Young's modulus. The Young's modulus on the **MAT1** entry must be equal to the initial slope on the **TABLENL** entry with identification number **TNLID**.
2. The three nonlinear material behavior types are:
   - **NLELAST** - nonlinear elastic
   - **PLASTIC** - elasto-plastic
   - **EPCOMB** - combined nonlinear elastic and plastic
3. The yield criterion is not used if the nonlinear material behavior is of type **NLELAST**. Otherwise, the four types of yield criteria which may be used are:
   - **VONM** - Hencky-von Mises
   - **TRESCA** - Tresca
   - **MC** - Mohr-Coulomb
   - **DP** - Drucker-Prager
4. These data apply only to BAR, ROD, BEAM, PILE, TRIA3, TRIAR, QUAD4, QUADR, TETRA, PENTA, and HEXA elements when using **APPROACH NONLINEAR**. Second-order elements are not supported with the exception of the TETRA.

7 BULK

Bulk Data Entry MAT1NL1

Defines the bilinear isotropic material properties for BAR, ROD, PILE, TRIA3, TRIAR, QUAD4, QUADR, TETRA, PENTA, andHEXA elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT1NL1	MID	TYPE	YC	YIELD	ET	HYSTYP	ϕ	β	
MAT1NL1	4			5.0+4	1.0+7	KINE			

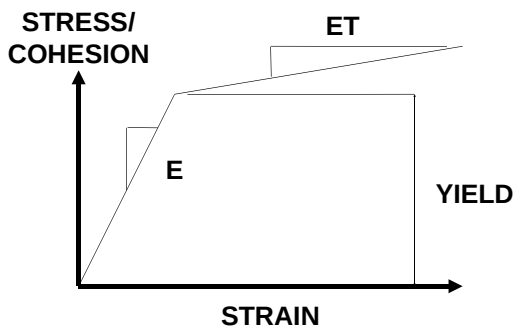
Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1,2]	Integer>0	Required
TYPE	Nonlinear material type. [3,4]	Character { **PLASTIC** **NLELAST** }	**PLASTIC**
YC	Yield criterion. [4,5]	Character { **VONM** **TRESCA** **MC** **DP** }	**VONM**
YIELD	Yield stress or cohesion. [2,6]	Real	Required
ET	The second slope of the bilinear stress-strain curve. [2]	Real	**0.0**
HYSTYP	Hysteretic hardening rule used with **PLASTIC** materials. [4,7]	Character { **ISOT** **KINE** **COMB** }	**ISOT**
f	Angle of internal friction in degrees. [4,8]	0.0≤Real≤45.0	**0.0**
b	Combination factor. [4,9]	0.0≤Real≤1.0	**0.5**

Remarks:

1. A **MAT1** Bulk Data entry with the same **MID** must be present to define any required properties including the linear value for Young's modulus.

2. These data apply only to BAR, ROD, BEAM, PILE, TRIA3, TRIAR, QUAD4, QUADR, TETRA, PENTA, and HEXA elements when using **APPROACH NONLINEAR**. Second-order elements are not supported with the exception of the TETRA.

   The data define a bilinear stress-strain curve as illustrated in the figure.



7 BULK

3. One of two nonlinear material behavior types may be selected:
   - **NLELAST** - nonlinear elastic
   - **PLASTIC** - elasto-plastic
4. If **TYPE = NLELAST, YC, HYSTYP,** ϕ and β are not used.
5. A yield criterion may be selected from one of four options:
   - **VONM** - Hencky-von Mises
   - **TRESCA** - Tresca
   - **MC** - Mohr-Coulomb
   - **DP** - Drucker-Prager
6. **YIELD** denotes yield stress if **YC** is **VONM** or **TRESCA**. **YIELD** denotes cohesion in stress units if **YC** is **MC** or **DP**.
7. A hardening rule may be selected from one of three options:
   - **ISOT** - Isotropic hardening
   - **KINE** - Kinematic hardening
   - **COMB** - Combined hardening
8. The angle of internal friction is used in conjunction with the cohesion value used for yield criteria **MC** or **DP.**
9. **b** is used with the **COMB** hardening rule and controls the level of combination. A value of **0.0** will result in **KINE**matic hardening, while **1.0** represents **ISOT**ropic hardening.

Bulk Data Entry MAT2

Defines the material properties for linear, temperature-independent, anisotropic materials for plate and curved shell elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT2	MID	G11	G12	G13	G22	G23	G33	ρ	-cont-
-cont-	α1	α2	α12	T0	GE	ST	SC	SS	

MAT2	205	7.6+3			7.6+3		4.3+3	0.07	+A
+A	4.8-6	4.8-6		100.	.24				

Field	*Contents*	*Data Range*	*Default*
MID	Material identification number. [1]	Integer>0	Required
Gij	The elements of the 3x3 symmetric material property matrix. [2]	Real	**0.0**
ρ	Mass density. [3]	Real	**0.0**
α_{ij}	Thermal expansion coefficient vector. [4]	Real	**0.0**
T0	Thermal expansion reference temperature. [4]	Real	**0.0**
GE	Structural element damping coefficient. [5]	Real	**0.0**
ST, SC, SS	Stress / strain limits for tension, compression, and shear.	Real>0.0	[6]

Remarks:

1. **MID** must be unique for all **MAT1**, **MAT2**, **MAT3**, **MAT8** and **MAT9** entries.
2. The stress-strain relationship defined by these data is:

$$\begin{Bmatrix}\sigma_x\\ \sigma_y\\ \tau_{xy}\end{Bmatrix} = \begin{bmatrix}\mathbf{G11} & \mathbf{G12} & \mathbf{G13}\\ \mathbf{G12} & \mathbf{G22} & \mathbf{G23}\\ \mathbf{G13} & \mathbf{G23} & \mathbf{G33}\end{bmatrix} \left[\begin{Bmatrix}\varepsilon_x\\ \varepsilon_y\\ \gamma_{xy}\end{Bmatrix} - (T - \mathbf{T0}) \begin{Bmatrix}\alpha_1\\ \alpha_2\\ \alpha_{12}\end{Bmatrix} \right]$$

3. The mass density, ρ, will be used to automatically compute mass for all structural elements.
4. **MAT2** materials may be made temperature dependent by use of **MATT2** data.
5. An element damping matrix ($\boldsymbol{K4}_{ee}$, the element partition of matrix $\boldsymbol{K4}_{gg}$) is computed as the element stiffness matrix times the damping factor, $G_e\,\boldsymbol{K4}_{ee}$.
6. Stress/strain limits are required for composite materials if failure index calculations are requested.

7 BULK

Bulk Data Entry MAT3

Defines the material properties for linear, temperature-independent, orthotropic materials referenced by the axisymmetric elements TORDRG, TRAPRG, TRIARG, TRIAAX and TRAPAX.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT3	MID	Er	Eθ	Ez	νrθ	νθz	νzx	ρ	-cont-
-cont-			Gzr	αr	αθ	αz	T0	GE	

MAT3	103	1.0+7	3.3+7	12.+4	0.3	.24	0.2	1.7-3	+A
+A			1.9+6	1.1-5	1.6-5	1.7-5	687.2	0.29	

Field	*Contents*	*Data Range*	*Default*
MID	Material identification number. [1]	Integer>0	Required
Ei	Young's moduli in the ith direction. [2]	Real>0.0	Required
nij	Poisson's ratios. (Coupled strain ratios in the rθ, θz and zr directions respectively) [2]	-1.0<Real<1.0	Required
ρ	Mass density. [3]	Real	**0.0**
Gzr	Shear modulus. [2]	Real>0.0	Required
ai	Thermal expansion coefficient. [2]	Real	**0.0**
T0	Thermal expansion reference temperature. [2]	Real	**0.0**
GE	Structural element damping coefficient. [4]	Real	**0.0**

Remarks:

1. **MID** must be unique for all **MAT1**, **MAT2**, **MAT3**, **MAT8** and **MAT9** entries.
2. The stress-strain relationship defined by these data, in the axisymmetric coordinate system is:

$$\begin{Bmatrix} \varepsilon_r \\ \varepsilon_\theta \\ \varepsilon_z \\ \gamma_{zr} \end{Bmatrix} = \begin{bmatrix} \frac{1}{Er} & -\frac{\nu r\theta}{E\theta} & -\frac{\nu zr}{Ez} & 0 \\ -\frac{\nu r\theta}{Er} & \frac{1}{E\theta} & -\frac{\nu\theta z}{Ez} & 0 \\ -\frac{\nu zr}{Er} & -\frac{\nu\theta z}{E\theta} & \frac{1}{Ez} & 0 \\ 0 & 0 & 0 & \frac{1}{Gzr} \end{bmatrix} \begin{Bmatrix} \sigma_r \\ \sigma_\theta \\ \sigma_z \\ \tau_{zr} \end{Bmatrix} + \begin{bmatrix} (T - \mathbf{T0}) \begin{Bmatrix} \alpha_r \\ \alpha_\theta \\ \alpha_z \\ 0 \end{Bmatrix} \end{bmatrix}$$

3. The mass density, ρ, will be used to automatically compute mass for all structural elements.
4. An element damping matrix ($\boldsymbol{K4}_{ee}$, the element partition of matrix $\boldsymbol{K4}_{gg}$) is computed as the element stiffness matrix times the damping factor, $G_e\,\boldsymbol{K4}_{ee}$.
5. **MAT3** materials may be made temperature-dependent by use of **MATT3** data.

7 BULK

Bulk Data Entry MAT4

Defines the thermal material properties for temperature-independent, isotropic materials.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT4	MID	K	CP						

MAT4	501	2.5	0.4						

Field	*Contents*	*Data Range*	*Default*
MID	Material identification number. [1]	Integer>0	Required
K	Thermal conductivity or convective film coefficient. [2]	Real≥0.0	Required
CP	Thermal capacity per unit volume, or film capacity per unit area. [2]	Real≥0.0	**0.0**

Remarks:

1. **MID** must be unique with respect to other **MAT4** or **MAT5** entries.
2. If an HBDY element references this entry, **K** is the convective film coefficient and **CP** is the thermal capacity per unit area.
3. **MAT4** materials may be made temperature dependent by use of **MATT4** data.

7 BULK

Bulk Data Entry MAT5

Defines the thermal material properties for temperature-independent, anisotropic materials.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT5	MID	KXX	KXY	KXZ	KYY	KYZ	KZZ	CP	

MAT5	102	.061			.072		.017	0.4	

Field	*Contents*	*Data Range*	*Default*
MID	Material identification number. [1]	Integer>0	Required
Kij	Thermal conductivity. [2]	Real≥0.0	**0.0**
CP	Thermal capacity per unit volume.	Real≥0.0	**0.0**

Remarks:

1. **MID** must be unique with respect to other **MAT4** or **MAT5** entries.
2. The thermal conductivity matrix has the form:

$$K = \begin{bmatrix} \mathbf{KXX} & \mathbf{KXY} & \mathbf{KXZ} \\ \mathbf{KXY} & \mathbf{KYY} & \mathbf{KYZ} \\ \mathbf{KXZ} & \mathbf{KYZ} & \mathbf{KZZ} \end{bmatrix}$$

3. **MAT5** materials may be made temperature-dependent by use of **MATT5** data.

7 BULK

Bulk Data Entry MAT8

Defines the material property for an orthotropic material for plate elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT8	MID	E1	E2	ν12	G12	G1Z	G2Z	ρ	-cont-
-cont-	α1	α2	T0	XT	XC	YT	YC	S	-cont-
-cont-	GE	F12							

MAT8	299	32+6	4.2+5	0.33	2.9+6			0.0442	+A
+A	14.-6	2.3-6	175.						+B
+B	2.5-4								

Field	*Contents*	*Data Range*	*Default*
MID	Material identification number.	Integer>0	Required
E1,E2	Modulus of elasticity in the material x- and y-directions.	Real>0.0	Required
ν12	Poisson's Ratio.	Real	**0.0**
G12	In-plane shear modulus.	Real≥0.0	**0.0**
G1Z,G2Z	Transverse shear moduli.	Real≥0.0	**0.0**
ρ	Mass density.	Real	**0.0**
αi	Thermal expansion coefficients in the material x- and y-directions.	Real	**0.0**
T0	Thermal expansion reference temperature.	Real	**0.0**
XT	Allowable tensile stress/strain in the material x-direction.	Real>0.0	[3]
XC	Allowable compressive stress/strain in the material x-direction. [3]	Real>0.0	**XT**
YT	Allowable tensile stress/strain in the material y-direction.	Real>0.0	[3]
YC	Allowable compressive stress/strain in the material y-direction. [3]	Real>0.0	**YT**
S	Allowable stress/strain for in-plane shear.	Real>0.0	[3]
GE	Structural damping coefficient. [4]	Real	**0.0**
F12	Tsai-Wu interaction term.	Real	[5]

7 BULK

Remarks:

1. Material coordinate systems are defined by the plate element connection or property Bulk Data entries.

2. The stress-strain relationship defined by these data is:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \tau_{xy} \end{Bmatrix} = \begin{bmatrix} \frac{\mathbf{E1}}{\mathbf{1}-\nu\mathbf{12}\,\nu\mathbf{21}} & \frac{\nu\mathbf{12}\,\mathbf{E2}}{\mathbf{1}-\nu\mathbf{12}\,\nu\mathbf{21}} & 0 \\ \frac{\nu\mathbf{12}\,\mathbf{E2}}{\mathbf{1}-\nu\mathbf{12}\,\nu\mathbf{21}} & \frac{\mathbf{E2}}{\mathbf{1}-\nu\mathbf{12}\,\nu\mathbf{21}} & 0 \\ 0 & 0 & \mathbf{G12} \end{bmatrix} \left[\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \gamma_{xy} \end{Bmatrix} - (T - \mathbf{T0}) \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ 0 \end{Bmatrix} \right]$$

and

$$\begin{Bmatrix} \tau_{xz} \\ \tau_{yz} \end{Bmatrix} = \begin{bmatrix} \mathbf{G1Z} & 0 \\ 0 & \mathbf{G2Z} \end{bmatrix} \begin{Bmatrix} \gamma_{xz} \\ \gamma_{yz} \end{Bmatrix}$$

where

$$\nu\mathbf{21} = \nu\mathbf{12}\,\frac{\mathbf{E2}}{\mathbf{E1}}$$

3. Fields **XT**, **XC**, **YT**, **YC** and **S** are used only for composite materials when failure calculations are requested with **PCOMP**, **PCOMP1** or **PCOMP2** Bulk Data entries. Allowables represent stresses except when the maximum strain failure theory is used.

4. An element damping matrix ($\boldsymbol{K4}_{ee}$, the element partition of matrix $\boldsymbol{K4}_{gg}$) is computed as the element stiffness matrix times the damping factor, $G_e\,\boldsymbol{K4}_{ee}$.

5. The **F12** field is used only for composite materials when the Tsai-Wu failure theory is used and failure calculations are requested.

Bulk Data Entry MAT9

Defines the material properties for linear, temperature-independent, anisotropic materials for solid isoparametric elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MAT9	MID	G11	G12	G13	G14	G15	G16	G22	-cont-
-cont-	G23	G24	G25	G26	G33	G34	G35	G36	-cont-
-cont-	G44	G45	G46	G55	G56	G66	ρ	α1	-cont-
-cont-	α2	α3	α4	α5	α6	T0	GE		

MAT9	101	7.6+6						7.6+6	
					7.6+6				
	3.4+4			3.4+4		3.4+4			

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1]	Integer>0	Required
Gij	Elements of the 6x6 symmetric material property matrix. [2]	Real	**0.0**
ρ	Mass density. [3]	Real	**0.0**
αi	Thermal expansion coefficient vector. [2]	Real	**0.0**
T0	Thermal expansion reference temperature. [2]	Real	**0.0**
GE	Structural element damping coefficient. [4]	Real	**0.0**

Remarks:

1. **MID** must be unique for all **MAT1**, **MAT2**, **MAT3**, **MAT8** and **MAT9** entries.
2. The stress-strain relationship defined by these data is:

$$\begin{Bmatrix} \sigma_x \\ \sigma_y \\ \sigma_z \\ \tau_{xy} \\ \tau_{yz} \\ \tau_{zx} \end{Bmatrix} = \begin{bmatrix} \mathbf{G11} & & & & & \\ \mathbf{G12} & \mathbf{G22} & & \mathbf{Sym} & & \\ \mathbf{G13} & \mathbf{G23} & \mathbf{G33} & & & \\ \mathbf{G14} & \mathbf{G24} & \mathbf{G34} & \mathbf{G44} & & \\ \mathbf{G15} & \mathbf{G25} & \mathbf{G35} & \mathbf{G45} & \mathbf{G55} & \\ \mathbf{G16} & \mathbf{G26} & \mathbf{G36} & \mathbf{G46} & \mathbf{G56} & \mathbf{G66} \end{bmatrix} \left[\begin{Bmatrix} \varepsilon_x \\ \varepsilon_y \\ \varepsilon_z \\ \gamma_{xy} \\ \gamma_{yz} \\ \gamma_{zx} \end{Bmatrix} - (T - \mathbf{T0}) \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \alpha_4 \\ \alpha_5 \\ \alpha_6 \end{Bmatrix} \right]$$

3. The mass density ρ will be used to automatically compute mass for all structural elements.
4. An element damping matrix ($\boldsymbol{K4}_{ee}$, the element partition of matrix $\boldsymbol{K4}_{gg}$) is computed as the element stiffness matrix times the damping factor, $G_e\,\boldsymbol{K4}_{ee}$.
5. **MAT9** materials may be made temperature-dependent by use of a **MATT9** entry.

7 BULK

Bulk Data Entry MATF

Defines fluid density and compressibility properties for fluid materials.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MATF	MID	ρ	β						
MATF	2	.6	1.E+4						

Field	*Contents*	*Data Range*	*Default*
MID	Material identification number. [1]	Integer>0	Required
ρ	Mass density.	Real>0.0	Required
β	Bulk modulus.	Real≥0.0	**0.0**

Remarks:

1. **MID** must be unique with respect to all **MATF**, **MAT1** and **MAT9** entries.
2. If β is blank or 0.0, the fluid is assumed incompressible. If any fluid element is incompressible, the entire fluid is assumed incompressible.

7 BULK

Bulk Data Entry MATT1

Defines table references for isotropic material properties which are temperature-dependent.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MATT1	MID	TID1	TID2	TID3	TID4	TID5		TID6	-cont-
-cont-	TID7	TID8	TID9						

MATT1	118	133							+A
+A		163							

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1,2]	Integer>0	Required
TIDi	Table identification numbers.	Integer>0	[2,3]

Remarks:

1. Material temperature dependency information are used only when a temperature distribution for materials is defined by using **TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control commands.
2. The **TIDi** correspond to the entries defined on the **MAT1** Bulk Data entry with the same **MID**, and they are positional. A blank or zero entry means that the quantity is not temperature-dependent.
3. **TIDi** refer to table data defined by **TABLEM1**, **TABLEM2**, **TABLEM3** or **TABLEM4** entries.
4. The material property computation depends on the **TABLEMi** entry selected.

7 BULK

Bulk Data Entry MATT2

Defines table references for anisotropic material properties which are temperature-dependent.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MATT2	MID	TID1	TID2	TID3	TID4	TID5	TID6	TID7	-cont-
-cont-	TID8	TID9	TID10		TID11	TID12	TID13	TID14	

MATT2	901		1001					125	+A
+A	163	223							

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1,2]	Integer>0	Required
TIDi	Table identification numbers.	Integer>0	[2,3]

Remarks:

1. Material temperature dependency information are used only when a temperature distribution for materials is defined by using **TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control commands.
2. The **TIDi** correspond to the entries defined on the **MAT2** Bulk Data entry with the same **MID**, and they are positional. A blank or zero entry means that the quantity is not temperature-dependent.
3. **TIDi** refer to table data defined by **TABLEM1**, **TABLEM2**, **TABLEM3** or **TABLEM4** entries.
4. The material property computation depends on the **TABLEMi** entry selected.

7 BULK

Bulk Data Entry MATT3

Defines table references for orthotropic material properties which are temperature-dependent.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MATT3	MID	TID1	TID2	TID3	TID4	TID5	TID6	TID7	-cont-
-cont-			TID10	TID11	TID12	TID13		TID14	

MATT3	501	48			54				+A
+A	74								

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1,2]	Integer>0	Required
TIDi	Table identification numbers.	Integer>0	[2,3]

Remarks:

1. Material temperature dependency information are used only when a temperature distribution for materials is defined by using **TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control commands.
2. The **TIDi** correspond to the entries defined on the **MAT3** Bulk Data entry with the same **MID**, and they are positional. A blank or zero entry means that the quantity is not temperature-dependent.
3. **TIDi** refer to table data defined by **TABLEM1**, **TABLEM2**, **TABLEM3** or **TABLEM4** entries.
4. The material property computation depends on the **TABLEMi** entry selected.

7 BULK

Bulk Data Entry MATT4

Defines table references for temperature dependent thermal conductivity or convective film coefficients.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MATT4	MID	TID							

MATT4	101	554							

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1]	Integer>0	Required
TID	Table identification number. [2]	Integer>0	Required

Remarks:

1. Material temperature dependency information are used only when a temperature distribution for materials is defined by using **TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control commands.
2. **TID** refers to table data defined by **TABLEM1**, **TABLEM2**, **TABLEM3** or **TABLEM4** entries.
3. The material property computation depends on the **TABLEMi** entry selected.

7 BULK

Bulk Data Entry MATT5

Defines table references for a temperature dependent conductivity matrix.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MATT5	MID	TID1	TID2	TID3	TID4	TID5	TID6		
MATT5	50	121		345					

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1,2]	Integer>0	Required
TIDi	Table identification numbers.	Integer>0	[2,3]

Remarks:

1. Material temperature dependency information are used only when a temperature distribution for materials is defined by using **TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control commands.
2. The **TIDi** correspond to entries defined on the **MAT5** Bulk Data entry with the same **MID**, and they are positional. A blank or zero means the quantity is not temperature dependent.
3. **TIDi** refer to table data defined by **TABLEM1**, **TABLEM2**, **TABLEM3** or **TABLEM4** entries.
4. The material property computation depends on the **TABLEMi** entry selected.

Bulk Data Entry MATT9

Defines table references to properties for linear, temperature-dependent, anisotropic materials for solid isoparametric elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MATT9	MID	TID11	TID12	TID13	TID14	TID15	TID16	TID22	-cont-
-cont-	TID23	TID24	TID25	TID26	TID33	TID34	TID35	TID36	-cont-
-cont-	TID44	TID45	TID46	TID55	TID56	TID66	TID1	TID2	-cont-
-cont-	TID3	TID4	TID5	TID6	TID7		TID8		

MATT9	101	102						102	+A
+A					102				+B
+B	102			102		102			

Field	*Contents*	*Data Range*	*Default*
MID	Material property identification number. [1,2]	Integer>0	Required
TIDij	Table identification numbers.	Integer>0	[2,3]

Remarks:

1. Material temperature dependency information are used only when a temperature distribution for materials is defined by using **TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control commands.
2. The **TIDij** correspond to entries defined on the **MAT9** Bulk Data entry with the same **MID**, and they are positional. A blank or zero means that the quantity is not temperature dependent.
3. **TIDij** refer to table data defined by **TABLEM1**, **TABLEM2**, **TABLEM3** or **TABLEM4** entries.
4. The material property computation depends on the **TABLEMi** entry selected.

7 BULK

Bulk Data Entry MOMAX

Defines a static moment, $\vec{m}$, of the form:

$$\vec{m} = \mathbf{M} \cdot \vec{\mathbf{V}}$$

for use in an axisymmetric harmonic model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MOMAX	LID	RID	HID	M	V1	V2	V3		
MOMAX	102	2	3	2.5	1.0	0.0	0.0		

Field	*Contents*	*Data Range*	*Default*
SID	Load set identification number. [1]	Integer>0	Required
RID	Identification number of ring to be loaded.	Integer>0	Required
HID	Harmonic identification number. [2]	Integer>0 or Harmonic Code	Required
M	Moment scale factor.	Real	Required
Vi	Components of moment vector in the cylindrical system.	Real	[3]

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. A separate entry is needed to define the moment associated with each harmonic.
3. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.
4. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

7 BULK

Bulk Data Entry MOMENT

Defines a static moment at a GRID point of the form:

$$\vec{m} = \mathbf{M} \cdot \vec{\mathbf{V}}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
MOMENT	LID	GID	CID	M	V1	V2	V3		
MOMENT	102	527	205	2.9	0.0	0.0	1.0		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
GID	Identification number of GRID point to be loaded.	Integer>0	Required
CID	Coordinate system identification number in which the resulting vector is applied. [2]	Integer≥0	**0**
M	Moment scale factor.	Real	Required
Vi	Components of the moment vector measured in the coordinate system defined by **CID**.	Real	[3]

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. When **CID** is not defined, the resulting vector is applied in the Basic Coordinate System.

3. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.

7 BULK

Bulk Data Entry MOMENT1

Defines a static moment at a GRID point of the form:

$$\vec{m} = \mathbf{M} \cdot \frac{\mathbf{GID1}\vec{\ }\mathbf{GID2}}{|\ \mathbf{GID1}\vec{\ }\mathbf{GID2}\ |}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
MOMENT1	LID	GID	M	GID1	GID2				
MOMENT1	106	113	8.76	116	100				

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
GID	Identification number of GRID point to be loaded.	Integer>0	Required
M	Magnitude of moment.	Real	Required
GIDi	GRID point identification numbers. [2]	Integer>0 **GID1≠GID2**	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. The direction of the resulting vector is determined by the normalized vector from **GID1** to **GID2**.

7 BULK

Rev: V20.1

Bulk Data Entry MOMENT2

Defines a static moment at a GRID point of the form:

$$\vec{m} = \mathbf{M} \cdot \frac{\mathbf{GID1}\vec{}\,\mathbf{GID2} \times \mathbf{GID3}\vec{}\,\mathbf{GID4}}{|\ \mathbf{GID1}\vec{}\,\mathbf{GID2} \times \mathbf{GID3}\vec{}\,\mathbf{GID4}\ |}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
MOMENT2	LID	GID	M	GID1	GID2	GID3	GID4		
MOMENT2	116	1010	76.3	100	101	201	200		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
GID	Identification number of GRID point to be loaded.	Integer>0	Required
M	Magnitude of moment.	Real	Required
GIDi	GRID point identification numbers. [2]	Integer>0 **GID1≠GID2** **GID3≠GID4**	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. The direction of the resulting vector is determined by the normalized cross product of two vectors, defined from **GID1** to **GID2**, and from **GID3** to **GID4**.

7 BULK

Rev: V20.1

Bulk Data Entry MPC

Defines a multipoint constraint equation of the form:

$$u_d = -\frac{1}{\mathbf{B}} \sum_i \mathbf{Ai}\, u_i$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
MPC	SID	GIDD	DOFD	B	GID1	DOF1	A1		-cont-
-cont-		GID2	DOF2	A2	*CONTINUES IN GROUPS OF 3*				-cont-

MPC	104	167	3	6.8	254		9.2		+A
+A		763	4	-2.					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
GIDD	Identification number of dependent GRID or SCALAR point.	Integer>0	Required
DOFD	Single dependent degree of freedom.	DOF Code	Required
B	Coefficient of dependent degree of freedom.	Real≠0.0	Required
GIDi	Identification number of independent GRID or SCALAR point.	Integer>0	Required
DOFi	Single independent degree of freedom.	DOF Code	Required
Ai	Coefficient of independent degree of freedom.	Real≠0.0	Required

Remarks:

1. Multipoint constraint sets must be selected in the Case Control packet with the command:

   MPC = SID

2. The component **DOFD** specified is placed in the m-set. Therefore, it must not appear on Bulk Data entries such as **ASET, OMIT, SPC,** or **SUPORT**, which would place it in a mutually exclusive set.

7 BULK

Bulk Data Entry MPCADD

Defines a multipoint constraint set as the union of multipoint constraint sets defined with **MPC** Bulk Data.

Format and Example:

1	2	3	4	5	6	7	8	9	10
MPCADD	SID	S1	S2	S3	S4	S5	S6	S7	-cont-
-cont-	S8	S9	*CONTINUES WITH LIST OF VALUES*						-cont-

MPCADD	202	103	104	102	107	105	501	527	+A
+A	811	901	1001						

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
Si	Set identification numbers of multipoint constraints sets defined with **MPC** data. [2,3]	Integer>0	Required

Remarks:

1. Multipoint constraint sets must be selected in the Case Control packet with the command:

   MPC = SID

2. The **Si** must be unique.
3. **Si** may not be the identification number of a multipoint constraint set defined by another **MPCADD** entry.

7 BULK

Bulk Data Entry MPCAX

Defines a multipoint constraint equation for an axisymmetric harmonic model of the form:

$$u_d = -\frac{1}{\mathbf{B}} \sum_i \mathbf{Ai}\, u_i$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
MPCAX	SID				RIDD	HIDD	DOFD	B	-cont-
-cont-	RID1	HID1	DOF1	A1	*CONTINUES IN GROUPS OF 4*				-cont-

MPCAX	109				118	5	4	1.0	+A
+A	723	4	5	-2.3					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
RIDD	Ring identification number for dependent degree of freedom.	Integer>0	Required
HIDD	Harmonic identification number for dependent degree of freedom.	Integer≥0	**0**
DOFD	Single dependent degree of freedom.	DOF Code	Required
B	Coefficient for dependent degree of freedom.	Real≠0	Required
RIDi	Ring identification number.	Integer>0	Required
HIDi	Harmonic identification number.	Integer≥0	**0**
DOFi	Single independent degree of freedom.	DOF Code	Required
Ai	Coefficient of independent degree of freedom.	Real	Required

Remarks:

1. Multipoint constraint sets must be selected in the Case Control packet with the command:

   MPC = SID

2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

3. The component **DOFD** specified is placed in the m-set. Therefore, it must not appear on Bulk Data entries such as **ASET, OMIT, SPC,** or **SUPORT**, which would place it in a mutually exclusive set.

7 BULK

Bulk Data Entry MPCS

Defines multipoint constraints within or between substructures of the form:

$$u_d = -\frac{1}{\mathbf{B}} \sum_i \sum_j \mathbf{Aij}\, u_j$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
MPCS	SID	SNAMED	GIDD	DOFD	B				-cont-
-cont-		SNAME1	GID11	DOF11	A11	GID12	DOF12	A12	-cont-
-cont-		SNAME2	GID21	DOF21	A21	GID22	DOF22	A22	-cont-

MPCS	121	WINGRT	504	3	1.5				+A
+A		FUSLG	1036	3	0.25	1036	5	23.1	+B
+B		CABIN	39	4	.05				

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1,2]	Integer>0	Required
SNAMED	Basic Substructure name containing the dependent degree of freedom.	Name	Required
GIDD	Identification number of dependent GRID or SCALAR point in substructure **SNAMED**.	Integer>0	Required
DOFD	Single dependent degree of freedom.	DOF Code	Required
B	Coefficient of dependent degree of freedom.	Real≠0.0	Required
SNAMEi	Basic Substructure names containing the independent degrees of freedom.	Name	Required
GIDij	GRID or SCALAR point identification number in Basic Substructure **SNAMEi**.	Integer>0	Required
DOFij	Single degree of freedom of **GIDij** in **SNAMEi**.	DOF Code	Required
Aij	Coefficient of independent degree of freedom.	Real	**0.0**

Remarks:

1. Multipoint constraint sets must be selected in the Case Control packet with the command:

   MPC = SID

2. **MPCS** entries may be referenced by an **MPCADD** entry.
3. **MPCS** constraints may be imposed only in PHASE 2 Substructuring analyses during the **SOLVE** step. The referenced GRID point components must exist in the final solution substructure.
4. The component **DOFD** specified is placed in the m-set. Therefore, it must not appear on Bulk Data entries such as **ASET, OMIT, SPC,** or **SUPORT**, which would place it in a mutually exclusive set.

Bulk Data Entry NLSOLVE

Selects the algorithm and defines control data for nonlinear analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
NLSOLVE	NLSID	SMETH	IMETH	PMETH	CONV	TENDIV	UMAX	MINK	-cont-
-cont-	MAXK	MAXP	PINC	MAXPINC	UDIV	LAMDA	EPMAX	ETAS	-cont-
-cont-	EPSE	EPSP	EPSU	ROTMAX	STNMAX	FACMAX	MINP		

NLSOLVE	20	TAN	SEMI	AUTO	E				+A
+A	50	5			1.5	1.5	50.		+B
+B	.001								

Field	*Contents*	*Data Range*	*Default*
NLSID	Nonlinear solution control identification number. [1]	Integer>0	Required
SMETH	Solution method. [2]	Character { **NR** **SEC** **ARC** **ARCFIX** **DISP** **LOAD** }	[2]
IMETH	Unbalanced force iteration method. [3]	Character { **AUTO** **FIX** **SEMI** }	AUTO
PMETH	External load or **ARC** length incrementation method. [4]	Character { **AUTO** **FIX** **SEMI** }	AUTO
CONV	Overall convergence criteria. [5]	Character { **E** **P** **U** }	EPU
TENDIV	Tentatively divergent processing parameter. [6]	Integer	**1**
UMAX	Maximum absolute value of deflection which denotes divergence or solution termination.	Real>0.0	[8]
MINK	Minimum number of stiffness updates before the selected **IMETH** starts at a new load increment. [12]	Integer≥0	**0**
MAXK	Maximum number of stiffness updates for any nonlinear **STEP**.	Integer>0	**99**
MAXP	Maximum number of unbalanced force iterations to be performed before a stiffness update.	Integer>0	**30**
PINC	The number of load increments if **PMETH=FIX**. If **PMETH=AUTO** or **SEMI**, the first load increment will be determined by the total load divided by **PINC**.	Integer>0	**10**

7 BULK

Rev: V20.1

MAXPINC	Maximum number of load increments. Only used when **PMETH=AUTO** or **SEMI**.	Integer>0	**99**
UDIV	Displacement norm change rate which indicates the solution is tentatively divergent.	Real>0.0	**0.9999**
LAMDA	Solution convergence rate signifying that the solution is tentatively divergent.	Real>0.0	**0.9999**
EPMAX	Load error above which the load increment size will be reduced. Only used with **PMETH=AUTO**.	Real>0.0	[7]
ETAS	Percentage of the yield to be processed in one sub-increment.	Real>0.0	**0.25**
EPSE	Strain energy convergence criterion tolerance value.	Real>0.0	[9]
EPSP	Load error convergence criterion tolerance value.	Real>0.0	[9]
EPSU	Displacement error convergence criterion tolerance value.	Real>0.0	[9]
ROTMAX	Maximum rotation, in degrees, allowed per load increment. [10]	Real>0.0	**7.5**
STNMAX	Maximum strain allowed per load increment. [11]	Real>0.0	**0.03**
FACMAX	Maximum absolute value of load factor which denotes solution divergence.	Real>0.0	**10.0**
MINP	Minimum number of unbalanced force iterations before the selected **IMETH** starts at a new load increment. Used only when **IMETH** is **AUTO** or **SEMI**. [12]	Integer≥2	[11]

Remarks:

1. **NLSOLVE** Bulk Data is requested using the Case Control command:

   ```
   NLSOLVE = NLSID
   ```

   This Case Control command may be used anywhere from above the **SUBCASE** level to below the **STEP** level.

2. The available solution methods are the Newton-Raphson method (the tangential modulus method), **NR**, the secant modulus method, **SEC**, the automatic arc length method, **ARC**, the fixed arc length method, **ARCFIX**, the displacement method, **DISP**, and the load control method, **LOAD**. **ARCFIX**, **DISP** and **LOAD** are variations of the arc length method. **ARC** uses **ARCFIX** as the basic solution method, and then switches to **DISP** or **LOAD** if the other methods will give a better solution, or if there are convergence problems. If there is any geometric nonlinear behavior, then the default is **ARC**.

3. The available iteration methods are **AUTO** which is the most efficient combination of unbalanced force iterations and stiffness updates, **FIX** where the iteration is controlled manually by other **NLSOLVE** data, and **SEMI** which is the same as **AUTO** except an additional stiffness update is performed at the completion of the **STEP** to better prepare for a subsequent **STEP**.

4. The available load incrementation methods are **AUTO**, where the load increment size will vary automatically depending on the solution characteristics and on the rate of convergence, **FIX**, where the load increments will be of a fixed size, and **SEMI**, where the load increment will vary depending on the rate of convergence only.

5. The available convergence criteria are **E** (strain energy), **P** (load), and **U** (displacement). Any or all of these criteria may be selected by concatenating the symbols. For example, to select them all you enter **EPU**.
6. **TENDIV** selects the procedure to follow when the solution is tentatively divergent.
   - A non-negative integer selects the tentative divergence processing and indicates the number of stiffness updates allowed with the current material state.
   - A value of **-1** suppresses tentatively divergence processing.
   - A value of **-2** causes a stiffness update at each automatic load reduction during tentative divergence processing.
7. If there are GAP elements in the model, the default value for **EPMAX** is 10^6. Otherwise the default value is determined by the iteration status.
8. The default value for **UMAX** is 10^5 if there is no critical point. If critical points are found during the solution, the default value is set to 10 x $U_{max,c}$, where $U_{max,c}$ is the maximum absolute displacement at the first critical point.
9. The default values **EPSE, EPSP and EPSU** are **0.01, 1.0 and 0.1**, respectively if the solution method is **NR** or **LOAD**. In all other cases, they are **0.001, 1.0 and 0.01**.
10. **ROTMAX** is used only if you have requested geometric nonlinear analysis with the **NLTYPE=GEOM** Case Control command.
11. **STNMAX** is used only if you have requested geometric nonlinear analysis with the option **STRAIN=STRETCH**.
12. At each new load increment, first **MINK** stiffness updates are performed, then **MINP** unbalanced force iteration are performed. Then the selected **IMETH** is used. The default value for **MINP** is determined automatically.

Bulk Data Entry NOLIN1

Defines a nonlinear transient forcing function of the form:

$$P_i(t) = \mathbf{S} \cdot T(r_j(t))$$

where r_j may be either displacement or velocity response.

Format and Example:

1	2	3	4	5	6	7	8	9	10
NOLIN1	NLLID	GIDI	DOFI	S	GIDJ	DOFJ	TID	RESPJ	
NOLIN1	21	3	2	2.1	3	2	6	DISP	

Field	*Contents*	*Data Range*	*Default*
NLLID	Nonlinear load set identification number. [1]	Integer>0	Required
GIDI	GRID, SCALAR or EXTRA point identification number at which the nonlinear load is to be applied.	Integer>0	Required
DOFI	Single degree of freedom for **GIDI**. [2]	DOF Code	Required
S	Scale factor.	Real≠0.0	Required
GIDJ	GRID, SCALAR or EXTRA point identification number whose response helps determine the value of the forcing function.	Integer>0	Required
DOFJ	Single degree of freedom for **GIDJ**. [2]	DOF Code	Required
TID	Identification number of a **TABLEDi** entry.	Integer>0	Required
RESPJ	Type of response at degree of freedom **j**.	Character { **DISP** | **VELO** }	**DISP**

Remarks:

1. Nonlinear loads must be selected in the Case Control packet with the command:

   NONLINEAR = NLLID

   Nonlinear loads ***may not*** be referenced by a **DLOAD** Bulk Data entry.

2. All degrees of freedom referenced on **NOLIN1** entries must be members of the solution set.

7 BULK

Bulk Data Entry NOLIN2

Defines a nonlinear transient forcing function of the form:

$$P_i(t) = \mathbf{S} \cdot r_j(t) \ \mathbf{.op.} \ r_k(t)$$

where r_j and r_k may be either displacement or velocity responses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
NOLIN2	NLLID	GIDI	DOFI	S	GIDJ	DOFJ	RESPJ	OP	-cont-
-cont-	GIDK	DOFK	RESPK						

NOLIN2	214	2	1	2.9	2	1	VELO	-	
	3	1	VELO						

Field	*Contents*	*Data Range*	*Default*
NLLID	Nonlinear load set identification number. [1]	Integer>0	Required
GIDI	GRID, SCALAR or EXTRA point identification number at which the nonlinear load is to be applied.	Integer>0	Required
DOFI	Single degree of freedom for **GIDI**. [2]	DOF Code	Required
S	Scale factor.	Real≠0.0	Required
GIDJ	GRID, SCALAR or EXTRA point identification number. [3]	Integer>0	Required
DOFJ	Single degree of freedom for **GIDJ**. [2]	DOF Code	Required
RESPJ	Type of response at degree of freedom **j**. [3]	Character { **DISP** / **VELO** }	**DISP**
OP	Mathematical operator. [3]	Character { **+** / **–** / ***** }	Required
GIDK	GRID, SCALAR or EXTRA point identification number. [3]	Integer>0	Required
DOFK	Single degree of freedom for **GIDK**. [2]	DOF Code	Required
RESPK	Type of response at degree of freedom **k**. [3]	Character { **DISP** / **VELO** }	**DISP**

7 BULK

Remarks:

1. Nonlinear loads must be selected in the Case Control packet with the command:

   NONLINEAR = NLLID

   Nonlinear loads ***may not*** be referenced by a **DLOAD** Bulk Data entry.

2. All degrees of freedom referenced on **NOLIN2** entries must be members of the solution set.

3. The responses, displacements or velocities, at points **j** and **k** are used with the defined mathematical operator to determine the value of the forcing function as shown in the equation.

Bulk Data Entry NOLIN3

Defines nonlinear transient forcing functions of the form:

$$P_i(t) = \begin{cases} \mathbf{S} \cdot r_j(t)^{\mathbf{A}} & \text{for } r_j(t) > 0 \\ 0.0 & \text{for } r_j(t) \le 0 \end{cases}$$

where r_j may be either displacement or velocity response.

Format and Example:

1	2	3	4	5	6	7	8	9	10
NOLIN3	NLLID	GIDI	DOFI	S	GIDJ	DOFJ	RESPJ	A	

NOLIN3	4	102	1	-6.1	2	1	DISP	-3.5	

Field	*Contents*	*Data Range*	*Default*
NLLID	Nonlinear load set identification number. [1]	Integer>0	Required
GIDI	GRID, SCALAR or EXTRA point identification number at which the nonlinear load is to be applied.	Integer>0	Required
DOFI	Single degree of freedom for **GIDI**. [2]	DOF Code	Required
S	Scale factor.	Real≠0.0	Required
GIDJ	GRID, SCALAR or EXTRA point identification number whose response helps determine the value of the forcing function.	Integer>0	Required
DOFJ	Single degree of freedom for **GIDJ**.	DOF Code	Required
RESPJ	Type of response at degree of freedom **j**.	Character { **DISP** } { **VELO** }	**DISP**
A	Amplification factor.	Real	**0.0**

Remarks:

1. Nonlinear loads must be selected in the case Control packet with the command:

   NONLINEAR = NLLID

   Nonlinear loads ***may not*** be referenced by a **DLOAD** Bulk Data entry.

2. All degrees of freedom referenced on **NOLIN3** entries must be members of the solution set.

7 BULK

Bulk Data Entry NOLIN4

Defines a nonlinear transient forcing function of the form:

$$P_i(t) = \begin{cases} -\mathbf{S} \cdot \left(-r_j\,(t)\right)^{\mathbf{A}} & \text{for } r_j(t) < 0 \\ 0.0 & \text{for } r_j(t) \geq 0 \end{cases}$$

where r_j may be either displacement or velocity response.

Format and Example:

1	2	3	4	5	6	7	8	9	10
NOLIN4	NLLID	GIDI	DOFI	S	GIDJ	DOFJ	RESPJ	A	

NOLIN4	2	4	3	2.0	101		16.3		

Field	*Contents*	*Data Range*	*Default*
NLLID	Nonlinear load set identification number. [1]	Integer>0	Required
GIDI	GRID, SCALAR or EXTRA point identification number at which the nonlinear load is to be applied.	Integer>0	Required
DOFI	Single degree of freedom for **GIDI**. [2]	DOF Code	Required
S	Scale factor.	Real≠0.0	Required
GIDJ	GRID, SCALAR or EXTRA point identification number whose response helps determine the value of the forcing function.	Integer>0	Required
DOFJ	Single degree of freedom for **GIDJ**. [2]	DOF Code	Required
RESPJ	Type of response at degree of freedom **j**.	Character { **DISP** } { **VELO** }	**DISP**
A	Amplification factor.	Real	**0.0**

Remarks:

1. Nonlinear loads must be selected in the Case Control packet with the command:

   NONLINEAR = NLLID

   Nonlinear loads ***may not*** be referenced by a **DLOAD** Bulk Data entry.

2. All degrees of freedom referenced on **NOLIN4** entries must be members of the solution set.

7 BULK

Bulk Data Entry OMIT

Defines component degrees of freedom that the user desires to omit, using static condensation, from the analysis set (a-set).

Format and Example:

1	2	3	4	5	6	7	8	9	10
OMIT	GID1	DOF1	GID2	DOF2	GID3	DOF3	GID4	DOF4	
OMIT	16	2	23	3516			1	4	

Field	*Contents*	*Data Range*	*Default*
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom. [1]	DOF Code	Required

Remarks:

1. The component **DOF** specified are placed in the o-set. Therefore, they must not appear on Bulk Data entries such as **ASET**, **MPC**, **SUPORT** or **SPC**, which place them in a mutually exclusive set.
2. When **OMIT** or **OMIT1** entries are present, all degrees of freedom not otherwise constrained will be placed in the a-set.

Bulk Data Entry OMIT1

Defines component degrees of freedom that the user desires to omit, using static condensation, from the analysis set (a-set).

Format and Example:

1	2	3	4	5	6	7	8	9	10
OMIT1	DOF	*GRID ID LIST*							-cont-

OMIT1	345	2	1	3	10	9	6	5	+A
+A	7	8	101	THRU	*301*				

Field	*Contents*	*Data Range*	*Default*
DOF	List of degrees of freedom. [1]	DOF Code	Required
GRID ID LIST	List of one or more GRID or SCALAR point identification numbers.	Integer>0	Required

Remarks:

1. The component **DOF** specified are placed in the o-set. Therefore, they must not appear on Bulk Data entries such as **ASET**, **MPC**, **SUPORT** or **SPC**, which place them in a mutually exclusive set.
2. When **OMIT** or **OMIT1** entries are present, all degrees of freedom not otherwise constrained will be placed in the a-set.

7 BULK

Bulk Data Entry OMITAX

Defines component degrees of freedom that the user desires to omit, using static condensation, from the analysis set (*a-set*) in axisymmetric harmonic analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
OMITAX	RID	HID	DOF						

OMITAX	2	6	3						

Field	*Contents*	*Data Range*	*Default*
RID	RINGAX identification number.	Integer>0	Required
HID	Harmonic identification number.	Integer≥0 or Harmonic Code	Required
DOF	List of degrees of freedom. [1]	DOF Code	Required

Remarks:

1. The component **DOF** specified are placed in the o-set. Therefore, they must not appear on Bulk Data entries that place them in another dependent set such as **MPC**, **SUPORT** or **SPC**.
2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.
3. When **OMITAX** entries are present, all degrees of freedom not otherwise constrained will be placed in the a-set.

Bulk Data Entry PACABS

Defines the properties of an acoustic absorber element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PACABS	PID	OP='YES'	RDEF	XDEF	WDEF	TSTAREA	FCUTOFF		
		OP='NO'						B	-cont-
-cont-	K	M							

PACABS	12		1	2	3	3.5	600.0		

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
OP	Specifies whether the properties will be defined by tabular data (**YES**) or derived from an equivalent structural model (**NO**). [1]	Character { **YES** / **NO** }	**YES**
RDEF	Identification number of a **TABLEDi** entry which defines the resistance. [2]	Integer>0	Required
XDEF	Identification number of a **TABLEDi** entry which defines the reactance. [2]	Integer>0	Required
WDEF	Identification number of a **TABLEDi** entry which defines the weighting function. [2]	Integer>0	**1.0**
TSTAREA	Area of the test specimen.	Real>0.0	**1.0**
FCUTOFF	Cutoff frequency for tables defined by **RDEF**, **XDEF**, and **WDEF**. [2]	Real>=0.0	**500.0**
B,K,M	Equivalent damping, stiffness and mass values.	Real≥0	Required

Remarks:

1. When **OP=YES**, then the fields **RDE**, **XDEF**, **WDEF**, and **FCUTOFF** are used. If **OP=NO**, then the equivalent structural model is derived from the values of **B**, **K**, and **M** entered. Note that the continuation entry is used only in the second case.
2. Data defined in tables must be a function of frequency in Hz.
3. **PACABS** entries, which are referenced only by **CHACAB** entries, are used only in fluid-structure interaction analyses.

7 BULK

Bulk Data Entry PACBAR

Defines the properties of an acoustic barrier element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PACBAR	PID	MBACK	MSEPTM	FRESON	KRESON				
PACBAR	101	1.0	0.01	400.0					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MBACK	Mass per unit area of the backing material. [1]	Real>0.0	Required
MSEPTM	Mass per unit area of the septum material. [1]	Real>0.0	Required
FRESON	Resonant frequency of the sandwich construction (Hz).	Real>0.0	[1]
KRESON	Resonant stiffness per unit area of the sandwich construction. [1]	Real>0.0	[1]

Remarks:

1. Either **FRESON** or **KRESON** must be specified, but not both.
2. **PACBAR** entries, which are referenced only by **CHACBR** entries, are used only in fluid-structure interaction analyses.

7 BULK

Mod: V20.1

Bulk Data Entry PARAM

Defines values for parameters used in DMAP sequences.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PARAM	NAME	VAL1	VAL2						

PARAM	WTMASS	.00259							

Field	*Contents*	*Data Range*	*Default*
NAME	Parameter name. [1]	Name	Required
VAL1, VAL2	Value of the parameter.	Any	[2]

Remarks:

1. The **PARAM** entry may only be used to assign a value to a parameter referenced in the Rigid Format, DMAP ALTER, or user written DMAP program. The parameter must be referenced in the DMAP using the notation **C,Y,NAME** or **V,Y,NAME**.
2. The **VAL1** and **VAL2** fields differ depending on the type of the parameter. For integers, real single-precision, real double-precision and character string data, the value is entered in **VAL1**. For complex values, both **VAL1** and **VAL2** are used; **VAL1** for the real part, and **VAL2** for the imaginary.
3. The parameters used in Rigid Formats are summarized in the table below:

Parameter	APP	SOL	Type	Purpose	Default
ALPHA	All	All	Complex	Scale factor for BGG, KGG, MGG when added to direct input matrices B2GG, K2GG, M2GG respectively.	**(1.0,0.0)**
ASING	**DISP**	3 10 11 12 13 15 52 MULTI	Integer	The default value allows an automatic static condensation of the equations, eliminating the massless degrees of freedom, whenever the **GIV** or **MGIV** eigenvalue extraction method is selected. A value of **-1** detects a fatal condition if there are mass singularities.	**0**
AUTOSPC	**DISP NONLIN**	All	Character	**YES** requests the **AUTOSPC** function with the SPC option. This parameter is overridden by any **AUTOSPC** Case Control command.	**NO**

7 BULK

Mod: V20.1

Parameter	APP	SOL	Type	Purpose	Default
BAILOUT	**DISP NONLIN**	All	Integer	The default value causes program termination if parameter **MAXRATIO** is exceeded. A negative value allows the program to continue. This parameter is overridden by an **AUTOSING** Case Control command.	**0**
BARCONV	**NONLIN**	1 MULTI	Real	Convergence criterion for nonlinear bar cross section yield calculation.	**.01**
BARITER	**NONLIN**	1 MULTI	Integer	Maximum number of iterations for nonlinear bar cross section yield calculation.	**50**
BARREDM	**DISP NONLIN**	All	Character	Requests (or suppresses) the Guyan reduction procedure for the BAR and BEAM element mass matrices, based on the presence of element pin flags. The **NO** option is for compatibility with other versions of NASTRAN.	**YES**
BARTORM	**DISP NONLIN**	All	Character	Requests the inclusion of torsional inertia terms in the BAR element mass matrix, similar to the terms computed for the BEAM element	**NO**
BETA	**All**	All	Complex	Scale factor for direct input matrices B2GG, K2GG, M2GG when added to BGG, KGG, MGG respectively.	**(1.0,0.0)**
BETAH	**HEAT**	9	Real	Integration stability and accuracy control parameter.	**.55**
COUPMASS	**DISP NONLIN**	All	Integer	A value of **+1** selects coupled (consistent) element mass matrices. The default value selects lumped mass.	**-1**
DELTAB	**DISP**	51 52 MULTI	Real	Design variable perturbation factor for design sensitivity calculations.	10^{-3}
DIAGNLPP	**DISP**	11 12 17	Character	**YES** causes the diagonal terms of the matrices transformed to modal coordinates to be retained for analysis.	**NO**
EPSHT	**HEAT**	3	Real	Nonlinear heat transfer convergence test value.	10^{-3}
EPSIO	**DISP**	4	Real	Differential stiffness load convergence test value.	10^{-5}
EPZERO	**DISP NONLIN**	All	Real	The **AUTOSPC** singularity test value.	10^{-8}
FRSHIFT	**NONLIN**	1	Character	Defines a rule for shifting the origin of the Coulomb friction curve during nonlinear GAP analyses. **YES** - Shift occurs at change in loading direction and at each new load step. **UNLOAD** - Shift occurs only during element sensing local change in load direction. **NO** - No shifting occurs.	**YES**

7 BULK

Parameter	APP	SOL	Type	Purpose	Default
G	**DISP**	7 8 9 10 11 12 17 MULTI	Real	Defines uniform structural damping. Requires paramater **W3** in **SOL 9** and **12**.	**0.0**
GPMTX	**NONLIN**	1	Integer	A value of **0** selects the continuum GP matrix.	**1**
GRDPNT	**All**	All	Integer	A value ≥**0** requests the calculation of the weight and balance data for the model.	**-1**
HFREQ	**DISP**	10 11 12 17	Real	Defines the upper limit of the cyclic frequency range of the modes to be used in performing a modal dynamic analysis.	Upper limit is not enforced
IRES	**DISP**	1 2 4 5 13 14 51 MULTI	Integer	A value of **+1** requests print of the residual vectors following solution.	**-1**
	HEAT	1 3			
K6ROT	**DISP** **NONLIN**	All	Real	Provides a stiffness value for plate element "drilling" degrees of freedom. [4]	**0.0**
KTINY	**DISP**	3 8 11 15 17 52 MULTI	Real	A filter for controlling the print of **EKE** output requests. Only values greater than **KTINY** percent will be printed. For example, if **KTINY = 10.0**, all element kinetic energies at least 10% of the total kinetic energy will be printed.	**0.0**
LFREQ	**DISP**	10 11 12 17	Real	Defines the lower limit of the cyclic frequency range of the modes to be used in performing a modal dynamic analysis.	Lower limit is not enforced
LMODES	**DISP**	10 11 12 17	Integer	Specifies the number of lowest frequency modes that will be used in a modal dynamic analysis. This parameter overrides **HFREQ** and **LFREQ**.	All modes with computed eigenvectors are used
MAXIT	**HEAT**	3	Integer	Maximum number of iterations for nonlinear heat transfer solution.	**4**
MAXKITER	**DISP**	4	Integer	The maximum number of differential stiffness updates.	**4**
MAXPITER	**DISP**	4	Integer	Maximum number of load iterations between differential stiffness updates.	**4**

Parameter	APP	SOL	Type	Purpose	Default
MAXRATIO	**All**	All	Real	Filter for decomposition accuracy tests. This number is compared to the ratio of the diagonal of the stiffness matrix to the corresponding term on the diagonal of the triangular factor matrix. Ratios greater than **MAXRATIO** will be identified and program action will be controlled by the value of parameter **BAILOUT**. This parameter is overridden by any **AUTOSING** Case Control command.	10^7
MODACC	**DISP**	11 12	Integer	A value of **+1** requests the use of the modal acceleration method for dynamics data recovery.	**-1**
NOTRACK	**DISP**	MULTI	Integer	A value of **-1** disables automatic mode tracking during dynamic analyses.	**1**
NRB	**DISP**	12	Integer	A value ≥**0** initiates the automatic computation of modal initial condititions when loads are not zero at t=0. The value must be the number of rigid body modes present.	**-1**
PLYDATA	**DISP** **NONLIN**	All	Character	Requests output of the **PSHELL** and **MAT2** data generated from **PCOMPi** input data. Legal character strings are **PRINT, PUNCH** or **BOTH.**	NONE
PRGPST	**DISP** **NONLIN**	All	Character	A value of **NO** will suppress the printing of the GRID Point Singularity Table. This parameter is overridden by any **AUTOSPC** Case Control command.	**YES**
PTHRESH	**DISP**	Substructuring	Real	If the absolute value of an applied load quantity is less than this value it will be set to zero for output.	**0.0**
QTHRESH	**DISP**	Substructuring	Real	If the absolute value of a reaction force quantity is less than this value it will be set to zero for output.	**0.0**
RADLIN	**HEAT**	9	Integer	A value of **+1** causes radiation effects to be linearized.	**-1**
RBDISP with **NRB<0**	**DISP**	12	Character	This parameter has no meaning for the case when parameter **NRB**<**0**.	**NO**
RBDISP with **NRB≥0**	**DISP**	12	Character	A value of **YES** will result in the rigid body displacements being included in the modal solution data recovery.	**YES**
SECTION	**DISP** **NONLIN**	All	Character	Requests output of the **PBAR** or **PBEAM** data generated from **PBAR1** or **PBEAM1** input data. Legal character strings are **PRINT, PUNCH** or **BOTH.**	NONE
SENSVAL	**DISP**	51 52	Character	A value of **SCALED** causes constraint sensitivities to be computed as nondimensional.	NONE
SHKDAMP	**DISP**	17	Integer	A value of 1 requests a print of modal damping in Response Spectrum analysis.	**-1**
SIGMA	**HEAT**	3 9	Real	The Stefan-Boltzman constant.	**0.0**

Parameter	APP	SOL	Type	Purpose	Default
TABS	**HEAT**	3 9	Real	The absolute reference temperature value.	**0.0**
TINY	**DISP**	1 2 3 14 15 MULTI	Real	A filter for controlling the print of **ESE** output requests. Only values greater than **TINY** percent will be printed. For example, if **TINY = 10.0**, all element kinetic energies at least 10% of the total kinetic energy will be printed.	**0.0**
USETPRT	**DISP** **NONLIN**	All	Integer	A value of **0** requests the printing of the USET set definition table sorted by the set type. A value of **1** prints the table sorted by DOF. A **2** selects both of these options. The above tables are sorted by internal point. The same set of tables may be obtained in external sort by using the values **10**, **11** and **12**, respectively.	**-1**
UTHRESH	**DISP**	Substructuring	Real	If the absolute value of a displacement, velocity or acceleration quantity is less than this value it will be set to zero for output.	**0.0**
W3 **W4**	**DISP**	9 12 17	Real	Damping normalization factors for transient response.	**0.0**
W4MODAL	**DISP**	12	Character	**YES** requests that the W4 matrix be entered in modal transient response calculations. If so, parameter **W4** may bot be present in the Bulk Data packet.	**NO**
WTMASS	**DISP** **NONLIN**	All	Real	Multiplier for the system mass matrix. Used to convert weight values to mass values by multiplication.	**1.0**

4. You may provide a default value for **K6ROT** in the Preference File. The default is found in the **<Solution Techniques>** group of the **[UAI/NASTRAN]** Configuration Section. You can print the contents of the Preference File with the command:

   PRINT PREFERENCES

   Your System Support Specialist can provide you with more information about this advanced feature.

Bulk Data Entry PBAR

Defines the properties of a general BAR element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PBAR	PID	MID	A	I1	I2	J	NSM	SHAPE	-cont-
-cont-	Y1	Z1	Y2	Z2	Y3	Z3	Y4	Z4	-cont-
-cont-	K1	K2	I12	YN	ZN	POFFSET	YR	ZR	

PBAR	101	56	2.9		8.76				+A
+A			2.0	4.0					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
A	Area of the BAR cross-section. [2,3]	Real≥0.0	**0.0**
I1,I2,I12	Area moments of inertia. [2,3]	Real≥0.0 $I_1 I_2 > I_{12}^2$	**0.0**
J	Torsional constant. [2,3]	Real≥0.0	**0.0**
NSM	Nonstructural mass per unit length.	Real≥0.0	**0.0**
SHAPE	Cross-sectional shape for nonlinear material analysis. [3]	Character { **BAR** **TUBE** **ROD** }	**BAR**
Yi,Zi	Coordinates of stress recovery points on the element cross section. [4]	Real	0.0
K1,K2	Area factors for shear. [5]	Real≥0.0	0.0
YN,ZN	Coordinates of the neutral axis in the element coordinate system with origin at the shear center.	Real	0.0
POFFSET	Specifies direction of the offset vector specified on the element **CBAR** entry. [6]	Character { **SHEARCEN** **CENTROID** **REFPOINT** }	**SHEARCEN**
YR,ZR	Coordinates of the reference point in the element coordinate system with origin at the shear center.	Real	0.0

7 BULK

Remarks:

1. For structural analyses, **PBAR** entries may only reference **MAT1**, **MAT1NL** or **MAT1NL1** material data. For heat transfer analyses, they may only reference **MAT4** or **MAT5** material data.
2. Any of the fields **A**, **I1**, **I2**, or **J** may be left blank or given a value of 0.0. In such cases, the BAR element will have no stiffness in the corresponding direction.
3. For nonlinear material analysis, the strain evaluation points are determined from the specified **SHAPE**, and the values of **A**, **I1**, and **I2**. A more accurate evaluation is automatically made if the **PBAR1** Bulk Data entry is used.
4. The location of stress recovery points are specified in the element coordinate system.
5. If area factors for shear, **K1** and **K2**, are entered as **1.0**, then all of the BAR cross-sectional area is effective for flexibility. Factors of 0.833 are accepted standards for shear flexibility of a rectangular cross section bar, and 0.50 are appropriate for a rod shape. If the factors are not entered, then the transverse shear flexibilities in the corresponding directions are assumed to be zero.
6. The direction may be selected from:
   - **SHEARCEN** - The element shear center
   - **CENTROID** - The centroidal location intersected by the neutral axis
   - **REFPOINT** - An arbitrary reference point defined by the coordinates **YR** and **ZR**.

Bulk Data Entry PBAR1

Defines the properties of a BAR element by specifying its cross-sectional characteristics.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PBAR1	PID	MID	SHAPE	D1	D2	D3	D4	D5	-cont-
-cont-	NSM	POFFSET		D6	D7	D8	D9	D10	

PBAR1	101	56	TUBE	2.0	0.1				+A
+A	1.25								

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
SHAPE	Cross-sectional shape. [2]	Character { **I**, **T**, **BOX**, **BAR**, **TUBE**, **ROD**, **HAT**, **GBOX** }	**BAR**
Di	Cross-sectional dimensions. [2]	Real>0.0	[2]
NSM	Nonstructural mass per unit length.	Real	**0.0**
YN,ZN	Coordinates of the neutral axis in the element coordinate system with origin.	Real	**0.0**
POFFSET	Specifies direction of the offset vector specified on the element **CBAR** entry. [3]	Character { **SHEARCEN**, **CENTROID**, **C**, **D** }	**SHEARCEN**
YR,ZR	Coordinates of the reference point in the element coordinate system.	Real	**0.0**

7 BULK

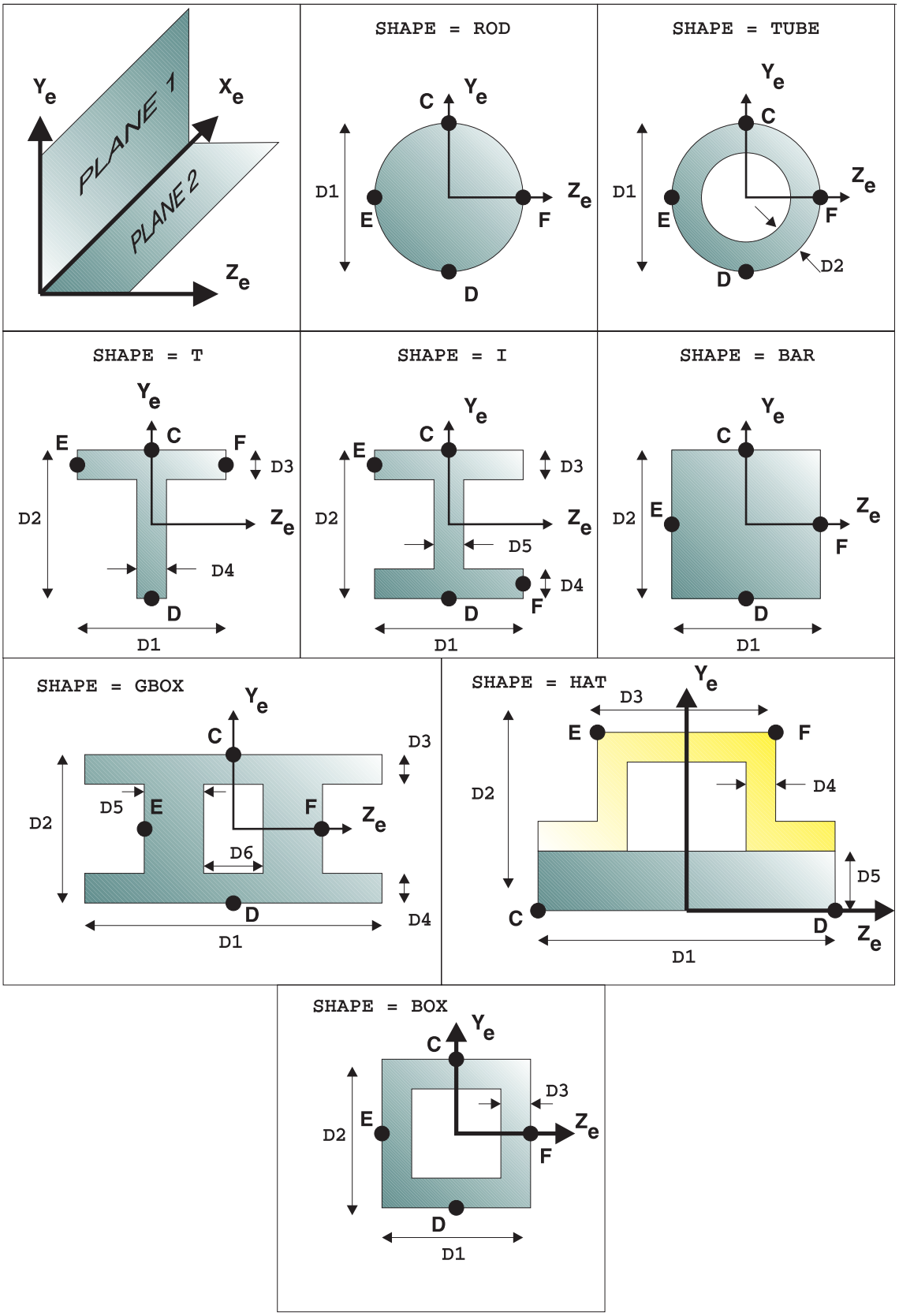
Remarks:

1. For structural analyses, **PBAR1** entries may only reference **MAT1**, **MAT1NL** or **MAT1NL1** material data. For heat transfer analyses, they may only reference **MAT4** or **MAT5** material data.
2. The cross-sectional properties and shear flexibility factors of the BAR are computed using the **SHAPE** and **Di** geometric data as defined by the figures on the following page. The stress recovery points are also shown. Note that the orientation of the element coordinate system is important for the element definition.
3. The direction may be selected from:
   - **SHEARCEN** - The element shear center
   - **CENTROID** - The centroidal location
   - **C**,**D** - A location specified on the element cross-section as defined in the figure on the following page.

   Recall that the element coordinate system origin is at the shear center, and that the neutral axis passes through the centroid.
4. You may print the section properties that are computed using the Bulk Data entry:

   ```
   PARAM,SECTION,PRINT
   ```

Definition of Cross-Sectional Geometry and Stress Recovery Points



7 BULK

Bulk Data Entry PBEAM

Defines the properties of a general BEAM element.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
PBEAM	PID	MID	AA	I1A	I2A	I12A	JA	NSMA	-cont-
-cont-	Y1A	Z1A	Y2A	Z2A	Y3A	Z3A	Y4A	Z4A	-cont-
THE NEXT TWO CONTINUATIONS ARE REPEATED FOR EACH INTERMEDIATE STATION [I]									
-cont-	RESULTS	STATN	A	I1	I2	I12	J	NSM	-cont-
-cont-	Y1	Z1	Y2	Z2	Y3	Z3	Y4	Z4	-cont-
-cont-	K1	K2	S1	S2	NSIA	NSIB	CWA	CWB	-cont-
-cont-	YMA	ZMA	YMB	ZMB	YNA	ZNA	YNB	ZNB	-cont-
-cont-	SHAPE	BEHAVIOR	POFFSET	YRA	ZRA	YRB	ZRB		

All BEAM Elements Require the First Record, and, if Necessary, the second:

PBEAM	156	101	2.5	2.65	9.82				+A
+A	1.5	0.8							+B

Only Elements with Intermediate Stations Require Groups of Two Records for each Station:

+B	YES	0.667	7.4	23.5	12.9			0.12	+C
+C	1.0	1.0	2.0	2.0					+D
+D	YES	0.333	5.2	7.6	2.0			0.15	+E
+E	.5	.5	.75		1.0				+F

The Data Entry Requires the Last Three Records if any Fields in Them are Used:

+F					.5	.5	.75	1.0	+G
+G									+H
+H	TUBE	COLUMN							

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
AA	Cross sectional area at End A. [2,3]	Real≥0.0	**0.0**
I1A,I2A	Area moments of inertia about the neutral axis at End A for planes 1 and 2. [2,3]	Real≥0.0	**0.0**
I12A	Area product of inertia at End A. [2,3]	Real≥0.0 $I_{1A}I_{2A}>I_{12A}^2$	**0.0**
JA	Torsional stiffness parameter at End A. [2,3]	Real≥0.0	**0.0**
NSMA	Nonstructural mass per unit length at End A.	Real≥0.0	**0.0**

7 BULK

YiA,ZiA	Coordinates of stress recovery points on the cross section at End A. [4]	Real≥0.0	**0.0**
RESULTS	Solution results recovery option for this axial station. [5]	Character	Required
STATN	Position of the axial station as a fraction of the element length measured from End A.	Real>0.0	[6]
A	Cross-sectional area at this axial station.	Real≥0.0	[3]
I1,I2,I12	Moments of inertia at this axial station.	Real≥0.0 $I_1 I_2 > I_{12}^2$	[3]
J	Torsional stiffness at this axial station.	Real≥0.0	[3]
NSM	Nonstructural mass per unit length at this axial station.	Real≥0.0	[3]
Yi,Zi	Coordinates of stress recovery points on the cross-section at this station. [4]	Real≥0.0	**0.0**
K1,K2	Area factors for shear for planes 1 and 2. [7]	Real≥0.0	**1.0**
S1,S2	Shear relief coefficients due to taper for planes 1 and 2.	Real	**0.0**
NSIA, **NSIB**	Moments of inertia per unit length about the nonstructural mass center of gravity at End A and End B.	Real≥0.0	**0.0**
CWA, **CWB**	Warping coefficients for End A and End B.	Real≥0.0	**0.0**
YMA,ZMA **YMB,ZMB**	Coordinates of the nonstructural mass center of gravity at End A and End B. [4]	Real	**0.0**
YNA,ZNA **YNB,ZNB**	Coordinates of the neutral axis at End A and End B. [4]	Real	**0.0**
SHAPE	Cross-sectional shape for nonlinear material analysis correction. [8]	Character { **BAR** **TUBE** **ROD** }	**BAR**
BEHAVIOR	Selects beam or column behavior for material nonlinear analysis. [9]	Character { **BEAM** **COLUMN** }	**COLUMN**
POFFSET	Specifies direction of the offset vector specified on the element **CBEAM** entry. [10]	Character { **SHEARCEN** **CENTROID** **REFPOINT** }	**SHEARCEN**
YRA,ZRA, **YRB,ZRB**	Coordinates of the reference points at End A and End B in the element coordinate system.	Real	**0.0**

Remarks:

1. For structural analyses, **PBEAM** entries may reference only **MAT1** material data. For heat transfer analyses, they may reference only **MAT4** and **MAT5** material data.
2. An alternate input for standard shapes is the **PBEAM1** Bulk Data entry.
3. Any of the fields **AA**, **I1A**, **I2A**, **I12A**, or **JA** may be left blank or given a value of 0.0. If the section properties **A**, **I1**, **I2**, **I12**, **J** and **NSM** are omitted for the axial station defining End B, then the properties at End A will be used. For other intermediate axial stations, missing properties will be determined by linear interpolation between the values at End A and End B.
4. All coordinates for stress recovery, nonstructural mass center of gravity and neutral axis are in the element coordinate system.
5. The **RESULTS** options, which include sresses, forces and strain, are described in the following table:

RESULTS value	Meaning
YES	Stress recovery points are given on the next continuation entry and they must be present.
YESA	The same results recovery points will be used for the axial station as were used at End A.
NO	No results recovery will be performed.

   If you select **YES**, then the following contunation entry must be present for the station you are defining. If this is not the case, the continuation entry ***must*** be omitted. For nonlinear analyses, results are recovered only at the two end stations. Any **RESULTS** options for intermediate stations are ignored.
6. From zero to nine intermediate axial stations between End A and End B may be defined. The ordering of these stations is not important, but if defined, one must have a value of **STATN=1.0** that corresponds to End B.
7. If area factors for shear, **K1** and **K2**, are not entered, then all of the BEAM cross-sectional area is effective for flexibility. Factors of 0.833 are accepted standards for shear flexibility of a rectangular shape, and 0.50 are appropriate for a rod shape. If a factor is entered as **0.0**, then the transverse shear flexibility in the corresponding direction is assumed to be zero.
8. Nonlinear material behavior is corrected for basic geometric cross-sectional shapes. The three shapes that may be selected are shown in the figures for the **PBAR1** and **PBEAM1** Bulk Data entries.
9. **BEHAVIOR** is used in nonlinear analysis to specify the type of behavior. If **BEAM** is selected, then only the end stations behave nonlinearly. If **COLUMN** is selected, then all stations have nonlinear behavior.
10. The direction may be selected from:
    - **SHEARCEN** - The element shear center
    - **CENTROID** - The centroidal location
    - **REFPOINT** - Two arbitrary reference points at End A and End B defined by the coordinates **YRA**, **ZRA**, **YRB**, and **ZRB**.

    Recall that the element coordinate system origin is at the shear center, and that the neutral axis passes through the centroid.
11. If you use the continuation entry with the **K1** field, then you must also use the first continuation with the **Y1A** field. If you specify the **YMA** field, then you must include the continuation entry with the **K1** field.

Bulk Data Entry PBEAM1

Defines the properties of a BEAM element by specifying its cross-sectional characteristics.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PBEAM1	PID	MID	SHAPE	D1A	D2A	D3A	D4A	D5A	-cont-
-cont-	NSMA	NSMB	NSTAT	D1B	D2B	D3B	D4B	D5B	-cont-
-cont-	BEHAVIOR	STOUT	POFFSET						

PBEAM1	101	56	TUBE	0.1					+A
+A	1.25								

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
SHAPE	Cross-sectional shape. [2]	Character { **I**, **T**, **BOX**, **BAR**, **TUBE**, **ROD**, **HAT** }	BAR
DiA, DiB	Cross-sectional dimensions at End A and End B of the element. [2]	Real≥0.0	[2,3]
NSMA, NSMB	Nonstructural mass per unit length at End A and End B of the element.	Real≥0.0	**0.0**
NSTAT	Number of BEAM stations at which solutions results will be printed. [4]	2≤Integer≤11	5
BEHAVIOR	Selects beam or column behavior for material nonlinear analysis. [5]	Character { **BEAM**, **COLUMN** }	BEAM
STOUT	Stress output selection. **YES** prints all sections, while **NO** suppresses print for intermediate sections.	Character { **NO**, **YES** }	NO
POFFSET	Specifies direction of the offset vector specified on the element **CBAR** entry. [3]	Character { **SHEARCEN**, **CENTROID**, **C**, **D** }	**SHEARCEN**
YR, ZR	Coordinates of the reference point in the element coordinate system.	Real	**0.0**

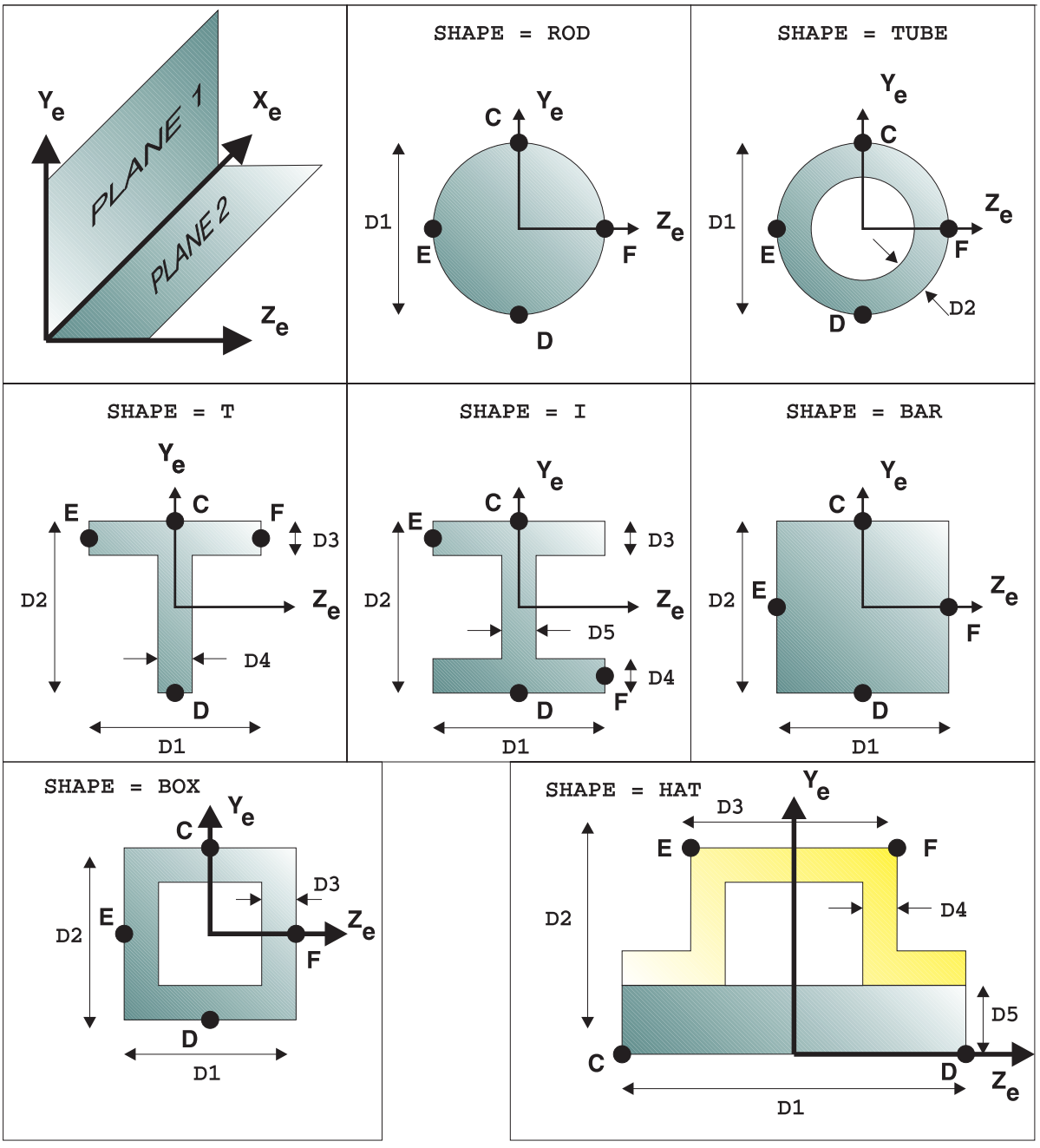
7 BULK

Remarks:

1. For structural analyses, **PBEAM1** entries may only reference **MAT1** material data. For heat transfer analyses, they may only reference **MAT4** or **MAT5** material data.
2. The sectional properties of the BEAM are computed using the **SHAPE**, **DiA** and **DiB** geometric data as defined by the figures on the following page. The default stress recovery points are also shown. Note that the orientation of the element coordinate system is important for the element definition.
3. If your BEAM element is not tapered, then you need not enter cross-sectional dimensions at End B.
4. BEAM stress and force resultants are output only at the two end points by default. **NSTAT** may be used to request output at up to 9 equally spaced intermediate stations in addition to the two end points, i.e. a total of eleven stations are output.
5. **BEHAVIOR** is used in nonlinear analysis to specify the type of behavior. If **BEAM** is selected, then only the end stations behave nonlinearly. If **COLUMN** is selected, then all stations have nonlinear behavior.
6. You may print the section properties that are computed using the Bulk Data entry:

```
PARAM,SECTION,PRINT
```

Definition of Cross-Sectional Geometry and Stress Recovery Points



7 BULK

Bulk Data Entry PBUSH

Defines the properties of a Bushing element by specifying the spring stiffness and damping factors.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PBUSH	PID	K1	K2	K3	K4	K5	K6		-cont-
-cont-		B1	B2	B3	B4	B5	B6		-cont-
-cont-		TIDK1	TIDK2	TIDK3	TIDK4	TIDK5	TIDK6		-cont-
-cont-		TIDB1	TIDB2	TIDB3	TIDB4	TIDB5	TIDB6		-cont-
-cont-	SA	ST	EA	ET					

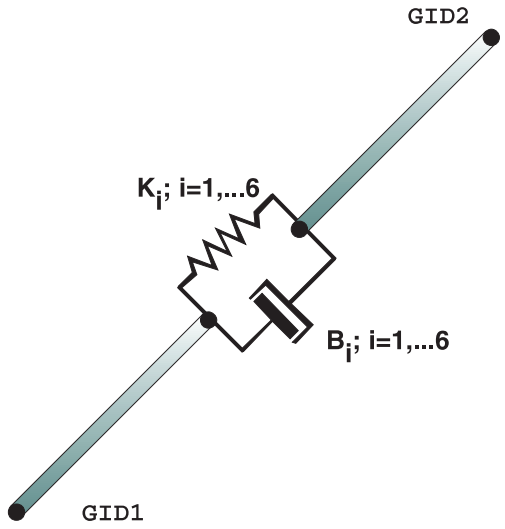
PBUSH	1001	0.1	0.1	0.1	0.1	0.1	0.1		

Field	*Comments*	*Data Range*	*Default*
PID	Property identification number	Integer>0	Required
Ki	Constant stiffness values in directions 1 through 6 (1,2)	Real	**0.0**
Bi	Constant damping values in directions 1 through 6 (1,2)	Real	**0.0**
TIDKi	Identification number of a **TABLEDi** or **TABLENL** entry that defines the stiffness vs. frequency or force vs. deflection relationship (2,3).	Integer>0	**0**
TIDBi	Identification number of a **TABLEDi** entry that defines the damping vs. frequency relationship (2).	Integer>0	**0**
SA	Stress recovery coefficients in axial (translational) directions 1 through 3 (4).	Real>0.0	**0.0**
ST	Stress recovery coefficients in torsional (rotational) directions 4 through 6 (4).	Real>0.0	**0.0**
EA	Strain recovery coefficients in axial directions 1 through 3 (5).	Real>0.0	**0.0**
ET	Strain recovery coefficients in torsional directions 4 through 6 (5).	Real>0.0	**0.0**

Remarks:

1. If constant stiffness and damping values are used, the element behaves linearly.
2. In direct frequency response the stiffness and damping vs. frequency curves are specified with **TABLEDi** Bulk Data entries. The **Ki** and **Bi** values are used to compute initial **KGG** and **BGG** matrices which are updated during frequency response analysis using the **TABLEDi** input. The following figure illustrates the element configuration.

7 BULK



3. In nonlinear static analysis, if the user specifies a **TIDKi**, the corresponding nominal stiffness value is ignored, and the slope of the force-deflection curve described by the referenced **TABLENL** is taken as the stiffness value. The constant stiffness values are ignored in this case.

   In all analyses except Direct Frequency Response,

4. The element stresses are computed by multiplying the stress coefficients with the recovered element forces.

5. The element strains are computed by multiplying the strain coefficients with the recovered element forces.

6. The BUSH element degrees of freedom need to be in the *d-set* if frequency dependent stiffness and or damping properties are specified for the element.

Bulk Data Entry PCOMP

Defines the properties of an n-ply laminated composite material for a plate element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PCOMP	PID	Z0	NSM	SBOND	FT	T0	GE	LAM	-cont-
-cont-	MID1	T1	θ1	SOUT1	MID2	T2	θ2	SOUT2	-cont-
-cont-	MID3	T3	θ3	SOUT3	*CONTINUES IN GROUPS OF 4*				-cont-

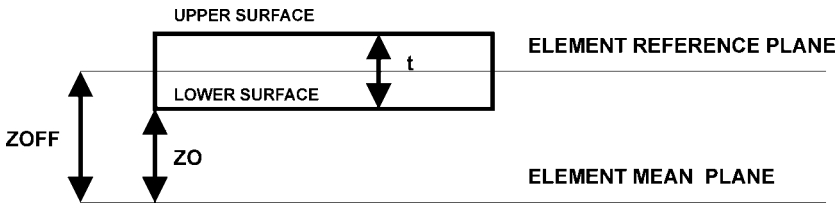
PCOMP	100	-0.5	1.5	5.+3	HOFF			SYMMEM	+A
+A	150	0.05	90.	YES			-45.		+B
			45.0						

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	0<Integer<10^6	Required
Z0	Offset of the laminate lower surface from the element mean plane. A positive value means the $+Z_e$ direction. [1,2]	Real	[1]
NSM	Non-structural mass per unit area.	Real≥0.0	**0.0**
SBOND	Allowable shear stress of the bonding material. [3]	Real≥0.0	[3]
FT	Failure theory. [4]	Character { **HILL** **HOFF** **TSAI** **STRESS** **STRAIN** }	[4]
T0	Thermal expansion reference temperature. [5]	Real≥0.0	**0.0**
GE	Structural damping coefficient. [6]	Real≥0.0	**0.0**
LAM	Lamination generation option. [7]	Character { **ALL** **SYM** **MEM** **SYMMEM** }	**ALL**
MIDi	Material identification number of the ith layer. [8,9,10]	Integer>0	[10]
Ti	Thickness of layer **i**. [8,10]	Real>0.0	[10]
qi	Angle between the longitudinal direction of the fibers of layer **i** and the material X-axis. [8,10]	Real	[10]
SOUTi	Stress output request for layer **i**.	Character { **YES** **NO** }	**NO**

7 BULK

Remarks:

1. For composities there are two methods for specifying the offset of the element reference plane from the element mean plane: **Z0** on this entry and **ZOFF** on the **CQUAD4**, **CQUAD8**, **CTRIA3** or **CTRIA6** Bulk Data entries. The distinction is shown in the figure below:



2. You may only specify a **Z0** on this entry if the **ZOFF** field of any element, **CQUAD4**, **CQUAD8**, **CTRIA3**, or **CTRIA6**, referencing it is blank. The default value for **Z0** is -t/2, where t is the overall thickness of the laminate.
3. **SBOND** is required if bonding material failure index calculations are desired. A blank field means that no bonding failure checks are made.
4. The failure theory is used to determine the element failure on a ply-by-ply basis. The available theories are:
   - **HILL** - Hill Theory
   - **HOFF** - Hoffman Theory
   - **TSAI** - Tsai-Wu Theory
   - **STRESS** - Maximum Stress Theory
   - **STRAIN** - Maximum Strain Theory

   A blank field means no failure checks are made.

   NOTE: Print of individual ply layer failure indices requires that the Case Control stress request command contain the **LAYER** key word as shown by example below.

   ```
   STRESS( PRINT,LAYER ) = ALL
   ```
5. **T0** on this Bulk Data entry will override any such data specified on **MATi** Bulk Data.
6. **GE** on this Bulk Data entry will override any such data specified on **MATi** Bulk Data.
7. To minimize input requirements several lamination options, **LAM**, are available:
   - **ALL** - indicates that every ply is specified.
   - **SYM** - indicates that ply layup is symmetric and that the plies up to and including any engulfing the center line are specified.
   - **MEM** - indicates that the plies only support membrane behavior.
   - **SYMMEM** - indicates a symmetric layup of membrane only plies.

7 BULK

8. The plies are numbered from 1 to n beginning with the bottom layer.
9. The material properties, **MIDi**, may reference only **MAT1**, **MAT2** and **MAT8** Bulk Data entries.
10. If any of the **MIDi**, **Ti** or **θi** are blank, then the last non-blank values specified for each will be used to define the values for the ply.
11. All *referenced* composite properties are converted to equivalent properties for internal program use. You may use the **PARAM**,**PLYDATA** Bulk Data entry to obtain an output of the equivalent properties.

Bulk Data Entry PCOMP1

Defines the properties of an n-ply laminated composite material for plate elements where all plies are composed of the same material and are of equal thickness.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PCOMP1	PID	Z0	NSM	SBOND	FT		MID	LAM	-cont-
-cont-	TPLY	θ1	θ2	*CONTINUES WITH LIST OF VALUES*					-cont-

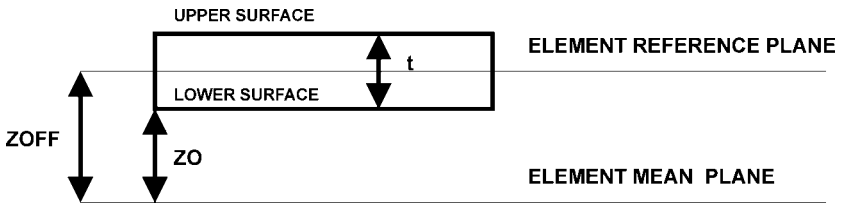
PCOMP1	100	-0.5	1.7	5.+3	TSAI		200	SYM	+A
+A	0.25	-45.	45.	90.	-45.	45.			

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	0<Integer<10^6	Required
Z0	Offset of the laminate lower surface from the element mean plane. A positive value means the $+Z_e$ direction . [1,2]	Real	[1]
NSM	Non-structural mass per unit area.	Real≥0.0	**0.0**
SBOND	Allowable shear stress of the bonding material. [3]	Real≥0.0	[3]
FT	Failure theory. [4]	Character { **HILL** **HOFF** **TSAI** **STRESS** **STRAIN** }	[4]
MID	Material identification number for all layers. [5]	Integer>0	Required
LAM	Lamination generation option. [6]	Character { **ALL** **SYM** **MEM** **SYMMEM** }	**ALL**
TPLY	Thickness of each layer.	Real>0.0	Required
qi	Angle between the longitudinal direction of the fibers of layer **i** and the material X-axis. [7]	Real	Required

7 BULK

Remarks:

1. For composities there are two methods for specifying the offset of the element reference plane from the element mean plane: **Z0** on this entry and **ZOFF** on the **CQUAD4**, **CQUAD8**, **CTRIA3** or **CTRIA6** Bulk Data entries. The distinction is shown in the figure below:



2. You may only specify a **Z0** on this entry if the **ZOFF** field of any element, **CQUAD4**, **CQUAD8**, **CTRIA3**, or **CTRIA6**, referencing it is blank. The default value for **Z0** is -t/2, where t is the overall thickness of the laminate.
3. **SBOND** is required if bonding material failure index calculations are desired. A blank field means that no bonding failure checks are made.
4. The failure theory is used to determine the element failure on a ply-by-ply basis. The available theories are:
   - **HILL** - Hill Theory
   - **HOFF** - Hoffman Theory
   - **TSAI** - Tsai-Wu Theory
   - **STRESS** - Maximum Stress Theory
   - **STRAIN** - Maximum Strain Theory

   A blank field means no failure checks are made.
5. The material property, **MID**, may reference only **MAT1**, **MAT2** and **MAT8** Bulk Data entries.
6. To minimize input requirements several lamination options, **LAM**, are available:
   - **ALL** - indicates that every ply is specified.
   - **SYM** - indicates that ply layup is symmetric and that the plies up to and including any engulfing the center line are specified.
   - **MEM** - indicates that the plies only support membrane behavior.
   - **SYMMEM** - indicates a symmetric layup of membrane only plies.
7. The plies are numbered from 1 to n beginning with the bottom layer.
8. All *referenced* composite properties are converted to equivalent properties for internal program use. You may use the **PARAM**,**PLYDATA** Bulk Data entry to obtain an output of the equivalent properties.

Bulk Data Entry PCOMP2

Defines the properties of an n-ply laminated composite material for plate elements where all plies are of the same material.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PCOMP2	PID	Z0	NSM	SBOND	FT		MID	LAM	-cont-
-cont-	T1	θ1	T2	θ2	*CONTINUES IN GROUPS OF 2*				-cont-

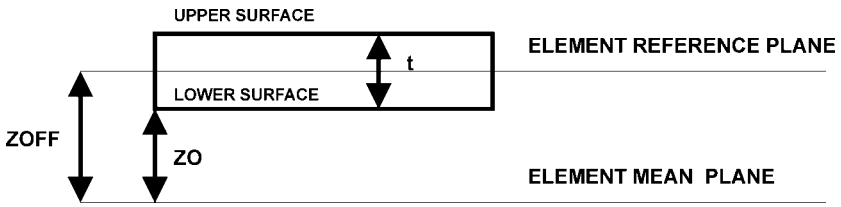
PCOMP2	100	-0.05	1.7	5.+3	TSAI		200	SYM	+A
+A	0.25	-45.0	0.5	90.0	0.25	45.0			

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	0<Integer<10^6	Required
Z0	Offset of the laminate lower surface from the element mean plane. A positive value means the $+Z_e$ direction . [1,2]	Real	[1]
NSM	Non-structural mass per unit area.	Real≥0.0	**0.0**
SBOND	Allowable shear stress of the bonding material. [3]	Real≥0.0	[3]
FT	Failure theory. [4]	Character { **HILL** **HOFF** **TSAI** **STRESS** **STRAIN** }	[4]
MID	Material identification number for all layers. [5]	Integer>0	Required
LAM	Lamination generation option. [6]	Character { **ALL** **SYM** **MEM** **SYMMEM** }	**ALL**
Ti	Thickness of layer **i**. [7,8]	Real≥0.0	Required
qi	Angle between the longitudinal direction of the fibers of layer **i** and the material X-axis. [7,8]	Real	Required

7 BULK

Remarks:

1. For composities there are two methods for specifying the offset of the element reference plane from the element mean plane: **Z0** on this entry and **ZOFF** on the **CQUAD4**, **CQUAD8**, **CTRIA3** or **CTRIA6** Bulk Data entries. The distinction is shown in the figure below:



2. You may only specify a **Z0** on this entry if the **ZOFF** field of any element, **CQUAD4**, **CQUAD8**, **CTRIA3**, or **CTRIA6**, referencing it is blank. The default value for **Z0** is -t/2, where t is the overall thickness of the laminate.
3. **SBOND** is required if bonding material failure index calculations are desired. A blank field means that no bonding failure checks are made.
4. The failure theory is used to determine the element failure on a ply-by-ply basis. The available theories are:
   - **HILL** - Hill Theory
   - **HOFF** - Hoffman Theory
   - **TSAI** - Tsai-Wu Theory
   - **STRESS** - Maximum Stress Theory
   - **STRAIN** - Maximum Strain Theory

   A blank field means no failure checks are made.
5. The material property, **MID**, may reference only **MAT1**, **MAT2** and **MAT8** Bulk Data entries.
6. To minimize input requirements several lamination options, **LAM**, are available:
   - **ALL** - indicates that every ply is specified.
   - **SYM** - indicates that ply layup is symmetric and that the plies up to and including any engulfing the center line are specified.
   - **MEM** - indicates that the plies only support membrane behavior.
   - **SYMMEM** - indicates a symmetric layup of membrane only plies.
7. The plies are numbered from 1 to n beginning with the bottom layer.
8. If any of the **Ti** or **θi** are blank, then the last non-blank values specified for each will be used to define the values for the ply.
9. All *referenced* composite properties are converted to equivalent properties for internal program use. You may use the **PARAM**,**PLYDATA** Bulk Data entry to obtain an output of the equivalent properties.

Bulk Data Entry PCONEAX

Defines the properties of a CONEAX conical shell element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PCONEAX	PID	MID1	T1	MID2	I	MID3	T2	NSM	-cont-
-cont-	Z1	Z2	ϕ1	ϕ2	ϕ3	ϕ4	ϕ5	ϕ6	-cont-
-cont-	ϕ7	ϕ8	ϕ9	ϕ10	ϕ11	ϕ12	ϕ13	ϕ14	

PCONEAX	102	104	1.0	106	16.3	108	2.1	0.6	+A
+A	0.001	-0.013	18.3	27.4	45.	60.	75.		

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MIDi	Material identification number for membrane, bending, and transverse shear. [1]	Integer>0	Required
T1,T2	Membrane thickness and transverse shear thickness.	Real>0.0	Required
I	Moment of Inertia per unit width.	Real	Required
NSM	Nonstructural mass per unit area.	Real	**0.0**
Z1,Z2	Fiber distances for stress recovery.	Real	**0.0**
fi	Azimuthal coordinates (in degrees) for stress recovery.	Real	**0.0**

Remarks:

1. **PCONEAX** entries may only reference **MAT1** material data.
2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

7 BULK

Bulk Data Entry PDAMP

Defines the damping value for DAMP1 and DAMP3 damping elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PDAMP	PID	B							

PDAMP	114	42.5							+A

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
B	Damping value. [1]	Real	**0.0**

Remarks:

1. Negative damping values, although allowed, should be used with great caution.
2. Damping values may be defined directly with **CDAMP2** and **CDAMP4** Bulk Data entries. A structural viscous damper, **CVISC**, may also be used for geometric GRID points.

7 BULK

Bulk Data Entry PELAS

Defines the stiffness, damping coefficient, and stress coefficient of spring elements ELAS1 and ELAS3.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PELAS	PID	K	GE	S					
PELAS	7	4.29	0.06	7.92					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
K	Spring constant. [1]	Real	Required
GE	Damping coefficient.	Real	**0.0**
S	Stress coefficient. [2]	Real	**0.0**

Remarks:

1. Negative spring constants, while allowed, should be used with great caution.
2. The element stresses are computed by multiplying the stress coefficient into the recovered element forces.
3. Spring constants may be defined directly with **CELAS2** and **CELAS4** Bulk Data entries.

7 BULK

Bulk Data Entry PGAP

Defines the properties of the GAP element.

Format and Example:

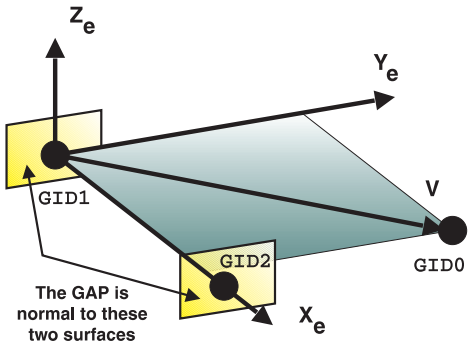
1	2	3	4	5	6	7	8	9	10
PGAP	PID	U0	F0	KCLS	KOPN	KTRAN	μy	μz	
PGAP	99	0.5		1.+9	10.0	1.+8	0.6	0.4	

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
U0	Initial gap opening.	Real	**0.0**
F0	Axial force preload value.	Real	**0.0**
KCLS	Axial stiffness for closed gap. [1]	Real≠0.0	Required
KOPN	Axial stiffness for opened gap. [1]	Real≠0.0	KCLSx10^{-10}
KTRAN	Transverse stiffness when surfaces are in contact.	Real	**0.0**
my	Coefficient of friction in y transverse direction.	Real≥0.0	**10^{-6}**
mz	Coefficient of friction in z transverse direction.	Real≥0.0	**10^{-6}**

Remarks:

1. A rule-of-thumb for selecting these values is to set **KCLS** to 10^3 or 10^4 times the local stiffness, and **KOPN** to 10^{-3} or 10^{-4} times the local stiffness. The ratio of **KCLS** to **KOPN** may be 10^8 for smaller models, but should be closer 10^6 for larger, more complex problems. These are simply guidelines. Individual problems may require different values.
2. The measurements for displacement and gap opening are made in the element coordinate system.
3. The following rules determine which axial stiffness value, *K*, is used:

If $u1 - u2 \geq$ **U0**, then $K =$ **KCLS**.
If $u1 - u2 <$ **U0**, then $K =$ **KOPN**.



7 BULK

Bulk Data Entry PHBDY

Defines the properties of the HBDY heat boundary element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PHBDY	PID	MID	AF	E	α	R1	R2		
PHBDY	100	103	300.	0.79					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number used for convective film coefficient and thermal capacity. [1]	Integer	[1]
AF	Area factor. [2]	Real≥0.0	**0.0**
E	Emissivity which is used only for radiation calculations.	0.0≤Real≤1.0	**0.0**
α	Absorptivity which is used only for thermal vector flux calculations.	0.0≤Real≤1.0	**E**
R1,R2	Radii of elliptic cylinder for HBDY type **ELCYL**.	Real	**0.0**

Remarks:

1. The **MID** must reference **MAT4** data. The convective film coefficient and thermal capacity are in terms of unit area. If no material is referenced the element convection and heat capacity are zero.
2. The area factor **AF** is used to determine the effective area for the HBDY element. When the element type is **POINT**, **AF** is simply the effective area. When the type is **LINE** or **ELCYL**, then **AF** is the effective width which, when multiplied by the effective length, yields the effective area. For other HDBY types, the effective area is automatically calculated.

7 BULK

Bulk Data Entry PILEOR

Defines default values of properties and orientation of PILE elements.

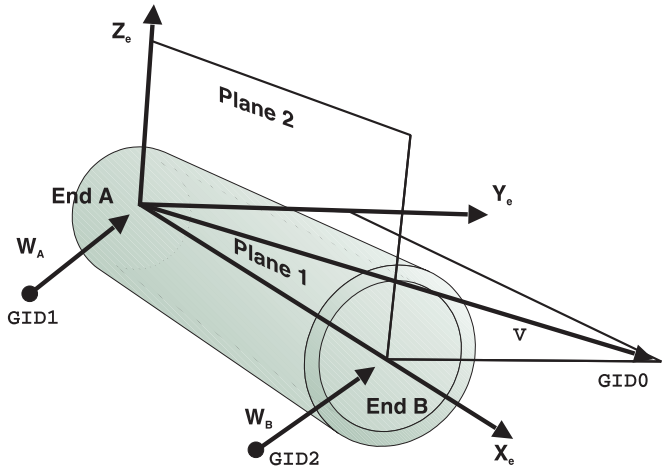
Format and Example:

1	2	3	4	5	6	7	8	9	10
PILEOR		PID				V1	V2	V3	
						G0			
PILEOR		20				1.0	0.0	0.0	

Field	*Contents*	*Data Range*	*Default*
PID	Identification number of **PPILE** or **PPILE1** property entry.	Integer>0	Required
Vi	Components of a vector, **V**, originating at End A. This vector, along with the vector from End A to End B, is used to determine the orientation of the element coordinate system. **V** must be specified in the output coordinate system for GRID **GID1**.	Real	[1,2]
G0	GRID point identification number used to define element orientation.	Integer>0	[1,2]

Remarks:

1. At least one of the data fields **PID**, **G0**, or **Vi** must be non-blank.
2. If Field 7 contains an integer value, then the GRID point **G0** is used to define the PILE element coordinate system orientation as shown below. If it contains a real value, or is left blank, then the vector **G0** is used.



3. The contents of fields on this entry are used for any **CPILE** entry whose corresponding fields are blank.
4. Only one **PILEOR** entry may appear in the Bulk Data packet.

7 BULK

Bulk Data Entry PLOAD

Defines a static pressure load on a triangular or quadrilateral surface.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PLOAD	LID	P	GID1	GID2	GID3	GID4			
PLOAD	205	-23.4	401	403	404				

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
P	Pressure value. [2,3]	Real	**0.0**
GIDi	GRID point identification numbers. [4]	Integer>0	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. The direction of the pressure is computed according to the right-hand rule from the GRID points **GID1**, **GID2**, and **GID3**. The direction of **P** is given by the vector *V* defined as:

   $$\vec{V} = \frac{\mathbf{GID1}\vec{\ }\mathbf{GID2} \times \mathbf{GID1}\vec{\ }\mathbf{GID3}}{|\mathbf{GID1}\vec{\ }\mathbf{GID2} \times \mathbf{GID1}\vec{\ }\mathbf{GID3}|}$$

   The total load on the surface is divided into equal parts and applied to the GRID points as concentrated loads.

3. In the case of irregular or warped quadrilateral surfaces, the surface is divided into two sets of overlapping triangular surfaces. Each triangular surface is bounded by two of the sides and one of the diagonals of the quadrilateral. One quarter of the pressure is then applied to each triangle and distributed as above.

4. The GRID points define either a triangular or a quadrilateral surface to which a pressure is applied. For a triangular surface, **GID4** must be left blank.

7 BULK

Bulk Data Entry PLOAD1

Defines concentrated, uniformly distributed, or linearly distributed applied loads to the BAR or BEAM elements at selected points along the element axis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PLOAD1	LID	EID	TYPE	SCALE	X1	P1	X2	P2	

PLOAD1	3	47	FY	FRPR	0.3	-100.0	0.7	-140.0	

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
EID	Element identification number.	Integer>0	Required
TYPE	The load type. [2]	Character { **FX** or **MX**, **FY** or **MY**, **FZ** or **MZ**, **FXE** or **MXE**, **FYE** or **MYE**, **FZE** or **MZE** }	Required
SCALE	The scaling rule for the positions **X1** and **X2**. [3]	Character { **LE**, **FR**, **LEPR**, **FRPR** }	Required
X1,X2	Two positions along the element between which the load is applied. [4]	Real **X2**≥**X1**≥0.0	**0.0**
P1,P2	Load intensity per unit length at positions **X1** and **X2**. [4,5,6]	Real	**0.0**

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. The load **TYPE**s may be forces or moments in the Basic Coordinate System as specified by the strings **FX, FY, FZ, MX, MY, MZ**; or in the element coordinate system as specified by **FXE, FYE, FZE, MXE, MYE** and **MZE**.

3. The **SCALE** defines the manner in which the positions **X1** and **X2** relate to the element X-axis. If they are specified as actual lengths, the string **LE** is used. If they are given as fractional lengths, then **FR** is used. Only if loads are specified in the Basic Coordinate System can the positions be given as lengths, or fractions, of the projected length of the element. These cases use the strings **LEPR** and **FRPR**.

4. If both **X1** and **X2** are given, the resulting load will vary lineary from **P1** at **X1** to **P2** at **X2**. If **X2** is not given, or it is equal to **X1**, then the concentrated load **P1** will be applied at **X1**.

5. If the loads **P1** and **P2** are equal, a uniform load will be applied between **X1** and **X2**.

6. For the BEAM element, the load is applied along the line that connects the shear centers.

Bulk Data Entry PLOAD2

Defines a uniform static pressure load applied to plate elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PLOAD2	LID	P	EID1	EID2	EID3	EID4	EID5	EID6	
			EIDF	"THRU"	EIDL				
PLOAD2	156	98.2	101	432	657				

Alternate Range Form:

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
P	Pressure value. [2,3]	Real	Required
EIDi	Element identification numbers. [4]	Integer>0	Required
EIDF, EIDL	First and last element identification numbers specifying a range. [4]	Integer>0 **EIDF<EIDL**	Required

Remarks:

1. Load sets must be selected in the Case Control Packet with the command:

   LOAD = LID

2. The pressure intensity is the load per unit surface area.
3. The direction of the pressure is computed according to the right-hand rule using the GRID point sequence specified on the element connection entry. If the surface of an element is curved, the direction of the pressure may vary over the surface. Refer to **PLOAD4** for a more general pressure load capability.
4. The pressure is applied to all existing plate elements with specified identification numbers.
5. Equivalent GRID point loads are computed which depend on the specific element geometry and type. A uniform pressure may not result in equal GRID point loads.

7 BULK

Bulk Data Entry PLOAD4 (PLATE ELEMENTS)

Defines a load on the surface of a TRIA3, TRIA6, TRIAR, QUAD4, QUAD8, or QUADR element.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
PLOAD4	LID	EID EIDF	P1	P2	P3	P4	"THRU"	EIDL	-cont-
-cont-	CID	V1	V2	V3					

PLOAD4	101	2043	15.	18.	23.6	12.4			+A
+A	52	1.0	0.0	0.0					

PLOAD4	1	101	10.	10.	20.	20.	THRU	201	

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
EID	Element identification number. [2]	Integer>0	Required
EIDF, EIDL	First and last element identification numbers in a range definition. [2]	Integer>0 **EIDF<EIDL**	Required
Pi	Pressure at the GRID points defining the element surface. [3,4,5]	Real	**0.0**
CID	Coordinate system identification number. [4,5]	Integer>0	**0**
Vi	Components of a vector in system **CID** that defines the direction of the GRID point loads generated by the pressure. [4,5]	Real	**0.0**

Remarks:

1. Load sets are selected in the Case Control packet with the command:

   LOAD = LID

2. The pressure is applied to all existing plate elements with specified identification numbers.
3. If only **P1** is given, the pressure is assumed to be uniform over the element surface. The **P4** value is ignored for a triangular face. The pressure intensity is the load per unit surface area.
4. If a direction vector is not specified, the direction of the GRID point loads is normal to the element mid-surface at each GRID point in the local +z direction. If the surface of the element is curved, the direction of pressure may vary from point to point.
5. When the direction vector is defined and a value for **CID** is not entered, the GRID point load vectors are applied in the Basic Coordinate System.
6. Equivalent GRID point loads are computed which depend on the specific element geometry and type. A uniform pressure may not result in equal GRID point loads.

Bulk Data Entry PLOAD4 (HEXA SOLID ELEMENTS)

Defines a load on a face of a HEXA solid element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PLOAD4	LID	EID	P1	P2	P3	P4	GIDD1	GIDD2	-cont-
-cont-	CID	V1	V2	V3					

PLOAD4	1	101	50.	50.	50.	50	201	204	+A
+A	101	1.0	0.0	0.0					

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
EID	Element identification number. [2]	Integer>0	Required
Pi	Pressure at the GRID points defining the loaded element face. [3,4,5]	Real	**0.0**
GIDDi	Identification numbers of GRID points defining a diagonal of the loaded face of the element.	Integer>0	Required
CID	Coordinate system identification number. [4,5]	Integer>0	**0**
Vi	Components of a vector in system **CID** that defines the direction of the GRID point loads generated by the pressure. [4,5]	Real	**0.0**

Remarks:

1. Load sets are selected in the Case Control packet with the command:

   LOAD = LID

2. The pressure is applied to all existing HEXA elements with specified identification numbers.
3. If only **P1** is given, the pressure is assumed to be uniform over the loaded element face. The pressure intensity is the load per unit surface area.
4. If a direction vector is not specified, the direction of the GRID point loads is normal to the loaded face at each GRID point and inward. If the loaded face is curved, the direction of pressure may vary from point to point.
5. When the direction vector is defined and a value for **CID** is not entered, the GRID point load vectors are applied in the Basic Coordinate System.
6. Equivalent GRID point loads are computed which depend on the specific element geometry and type. A uniform pressure may not result in equal GRID point loads.

7 BULK

Bulk Data Entry PLOAD4 (PENTA SOLID ELEMENTS)

Defines a load on a face of a PENTA solid element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PLOAD4	LID	EID	P1	P2	P3	P4	GIDD1	GIDD2	-cont-
-cont-	CID	V1	V2	V3					

PLOAD4	1000	999	100.0	200.	100.		902		+A
+A	1	0.0	0.0	1.0					

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
EID	Element identification number. [2]	Integer>0	Required
Pi	Pressure at the GRID points defining the element face. [3,5,6]	Real	**0.0**
GIDDi	Identification number(s) of GRID point(s) uniquely defining the loaded face of the element. [4]	Integer>0	Required
CID	Coordinate system identification number. [5,6]	Integer>0	**0**
Vi	Components of a vector in system **CID** that defines the direction of the GRID point loads generated by the pressure. [5,6]	Real	**0.0**

Remarks:

1. Load sets are selected in the Case Control packet with the command:

   LOAD = LID

2. The pressure is applied to all existing PENTA elements with specified identification numbers.
3. If only **P1** is given, the pressure is assumed to be uniform over the loaded element face. The **P4** value is ignored for a triangular face. The pressure intensity is the load per unit surface area.
4. Both **GIDD1** and **GIDD2** are required to specify a quadrilateral face, but only **GIDD1** is required when applying the load to a triangular face. The value of **GIDD1** in this case may be any of the GRID points on the face, and **GIDD2** must be left blank.
5. If a direction vector is not specified, the direction of the GRID point loads is normal to the loaded face at each GRID point and inward. If the loaded face is curved, the direction of pressure may vary from point to point.
6. When the direction vector is defined and a value for **CID** is not entered, the GRID point load vectors are applied in the Basic Coordinate System.
7. Equivalent GRID point loads are computed which depend on the specific element geometry and type. A uniform pressure may not result in equal GRID point loads.

Bulk Data Entry PLOAD4 (TETRA SOLID ELEMENTS)

Defines a load on a face of a TETRA solid element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PLOAD4	LID	EID	P1	P2	P3		GIDON	GIDOPP	-cont-
-cont-	CID	V1	V2	V3					

PLOAD4	1	101	10.0	15.5	20.5		1	3	+A
+A	101	1.0	0.0	0.0					

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
EID	Element identification number. [2]	Integer>0	Required
Pi	Pressure at the GRID points defining the loaded element face. [3]	Real	**0.0**
GIDON	Identification number of a GRID point on the loaded face of the element.	Integer>0	Required
GIDOPP	Identification number of the GRID point **opposite** to the loaded face of the element.	Integer>0	Required
CID	Coordinate system identification number. [4,5]	Integer>0	**0**
Vi	Components of a vector in system **CID** that defines the direction of the GRID point loads generated by the pressure. [4,5]	Real	**0.0**

Remarks:

1. Load sets are selected in the Case Control packet with the command:

   LOAD = LID

2. The pressure is applied to all existing TETRA elements with specified identification numbers.
3. If only **P1** is given, the pressure is assumed to be uniform over the loaded element face. The pressure intensity is the load per unit surface area.
4. If a direction vector is not specified, the direction of the GRID point loads is normal to the loaded face at each GRID point and inward. If the loaded face is curved, the direction of pressure may vary from point to point.
5. When the direction vector is defined and a value for **CID** is not entered, the GRID point load vectors are applied in the Basic Coordinate System.
6. Equivalent GRID point loads are computed which depend on the specific element geometry and type. A uniform pressure may not result in equal GRID point loads.

Bulk Data Entry PLOTEL

Defines a line element used only for plotting. This element is not used in any manner to affect analysis results.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PLOTEL	EID	GID1	GID2						

PLOTEL	29	101	102						

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GID1i	GRID point identification numbers.	Integer>0	Required

Remarks:

1. Element identification numbers must be unique.

7 BULK

Bulk Data Entry PMASS

Defines the value of mass elements MASS1 and MASS3.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PMASS	PID	M							
PMASS	101	8.72							

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
M	Mass value. [1]	Real	Required

Remarks:

1. Negative masses, although allowed, should be used with great caution.
2. Masses may be defined directly with **CMASS2** and **CMASS4** Bulk Data entries.

7 BULK

Bulk Data Entry POINTAX

Defines the location of a discrete point on an axisymmetric ring. These points may be used to obtain the motions of discrete points or to apply discrete point loads.

Format and Example:

1	2	3	4	5	6	7	8	9	10
POINTAX	GID	RID	ϕ						

POINTAX	123	103	35.0						

Field	*Contents*	*Data Range*	*Default*
GID	Point identification number. [1]	Integer>0	Required
RID	Identification number of a **RINGAX** entry.	Integer>0	Required
ϕ	Azimuthal angle in degrees.	Real	**0.0**

Remarks:

1. **GID** must be unique with respect to all other point identification numbers.
2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.
3. Points defined by **POINTAX** entries may not appear in the constraint entries **MPCAX**, **SPCAX** or **OMITAX**.

7 BULK

Bulk Data Entry PPILE

Defines the properties of a tubular PILE element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PPILE	PID	MID	A	T	D	NSM			
PPILE	10	10		1.0	24.0				

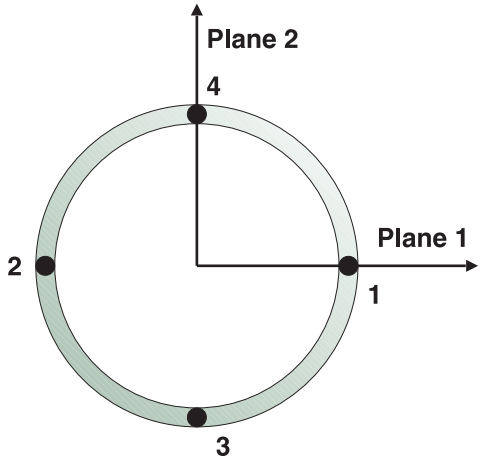
Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
A	Area of the PILE cross-section.	Real>0.0	[2]
T	Wall thickness of the PILE.	Real>0.0	[2]
D	Outside diameter of the PILE.	Real>0.0	[2]
NSM	Nonstructural mass per unit length.	Real≥0.0	**0.0**

Remarks:

1. For structural problems, **PPILE** entries may only reference **MAT1**, **MAT1NL**, or **MAT1NL1** material data.
2. One of the **A**, **T** or **D** fields must be blank. The missing value is calculated from the relationship:

$$\mathbf{A} = \pi \mathbf{T} (\mathbf{D} - \mathbf{T})$$

3. Stress data will be recovered at the outside diameter of the PILE in Plane 1 and Plane 2 as illustrated in the figure below.



7 BULK

Bulk Data Entry PPILE1

Defines the properties of a general PILE element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PPILE1	PID	MID	A	I1	I2	J	NSM	R	-cont-
-cont-	Y1	Z1	Y2	Z2	Y3	Z3	Y4	Z4	-cont-
-cont-	K1	K2	I12	B1	B2	C	SHAPE		

PPILE1	101	56	2.9		8.76				+A
+A			2.0	4.0					+B
+B									

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
A	Area of PILE cross-section.	Real≥0.0	Required
I1,I2,I12	Area moments of inertia.	Real≥0.0 $I_1 I_2 > I_{12}^2$	**0.0**
J	Torsional constant.	Real≥0.0	**0.0**
NSM	Nonstructural mass per unit length.	Real≥0.0	**0.0**
R	Radius of gyration.	Real≥0.0	**0.0**
Yi,Zi	Coordinates of stress recovery points on the element cross- section in the element coordinate system.	Real	**0.0**
K1,K2	Area factors for shear. [2]	1.0≥Real≥0.0	**0.0**
B1,B2	Effective width of the PILE in the lateral direction of Plane 1, Plane 2.	Real≥0.0	**0.0**
C	Effective circumference of the PILE for torsional friction.	Real≥0.0	**0.0**
SHAPE	Cross-sectional shape for nonlinear correction. [3]	Character { **BAR** / **ROD** / **TUBE** }	**BAR**

7 BULK

Remarks:

1. For structural problems, **PPILE1** entries may only reference **MAT1**, **MAT1NL**, or **MAT1NL1** material data.
2. If area factors for shear, **K1** and **K2**, are not entered, then the transverse shear flexibilities are assumed to be infinite. Positive values smaller than 1.0 mean that all or part of the PILE cross-sectional area is effective for flexibility. Factors of 0.833 are accepted standards for shear flexibilty of a rectangular cross section bar, and 0.50 are appropriate for a rod shape.
3. Nonlinear material behavior is corrected for basic geometric cross-sectional shapes. The three shapes that may be selected are shown in the figures for the **PBAR1** and **PBEAM1** Bulk Data entries.

Bulk Data Entry PPIPE

Defines the properties of a circular cross-section PIPE element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PPIPE	PID	MID	OD	T			NSM	P	-cont-
-cont-	C1	C2	D1	D2	E1	E2	F1	F2	

PPIPE	112	2.5	0.5					10.	+A
+A	0.	0.5	-.5	0.	0.	-.5	.5	0.	

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number of a **MAT1** entry.	Integer>0	Required
OD	Outside diameter of pipe cross-section.	Real>0.0	Required
T	Thickness of pipe cross-section.	Real <(**OD**/2)	[1]
NSM	Nonstructural mass per unit length.	Real≥0.0	**0.0**
P	Internal pressure.	Real	**0.0**
Ci,Di, Ei,Fi	Stress recovery coordinates in the element coordinate system. [2]	Real	[2]

Remarks:

1. If **T** is zero or blank, a solid circular shaft is assumed.
2. If all of the stress recovery coefficients are not specified, the values (relative to the **OD** entry) shown in the example are used for each of the fields.

7 BULK

Bulk Data Entry PRESAX

Defines the static pressure loading for an axisymmetric harmonic model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PRESAX	LID	P	RID1	RID2	ϕ1	ϕ2			
PRESAX	104	23.5	4	3	26.5	36.7			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
P	Pressure value.	Real	Required
RIDi	Ring identification numbers.	Integer>0	Required
fi	Azimuthal angles in degrees.	Real	**0.0**

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

7 BULK

Bulk Data Entry PRESPT

Defines the location of PRESSURE points in the fluid for recovery of pressure data.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PRESPT	FID		IDP1	$\phi 1$	IDP2	$\phi 2$	IDP3	$\phi 3$	
PRESPT	115		242	0.0			243	90.0	

Field	*Contents*	*Data Range*	*Default*
FID	Identification number of a fluid point defined by a **RINGFL** Bulk Data entry.	Integer>0	Required
IDPi	Pressure point identification number. [1,2]	Integer>0	Required
fi	Azimuthal position on fluid point, referenced by **FID**, in the fluid coordinate system.	Real	**0.0**

Remarks:

1. All PRESSURE point identification numbers must be unique with respect to other GRID, SCALAR and PRESSURE points.
2. The PRESSURE points are used primarily for the identification of output data. They may also be used as points at which to measure pressure for input to control devices.
3. This entry is allowed only if an **AXIF** entry is also present in the Bulk Data packet.
4. Output requests for velocity and acceleration of these degrees of freedom will result in derivatives of pressure with respect to time.

Bulk Data Entry PROD

Defines the properties of a ROD element.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
PROD	PID	MID	A	J	C	NSM			
PROD	118	101	5.4	34.23	2.5	0.5			

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1,2]	Integer>0	Required
A	Area of rod.	Real≥0.0	Required
J	Torsional constant.	Real≥0.0	**0.0**
C	Coefficient to determine torsional stress.	Real	**0.0**
NSM	Nonstructural mass per unit length.	Real≥0.0	**0.0**

Remarks:

1. For structural problems, **PROD** entries may only reference **MAT1** data.
2. For heat transfer problems, **PROD** entries may only reference **MAT4** or **MAT5** data.

7 BULK

Bulk Data Entry PSHEAR

Defines the elastic properties of a SHEAR panel element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PSHEAR	PID	MID	T	NSM	F1	F2			
PSHEAR	112	5	0.054	0.002	1.0	30.0			

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
T	Thickness of shear panel.	Real≥0.0	Required
NSM	Nonstructural mass per unit area.	Real≥0.0	**0.0**
F1	Area factor for extensional stiffness along sides 1-2 and 3-4. [2,3]	Real≥0.0	**0.0**
F2	Area factor for extensional stiffness along sides 2-3 and 1-4. [2,3]	Real≥0.0	**0.0**

Remarks:

1. **PSHEAR** entries may only reference **MAT1** data.
2. Poisson's ratio coupling for extensional effects is ignored.
3. The effective extensional area, A, of each of the equivalent rods along element edges is calculated as follows:

   For sides 1-2 and 4-3,

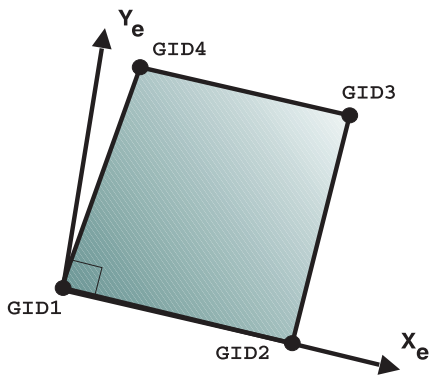
   $$A = \mathbf{F1}\,\frac{\mathbf{T}\,\overline{w}_x}{2} \quad \text{when} \quad \mathbf{F1} \le 1.01$$

   $$A = \mathbf{F1}\,\frac{\mathbf{T^2}}{2} \quad \text{when} \quad \mathbf{F1} > 1.01$$

   For sides 1-4 and 2-3,

   $$A = \mathbf{F2}\,\frac{\mathbf{T}\,\overline{w}_y}{2} \quad \text{when} \quad \mathbf{F2} \le 1.01$$

   $$A = \mathbf{F2}\,\frac{\mathbf{T^2}}{\mathbf{2}} \quad \text{when} \quad \mathbf{F2} > 1.01$$

   where $\overline{w}_x$ and $\overline{w}_y$ are the average width of the panel in directions x and y, respectively.

7 BULK

Bulk Data Entry PSHELL

Defines the membrane, bending, transverse shear, and coupling properties of the plate and shell elements, QUAD4, QUAD8, TRIA3 and TRIA6.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PSHELL	PID	MID1	T	MID2	12I/T3	MID3	TS/T	NSM	-cont-
-cont-	Z1	Z2	MID4	MCSID	SCSID	ZOFF		NSP	-cont-
				MCTYPE	SCTYPE				
				θm	θs				
-cont-	PLSTR								

PSHELL	203	204	1.90	205	1.2	206	0.8	6.32	+A
+A	+.95	-.95		0	0	0.01		5	+B
+B	STRESS								

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID1	Material identification number for membrane. [1,2,3,4]	Integer>0	**0**
T	Membrane thickness. [2,5]	Real	**0.0**
MID2	Material identification number for bending. [1,3]	Integer>0	**0**
12I/T3	Bending stiffness parameter.	Real	**1.0**
MID3	Material identification number for transverse shear. [1,3,6]	Integer>0	**0**
TS/T	Transverse shear thickness divided by membrane thickness.	Real	**0.833333**
NSM	Nonstructural mass per unit area.	Real≥0.0	**0.0**
Z1,Z2	Fiber distances for stress computation. The positive direction is determined by the righthand rule and the order in which the GRID points are listed on the connection entry.	Real	**Z1=-T/2** **Z2 = T/2**
MID4	Material identification number for membrane-bending coupling. [1,3,7]	Integer>0 ≠**MID1**≠**MID2**	**0**
MCSID	Identification number of material coordinate system.	Integer≥0	[8]
MCTYPE	Material coordinate system selector.	Character { **ELEMENT** POST BASIC }	[8]
θm	Material angle. [8]	Real	[8]

7 BULK

SCSID	Identification number of stress coordinate system.	Integer≥0	[9]
SCTYPE	Material coordinate system selector.	Character { MATERIAL / **ELEMENT** / POST / BASIC }	[9]
θ_s	Stress angle. [9]	Real	[9]
ZOFF	Offset of the element reference plane from the plane of GRID points. A positive value means the $+Z_e$ direction. [10]	Real	**0.0**
NSP	Number of sample points through the thickness of the shell for nonlinear analysis.	{ **1** / **5** / **7** / **9** }	[11]
PLSTR	Selects the plane stress or plane strain option.	{ **STRESS** / **STRAIN** }	**STRESS**

Remarks :

1. The results of leaving any **MID** field blank are:
   - **MID1** No membrane or coupling stiffness.
   - **MID2** No bending, coupling, or transverse shear stiffness.
   - **MID3** No transverse shear flexibility.
   - **MID4** No membrane-bending coupling.
2. The structural mass is computed from the density using the membrane thickness and membrane material properties.
3. For structural problems, **PSHELL** entries may reference **MAT1**, **MAT2** or **MAT8** material property data, and for heat transfer problems, **PSHELL** entries may reference **MAT4** or **MAT5** material property data.
4. Structural damping, when needed, is obtained from the **MID1** material.
5. **T** replaces any undefined membrane thicknesses at the GRID points defining the element.
6. If the transverse shear material, **MID3**, references **MAT2** data, then **G33** must be zero. If it references **MAT8** data, then **G1Z** and **G2Z** must not be zero.
7. The **MID4** field should be left blank if the material properties are symmetric with respect to the mid-surface of the element.
8. There are three ways in which you may define the element material coordinate system:

Field 5	RESULTING MATERIAL SYSTEM
MCSID	Selects a coordinate system identification number which gives the orientation of the material x-axis is along the projection of the x-axis of the specified coordinate system onto the xy plane of the element coordinate system
θ_m	Allows you to specify a real value, θ_m, which is the angle of rotation of the x-axis of the material coordinate system with respect to the x-axis of the element coordinate system in the xy plane of the latter.
{ **ELEMENT** / POST / BASIC }	Allows you to select the **ELEMENT** coordinate system, the **BASIC** coordinate system, or the coordinate system used by other NASTRAN varients, **POST**.

7 BULK

All **CQUAD4, CQUAD8, CTRIA3**, and **CTRIA6** Bulk Data entries which reference this property and do not have a material coordinate system specified will use the value specified here.

If Field 5 is left blank, the **ELEMENT** coordinate system is used for the material coordinate system.

9. There are also three ways in which you may define the element stress coordinate system:

Field 6	RESULTING STRESS SYSTEM
SCSID	Selects a coordinate system identification number which gives the orientation of the stress x-axis is along the projection of the x-axis of the specified coordinate system onto the xy plane of the element coordinate system
θs	Allows you to specify a real value, **θS**, which is the angle of rotation of the x-axis of the stress coordinate system with respect to the x-axis of the element coordinate system in the xy plane of the latter.
SCTYPE	Character string which allows you to select the **MATERIAL** coordinate system, the **ELEMENT** coordinate system, the **BASIC** coordinate system, or the coordinate system used by other NASTRAN varients, **POST**.

All **CQUAD4, CQUAD8, CTRIA3**, and **CTRIA6** Bulk Data entries which reference this property and do not have a stress coordinate system specified will use the value specified here.

If Field 6 is left blank, the default value which is usually specified in the Preference File is used. If such a value does not appear in the Preference File, then the default value is the **ELEMENT** coordinate system.

10. The value of **ZOFF** is the default value for the corresponding field on **CQUAD4**, **CQUAD8**, **CTRIA3**, and **CTRIA6** Bulk Data entries. The Z_e direction is along the out-of-plane axis of the element coordinate system.

11. **NSP** is **1** if the element is membrane only, otherwise it is **7**.

Bulk Data Entry PSOIL

Defines linear and nonlinear properties of soil attached to a PILE element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PSOIL	PID	TIDA	TIDJ	TIDL					
		KA	KJ	KL					
PSOIL	101	1	2	1.+6					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
TIDA	Identification number of a **TABLENL** entry that defines soil properties in the axial direction of the PILE. [1,2]	Integer>0	**0**
KA	Constant axial stiffness. [3]	Real	**0.0**
TIDJ	Identification number of a **TABLENL** entry that defines soil properties in the torsional direction of the PILE. [1,2]	Integer>0	**0**
KJ	Constant torsional stiffness. [3]	Real	**0.0**
TIDL	Identification number of a **TABLENL** entry that defines soil properties in the lateral direction of the PILE. [1,2]	Integer>0	**0**
KL	Constant lateral stiffness. [3]	Real	**0.0**

Remarks:

1. Soil properties are defined by force-deflection curves that are specified with **TABLENL** Bulk Data entries. Stiffness is determined from those curves. The **TIDA** and **TIDJ** data should be given as force per unit circumference of the PILE, and the **TIDL** data as force per unit width of the PILE.
2. A blank field or 0 value for any of **TIDA**, **TIDL** or **TIDJ** indicates that there is no soil stiffness in that direction.
3. If constant stiffness values are used, the element behaves linearly in the directions of constant stiffness.

7 BULK

Bulk Data Entry **PSOLID**

Defines the properties of the solid hexahedral element, HEXA, pentahedral element, PENTA, and tetrahedral element, TETRA.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PSOLID	PID	MATID	MCSID MCTYPE	INTOP	SCSID SCTYPE				
PSOLID	101	1	901		54				

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MATID	Material identification number. [1,2,3]	Integer>0	Required
MCSID	Identification number of material coordinate system.	Integer≥0	[4]
MCTYPE	Material coordinate system selector.	Character { **BASIC** | ELEMENT }	[4]
INTOP	Selects the numerical integration option. [5]	{ BUBBLE or 0 | REDUCE or 1 | TWO or 2 | THREE or 3 }	[6]
SCSID	Identification number of stress coordinate system.	Integer≥0	[7]
SCTYPE	Stress coordinate system selector.	Character { **MATERIAL** | ELEMENT | BASIC }	[7]

Remarks:

1. For structural problems, **PSOLID** entries may only reference **MAT1** or **MAT9** material property data.
2. For heat transfer problems, **PSOLID** entries may only reference **MAT4** or **MAT5** material property data.
3. For fluid elements, the **PSOLID** entry must reference a **MATF** material property Bulk Data Entry.
4. If Field 4 is blank, the default value from the Preference File is used. If the Preference File does not contain a default value, then the default is the **BASIC** system. You may also select the **ELEMENT** coordinate system or an arbitrary coordinate system defined by the integer identification number, **MCSID**.

7 BULK

5. Explicit control of the numerical itegration method may be useful in certain specific cases.

   The 3x3x3 isoparametric integration scheme is recommended when performing Heat Transfer with solid elements with midside nodes. Default approach for Heat Transfer and Fluid elements

   The 3x3x3 isoparametric integration scheme satisfies the patch test for the 20-node HEXA, 15-node PENTA and 10-node TETRA.

   The 2x2x2 isoparametric integration scheme is recommended when performing Heat Transfer with solid elements without midside nodes. Default approach

   For most engineering applications, the program automatically determines the optimal **REDUCE**d integration scheme except in the case of eight-nodes HEXA and six-noded PENTA elements which use **BUBBLE** functions. The other integration options, **TWO** and **THREE** select the use of 2x2x2 and 3x3x3 isoparametric integration schemes, respectively. These latter options may provide better answers for certain cases when performing Heat Transfer analysis.

6. The default value is **BUBBLE** for eight-nodes HEXA and six-noded PENTA elements and **REDUCE** for all others for structural elements. For Heat Transfer and Fluid elements, the default is described in the previous note.

7. If Field 6 is left blank, the default value which is usually specified in the Preference File is used. If such a value does not appear in the Preference File, then the default value is the **MATERIAL** system. You may also select the **BASIC** coordinate system, the **ELEMENT** coordinate system, or an arbitrary coordinate system defined by the integer identification number, **SCSID**.

8. You may provide a default value for **INTOP** in the Preference File. The default is found in the **<Solution Techniques>** group of the **[UAI/NASTRAN]** Configuration Section. You can print the contents of the Preference File with the command:

   ```
   PRINT PREFERENCES
   ```

   Your System Support Specialist can provide you with more information about this advanced feature.

Bulk Data Entry PTORDRG

Used to define membrane and bending properties of the toroidal ring, TORDRG, element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PTORDRG	PID	MID	TM	TF					
PTORDRG	105	201	0.2	0.15					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
TM	Thickness for membrane.	Real>0.0	Required
TF	Thickness for flexure.	Real>0.0	Required

Remarks:

1. **PTORDRG** entries may only reference **MAT1** or **MAT3** data.

Bulk Data Entry PTRAPAX

Defines the properties of a TRAPAX axisymmetric trapezoidal cross-section ring element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PTRAPAX	PID		MID	ϕ1	ϕ2	ϕ3	ϕ4	ϕ5	-cont-
-cont-	ϕ6	ϕ7	ϕ8	ϕ9	ϕ10	ϕ11	ϕ12	ϕ13	-cont-
-cont-	ϕ14								

PTRAPAX	101		201	0.0	5.0	6.0	7.0	8.0	+A
+A	9.0	10.0	15.0	20.0	25.0	30.0	35.0	40.0	+B
+B	45.0								

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
fi	Azimuthal coordinates (in degrees) for stress recovery.	Real	**0.0**

Remarks:

1. **PTRAPAX** entries may only reference **MAT1** or **MAT3** data.
2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

7 BULK

Bulk Data Entry PTRIAAX

Defines the properties of a TRIAAX axisymmetric trapezoidal cross-section ring element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PTRIAAX	PID		MID	ϕ1	ϕ2	ϕ3	ϕ4	ϕ5	-cont-
-cont-	ϕ6	ϕ7	ϕ8	ϕ9	ϕ10	ϕ11	ϕ12	ϕ13	-cont-
-cont-	ϕ14								

PTRIAAX	101		201	0.0	5.0	6.0	7.0	8.0	+A
+A	9.0	10.0	15.0	20.0	25.0	30.0	35.0	40.0	+B
+B	45.0								

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
fi	Azimuthal coordinates (in degrees) for stress recovery.	Real	**0.0**

Remarks:

1. **PTRIAAX** entries may only reference **MAT1** or **MAT3** material data.
2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

7 BULK

Bulk Data Entry PTUBE

Defines the properties of a TUBE cylindrical tube element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PTUBE	PID	MID	OD	T	NSM				
PTUBE	101	9001	5.32	0.75					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1,2]	Integer>0	Required
OD	Outside diameter of the tube.	Real>0.0	Required
T	Wall thickness. [3]	0.0≤Real≤½**OD**	Required
NSM	Nonstructural mass per unit area.	Real≥0.0	**0.0**

Remarks:

1. For structural problems, **PTUBE** entries may only reference **MAT1** material data.
2. For heat transfer problems, **PTUBE** entries may only reference **MAT4** or **MAT5** material data.
3. If **T** is zero, the tube will be a solid circular rod.

7 BULK

Bulk Data Entry PTWIST

Defines the elastic properties of a TWIST panel element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PTWIST	PID	MID	T	NSM					
PTWIST	101	9002	1.3	5.5					

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
MID	Material identification number. [1]	Integer>0	Required
T	Thickness of twist panel.	Real>0.0	Required
NSM	Nonstructural mass per unit area.	Real≥0.0	**0.0**

Remarks:

1. **PTWIST** entries may only reference **MAT1** material data.

7 BULK

Bulk Data Entry PVISC

Defines the viscous properties of a one-dimensional viscous element, VISC.

Format and Example:

1	2	3	4	5	6	7	8	9	10
PVISC	PID	B1	B2						
PVISC	104	2.1	6.54						

Field	*Contents*	*Data Range*	*Default*
PID	Property identification number.	Integer>0	Required
B1,B2	Viscous coefficients for extension and rotation.	Real	Required

Remarks:

1. Viscous properties are material and temperature independent.

7 BULK

Bulk Data Entry QBDY1

Defines a uniform heat flux into an HBDY element.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
QBDY1	LID	Q0	*ELEMENT ID LIST*						-cont-

QBDY1	345	1.-5	1	3	10	9	6	5	+A
+A	7	8							

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
Q0	Heat flux into element. [2]	Real	Required
ELEMENT ID LIST	HBDY element identification numbers.	Integer>0	Required

Remarks:

1. For statics, the load set is selected in the Case Control packet with the command:

   LOAD = LID

   For dynamics, the load **LID** is selected by reference using **TLOAD** data. The load at each point will be multiplied by the function of time defined on the **TLOAD** entry. The **TLOADi** entry is selected with the Case Control command:

   DLOAD = LID

2. **Q0** is positive for heat flux into the element.

7 BULK

Bulk Data Entry QBDY2

Defines GRID point flux into an HBDY element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
QBDY2	LID	EID	Q01	Q02	Q03	Q04			
QBDY2	210	812	1.-5	1.1-5	4.2-6	3.2-4			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
EID	Identification number of an HBDY element.	Integer>0	Required
Q0i	Heat flux at GRID point **i** of the referenced HBDY element. [2]	Real	Required

Remarks:

1. For statics, the load set is selected in the Case Control packet with the command:

   LOAD = LID

   For dynamics, the load **LID** is selected by reference using **TLOAD** data. The load at each point will be multiplied by the function of time defined on the **TLOAD** entry. The **TLOADi** entry is selected with the Case Control command:

   DLOAD = LID

2. Positive **Q0i** indicate heat flux into the element.

7 BULK

Bulk Data Entry QHBDY

Defines a uniform heat flux into a set of GRID points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
QHBDY	LID	TYPE	Q0	AF	GID1	GID2	GID3	GID4	
QHBDY	321	LINE	2.3+2	.25	114	115			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
TYPE	Type of area involved. [2]	Character { **POINT** **LINE** **REV** **AREA3** **AREA4** }	Required
Q0	Heat flux into element.	Real	Required
AF	Area factor which depends on **TYPE**. [3]	Real	**0.0**
GIDi	GRID point identification numbers.	Integer>0	Required

Remarks:

1. For statics, the load set is selected in the Case Control packet with the command:

   LOAD = LID

   For dynamics, the load **LID** is selected by reference using **TLOAD** data. The load at each point will be multiplied by the function of time defined on the **TLOAD** entry. The **TLOADi** entry is selected with the Case Control command:

   DLOAD = LID

2. The **TYPE** defines a surface in the same manner as the HBDY element physical descriptions of the geometry involved, see the **CHBDY** description.

7 BULK

3. The number of connecting GRID points and the meaning of the area factor **AF** for the five types are:

AREA TYPE	NUMBER OF GRID POINTS	AREA FACTOR AF
POINT	1	Effective area
LINE	2	Effective width
REV	2	Not used
AREA3	3	Not used
AREA4	4	Not used

4. The heat flux applied to the area is transformed to loads on the points. These points need not correspond to an HBDY element.

Bulk Data Entry QVECT

Defines thermal vector flux from a distant source into HBDY elements.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
QVECT	LID	Q0	V1	V2	V3	*ELEMENT ID LIST*			

QVECT	101	3.-4	-1.0	0.0	0.0	11	21	107	+A
+A	25	301	THRU	308					

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
Q0	Magnitude of thermal flux vector.	Real	Required
Vi	Vector components (in Basic Coordinate System) of the thermal vector flux. [2,3]	Real	**0.0**
ELEMENT ID LIST	Identification numbers of HBDY elements irradiated by the distant source.	Integer>0	Required

Remarks:

1. For statics, the load set is selected in the Case Control packet with the command:

   LOAD = LID

   For dynamics, the load is selected by reference using **TLOAD** data. The load at each point will be multiplied by the function of time defined on the **TLOAD** entry.

   For dynamics, the load **LID** is selected by reference using **TLOAD** data. The load at each point will be multiplied by the function of time defined on the **TLOAD** entry. The **TLOADi** entry is selected with the Case Control command:

   DLOAD = LID

2. If the referenced HBDY element has a **TYPE** of **REV**, the vector should be parallel to the Basic Coordinate System z-axis.
3. At least one component of **V** must be non-zero.

7 BULK

Bulk Data Entry QVOL

Defines a rate of internal heat generation in an element.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
QVOL	LID	QV	*ELEMENT ID LIST*						-cont-

QVOL	345	1.-5	1	3	10	9	6	5	+A
+A	7	8	1001	THRU	1005				

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
QV	Power input per unit volume produced by a heat conduction element.	Real	Required
ELEMENT ID LIST	Heat conduction element identification numbers.	Integer>0	Required

Remarks:

1. For statics, the load set is selected in the Case Control packet with the command:

   LOAD = LID

   For dynamics, the load **LID** is selected by reference using **TLOAD** data. The load at each point will be multiplied by the function of time defined on the **TLOAD** entry. The **TLOADi** entry is selected with the Case Control command:

   DLOAD = LID

7 BULK

Bulk Data Entry RADLST

Defines the relation of the columns of the radiation matrix **RADMTX** to specific HBDY elements in Nonlinear Steady State and Transient Heat Transfer analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RADLST	*ELEMENT ID LIST CONTINUES*								-cont-

RADLST	10	20	30	50	31	101	THRU	201	+A
+A	307	444							

Field	*Contents*	*Data Range*	*Default*
ELEMENT ID LIST	Element identification numbers of HBDY elements, given in the order that they appear in the **RADMTX** matrix. [1]	Integer>0	Required

Remarks:

1. An element may be listed more than once. For instance, if both sides of a panel are radiating, each side may participate in a different part of the view factor matrix.
2. Only one **RADLST** entry is allowed in the Bulk Data packet.
3. This entry is required only if a radiation matrix is defined.
4. This entry is *not* required when using the automatic view factor calculation capability of **UAI/*NASTRAN***. The automatic view factor calculations are controlled by **VIEW** and **VIEWOP** Bulk Data entries.

7 BULK

Bulk Data Entry RADMTX

Defines the matrix of radiation exchange coefficients in nonlinear steady state and transient heat transfer analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RADMTX	COLUMN	R1	R2	R3	R4	R5	R6	R7	-cont-
-cont-	R8	R9	*CONTINUES WITH LIST OF VALUES*						-cont-

RADMTX	3	1.0	2.6	1.5	7.3	9.8	11.2	3.1	+A
+A	6.5								

Field	*Contents*	*Data Range*	*Default*
COLUMN	The matrix column number.	Integer>0	Required
Ri	Matrix terms. [1,2]	Real	[2]

Remarks:

1. The **RADMTX** has one column for each radiating area.
2. The radiation exchange coefficient matrix is symmetric, and only the lower triangle is input. Data entry begins with the diagonal term and continues down the column. The column number associates the column with the HBDY element listed in the corresponding position on the **RADLST** entry. It is not required to enter null columns.
3. Each **RADMTX** logical entry is terminated by one or more blank **Ri** at the end of the last physical entry. All rows not entered are set to zero.
4. This entry is *not* required when using the automatic view factor calculation capability of **UAI/*NASTRAN***. The automatic view factor calculations are controlled by **VIEW** and **VIEWOP** Bulk Data entries.

Bulk Data Entry RANDPS

Defines load set power spectral density factors for use in random analysis having the frequency dependent form:

$$S_{jk}(f) = (\mathbf{X} + i\mathbf{Y})\, G(f)$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
RANDPS	SID	SUBJ	SUBK	X	Y	TID			
RANDPS	301	2	6	2.5	7.6	205			

Field	*Contents*	*Data Range*	*Default*
SID	Random analysis set identification number. [1]	Integer>0	Required
SUBJ	Subcase identification number of excited load set. [2]	Integer>0	Required
SUBK	Subcase identification number of applied load set. [2]	Integer>0 **SUBK≥SUBJ**	Required
X,Y	Components of the complex coefficient. [2]	Real	Required
TID	Identification number of a **TABRND1** entry which defines *G(f)*. [3]	Integer≥0	**0**

Remarks:

1. Power spectral density loads must be selected in the Case Control packet with the command:

   RANDOM = SID

2. If **SUBJ** and **SUBK** reference the same subcase, then **Y** must be zero.
3. If **TID** is zero or blank, then *G(f)* = 1.0.
4. Only 20 unique **SID** sets may be defined. However, as many **RANDPS** entries as desired with the same **SID** may be input.
5. Random response analysis is available only in the direct and modal approach frequency response Rigid Formats. Random analysis responses may only be output by using the XY-plotter post-processor.
6. No change in either frequency sets, SPC sets, or direct input matrices is allowed between subcases which are used for random response.

7 BULK

Bulk Data Entry RANDT1

Defines time lag constants for use in random analysis autocorrelation function computations.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RANDT1	SID	N	T0	TMAX					
RANDT1	102	12	2.5	14.5					

Field	*Contents*	*Data Range*	*Default*
SID	Random analysis set identification number. [1,2]	Integer>0	Required
N	Number of time lag intervals. [3]	Integer>0	Required
T0	Starting time lag.	Real≥0.0	**0.0**
TMAX	Maximum time lag.	Real>**T0**	Required

Remarks:

1. Time lag sets must be selected in the Case Control packet wiht the command:

   RANDOM = SID

2. At least one **RANDPS** entry must be present with the same set identification number.
3. N+1 equally spaced time lag points are defined by this entry.

7 BULK

Bulk Data Entry RBAR

Defines a rigid bar with six degrees of freedom at each end.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RBAR	EID	GID1	GID2	IDOF1	IDOF2	DDOF1	DDOF2		
RBAR	54	401	483	125	234				

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GID1, GID2	GRID point identification number of connection points.	Integer>0	Required
IDOF1, IDOF2	Independent degrees of freedom in the global coordinate system for the element at GRID points **GID1** and **GID2**, respectively.	DOF Code	[2,3]
DDOF1, DDOF2	Dependent degrees of freedom in the global coordinate system assigned by the element at GRID points **GID1** and **GID2**, respectively.	DOF Code	[3,4]

Remarks:

1. Element identification numbers must be unique.
2. The total number of components in **IDOF1** and **IDOF2** must equal six; for example, **IDOF1**=**1236**, then **IDOF2**=**34**. Furthermore, they must collectively be capable of representing any general rigid body motion of the element.
3. If both **DDOF1** and **DDOF2** are zero or blank, all of the degrees of freedom not in **IDOF1** and **IDOF2** will be placed in the dependent set, m-set.
4. The component **DDOFi** specified are placed in the m-set. Therefore, they must not appear as dependent components in other rigid element definitions, or on Bulk Data entries such as **ASET**, **OMIT**, **SPC** or **SUPORT**, that place them in mutually exclusive sets.
5. Rigid elements are ignored in heat transfer analysis.

7 BULK

Bulk Data Entry RBE1

Defines a rigid body connected to an arbitrary number of GRID points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RBE1	EID	IGID1	IDOF1	IGID2	IDOF2	IGID3	IDOF3		-cont-
-cont-		IGID4	IDOF4	IGID5	IDOF5	IGID6	IDOF6		-cont-
-cont-	"UM"	DGID1	DDOF1	DGID2	DDOF2	DGID3	DDOF3		-cont-
-cont-		DGID4	DDOF4	*CONTINUES IN GROUPS OF 2*					-cont-

RBE1	11	121	2	266	134	231	6		+A
+A		287	2						+B
+B	UM	632	13	682	4	783	5		+C
+A		84	1						

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
IGIDi	GRID point identification numbers at which independent degrees of freedom for the element are assigned.	Integer>0	Required
IDOFi	List of independent degrees of freedom in the global coordinate system for the rigid element at GRID point **IGIDi**. [2]	DOF Code	Required
"UM"	Indicates the start of the data defining the dependent degrees of freedom.	Character	Required
DGIDi	GRID point identification numbers at which dependent degrees of freedom for the element are assigned. [3]	Integer>0	Required
DDOFi	List of dependent degrees of freedom in the global coordinate system for the Rigid element at GRID point **DGIDi**. [3]	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. The total number of components in **IDOF1** to **IDOF6** must equal six and, furthermore, they must collectively be capable of representing any general rigid-body motion of the element.
3. The component **DDOFi** specified are placed in the m-set. Therefore, they must not appear as dependent components in other rigid element definitions, or on Bulk Data entries such as **ASET**, **OMIT**, **SPC** or **SUPORT**, that place them in mutually exclusive sets.
4. Rigid elements are ignored in heat transfer analysis.

7 BULK

Bulk Data Entry RBE2

Defines a rigid body whose independent degrees of freedom are specified at a single GRID point and whose dependent degrees of freedom are specified at an arbitrary number of GRID points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RBE2	EID	IGID	DDOF	*DEPENDENT GRID ID LIST*					-cont-

RBE2	429	101	12	101	114	167	231	256	+A
+A	1006	2198							

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
IGID	The GRID point to which all six independent degrees of freedom for the element are assigned.	Integer>0	Required
DDOF	Dependent degrees of freedom in the global coordinate system at GRID points **DGIDi**. [2,3]	DOF Code	Required
DEPENDENT GRID ID LIST	List of one or more GRID points at which dependent degrees of freedom are assigned. [2,3]	Integer>0	Required

Remarks:

1. Element identification numbers must be unique.
2. The component degrees of freedom indicated by **DDOF** are made dependent at all GRID points, **DGIDi**.
3. The component **DDOFi** specified are placed in the m-set. Therefore, they must not appear as dependent components in other rigid element definitions, or on Bulk Data entries such as **ASET**, **OMIT**, **SPC** or **SUPORT**, that place them in mutually exclusive sets.
4. Rigid elements are ignored in heat transfer analysis.

7 BULK

Bulk Data Entry RBE3

Defines the motion at a *reference* GRID point as the weighted average of the motions at a set of other GRID points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RBE3	EID	OPTION	RGID	RDOF	WT1	IDOF1	IGID11	IGID12	-cont-
-cont-	*LIST CONTINUES*		WT2	IDOF2	IGID21	IGID22	*LIST CONTINUES*		-cont-
-cont-	WT3	IDOF3	IGID31	IGID32	IGID33	*GROUPS OF WTi CONTINUE*			-cont-
-cont-	"UM"	DGID1	DDOF1	DGID2	DDOF2	DGID3	DDOF3		-cont-
-cont-		DGID4	DDOF4	*CONTINUES IN GROUPS OF 2*					-cont-

RBE3	215		401	1234	1.0	15	101	102	+A
+A	103	2.5	3	211	212				+B
+B	UM	178	45	235	23				

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
OPTION	Equilibrium option.	Integer { **0** **1** }	**0**
RGID	Reference GRID point identification number.	Integer>0	Required
RDOF	List of global degrees of freedom whose values will be computed at the reference GRID point.	DOF Code	Required
WTi	Weighting factor for degrees of freedom defined by following GRID points **IGIDij**.	Real	Required
IDOFi	List of global degrees of freedom which have weighting factor **WTi** at GRID points **IGIDij**.	DOF Code	Required
IGIDij	Identification numbers of GRID points whose components **IDOFi** have weighting factors **WTi** in the averaging equations.	Integer>0	[2]
"UM"	Indicates the start of the data defining the dependent degrees of freedom.	Character	[3,4]
DGIDi	Identification numbers of GRID points with dependent degrees of freedom.	Integer>0	[3,4,5]
DDOFi	Single dependent degree of freedom at **DGIDi**.	DOF Code	[3,4,5]

Remarks:

1. Element identification numbers must be unique.
2. Blank spaces may be left at the end of an **IGIDij** sequence.
3. If **"UM"** is not present, then all of the components in **RDOF** at the reference GRID point, and no others, will be made dependent.

7 BULK

4. The default for **UM** data should be used except in cases where the user wishes to include some or all **RDOF** components in displacement sets exclusive from the m-set. If the default is not used for **UM** data then:
   - The total number of components in the m-set (i.e., the total number of dependent degrees of freedom defined by the element) must be equal to the number of components in **RDOF** (four components in the example).
   - The dependent degrees of freedom, **DDOFi**, must be a subset of the components specified by **RDOF** and the groups of **IDOFi** and **IGIDij**.
   - The coefficient matrix R_m must be non-singular in the constraint equation:

     $$R_m\, u_m + R_n\, u_n = 0$$

5. The component **DDOFi** specified are placed in the m-set. Therefore, they must not appear as dependent components in other rigid element definitions, or on Bulk Data entries such as **ASET**, **OMIT**, **SPC** or **SUPORT**, that place them in mutually exclusive sets.
6. Rigid elements are ignored in heat transfer analysis.

Bulk Data Entry RELES

Defines sets of component degrees of freedom at substructure GRID points which are not to be connected during a substructure COMBINE operation.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RELES	SID	SNAME	GID1	DOF1	GID2	DOF2	GID3	DOF3	-cont-
-cont-	GID4	DOF4	*CONTINUES IN GROUPS OF 2*						-cont-

RELES	61	WINGR	110	45	119	124	137	456	+A
+A	189	123							

Field	*Contents*	*Data Range*	*Default*
SID	Connection set identification number. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GIDi	GRID or SCALAR point identification numbers.	Integer>0	Required
DOFi	List of degrees of freedom to be released.	DOF Code	Required

Remarks:

1. The release connectivity set must be selected in the Substructure Control packet with the command:

   CONNECT = SID

   This is a subcommand of the substructure command COMBINE.

2. The **RELES** data will override connections automatically generated or connections defined with **CONCT** Bulk Data.

3. The **RELES** data will *not* override connections defined with **CONCT1** Bulk Data.

4. Connectivities defined during previously executed COMBINE operations will be retained and may be referenced by the GRID point identifier and component of any one of the Basic Substructures associated with that connectivity.

7 BULK

Bulk Data Entry RELES1

Defines sets of component degrees of freedom at substructure GRID points which are not to be connected during a substructure COMBINE operation.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RELES1	SID	SNAME	DOF	*GRID ID LIST*					-cont-
RELES1	888	WING	1	1101	1102	1105	THRU	1110	+A
+A	1121	1130	THRU	1140					

Field	*Contents*	*Data Range*	*Default*
SID	Connection set identification number. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
DOFi	List of degrees of freedom to be released.	DOF Code	Required
GRID ID LIST	GRID or SCALAR point identification numbers.	Integer>0	Required

Remarks:

1. The release connectivity set must be selected in the Substructure Control packet with the command:

   CONNECT = SID

   This is a subcommand of the substructure command COMBINE.

2. The **RELES** data will override connections automatically generated or connections defined with **CONCT** Bulk Data.
3. The **RELES** data will *not* override connections defined with **CONCT1** Bulk Data.
4. Connectivities defined during previously executed COMBINE operations will be retained and may be referenced by the GRID point identifier and component of any one of the Basic Substructures associated with that connectivity.

7 BULK

Bulk Data Entry RFORCE

Defines a static loading condition due to centrifugal and centripetal force fields.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RFORCE	LID	GID	CID	A	V1	V2	V3	METHOD	-cont-
-cont-	RACC								

RFORCE	1	100		1.5	1.0	0.0	0.0		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
GID	GRID point identification number. [2]	Integer>0	**0**
CID	Coordinate system in which **V** is defined.	Integer≥0	**0**
A	Scale factor for rotational velocity in revolutions per unit time.	Real	Required
Vi	Components of the vector defining the direction of rotation in coordinate system **CID**.	Real	[3]
METHOD	Method used to compute the centrifugal force vector. [4]	Integer $\begin{Bmatrix} \mathbf{1} \\ \mathbf{2} \end{Bmatrix}$	**1**
RACC	Scale factor for rotational acceleration in units of revolutions per unit time squared. [5]	Real	**0.0**

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. If **GID** is not given, the origin of the Basic Coordinate System is used.

3. The right hand rule is used to define the direction of rotation and the resulting vector **V**. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.

4. Use **METHOD=1** if the model does not contain coupled mass terms, such as those introduced with the Bulk Data entry **PARAM,COUPMASS,1**. Use **METHOD=2** for models with coupled mass terms.

5. The continuation entry is optional and is used only to define a non-zero value for rotational acceleration.

7 BULK

Rev: V20.1

Bulk Data Entry RFORCE1

Defines a static loading condition due to centrifugal and centripetal force fields.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RFORCE1	LID	GIDA	CIDA	A	V1	V2	V3	METHOD	-cont-
-cont-		GIDB	CIDB	B	W1	W2	W3		

RFORCE1	1								
		200	3	24.	0.	0.	1.		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
GIDA	GRID point identification number. [2]	Integer>0	**0**
CIDA	Coordinate system in which **V** is defined.	Integer≥0	**0**
A	Scale factor for rotational velocity in revolutions per unit time.	Real	Required
Vi	Components of the vector defining the direction of rotation in coordinate system **CIDA**. [3]	Real	**0.0**
METHOD	Method used to compute the centrifugal force vector. [4]	Integer $\begin{Bmatrix} \mathbf{1} \\ \mathbf{2} \end{Bmatrix}$	**1**
GIDB	GRID point identification number. [5]	Integer>0	**0**
CIDB	Coordinate system in which **W** is defined.	Integer≥0	**0**
B	Scale factor for rotational acceleration in revolutions per unit time squared. [6]	Real	**0.0**
Wi	Components of the vector defining the direction of rotational acceleration in coordinate system **CIDB**. [3]	Real	**0.0**

7 BULK

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

2. If **GIDA** is not given, the origin of the Basic Coordinate System is used. **GIDA** is the GRID point through which the rotational velocity vector acts.

Rev: V20.1

3. The right hand rule is used to define the direction of rotation and the resulting vector **V**. Although the default value for each **Vi** is **0.0**, at least one of the components must be non-zero. Note that **V** is not normalized, and may thus contribute to the magnitude of the load as well as its direction.
4. Use **METHOD=1** if the model does not contain coupled mass terms, such as those introduced with the Bulk Data entry **PARAM,COUPMASS,1**. Use **METHOD=2** for models with coupled mass terms.
5. If **GIDB** is not given, the origin of the Basic Coordinate System is used. **GIDB** is the GRID point through which the rotational acceleration vector acts.
6. The continuation entry is optional and is used only to define a non-zero value for rotational acceleration. If the rotational velocity is zero, fields 3 through 9 of the parent entry may be left blank.

Bulk Data Entry RINGAX

Defines a ring for an axisymmetric harmonic model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RINGAX	RID		R	Z			PSPC		
RINGAX	104		4.5	13.2			162		

Field	*Contents*	*Data Range*	*Default*
RID	Ring identification number. [1]	Integer>0	Required
R	Ring radius. [2]	Real>0.0	Required
Z	Ring axial location. [2]	Real	Required
PSPC	Permanent single-point constraints.	DOF Code	**0**

Remarks:

1. RINGAX identification numbers must be unique with respect to all other point identification numbers.
2. Axisymmetric rings are always located in the implicit cylindrical coordinate system introduced for axisymmetric harmonic analysis.
3. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.
4. A ring defines six degrees of freedom for each harmonic requested by the **AXIC** entry.

7 BULK

Bulk Data Entry RINGFL

Defines a fluid point in an axisymmetric fluid model.

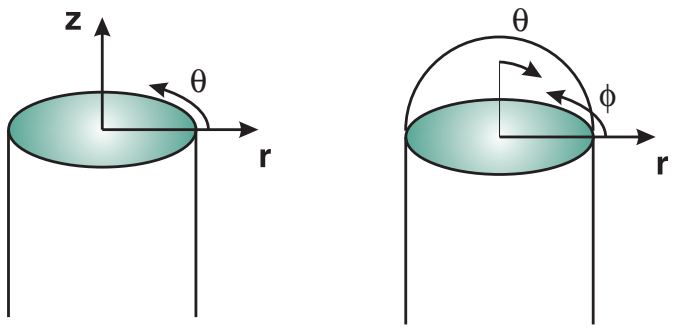
Format and Example:

1	2	3	4	5	6	7	8	9	10
RINGFL	FID	X1	X2	X3					
RINGFL	104	1.2		30.0					

Field	*Contents*	*Data Range*	*Default*
FID	Fluid point identification number. [1]	0<Integer<10^5	Required
Xi	Coordinates of point in fluid coordinate system defined by the **AXIF** Bulk Data entry. [2,3]	Real **X1**>0.0	Required

Remarks:

1. All fluid point identification numbers must be unique with respect to other SCALAR, GRID and fluid points.
2. When the fluid coordinate system is cylindrical, **X1** and **X3** represent (r,z), and **X2** (θ) must be left blank. Similarly, when the fluid coordinate system is spherical, **X1** and **X2** represent (r,θ), and **X3** (ϕ) must be left blank. θ is measured in degrees.



CYLINDRICAL SYSTEM SPHERICAL SYSTEM

3. This entry is allowed only if an **AXIF** entry is also present in the Bulk Data packet.

7 BULK

Bulk Data Entry RLOAD1

Defines a frequency-dependent dynamic load for use in frequency response analyses, of the form:

$$P(f) = A\,[C(f) + iD(f)]\,e^{i(\theta - 2\pi f\tau)}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
RLOAD1	LID	ADEF	τDEF	θDEF	CTAB	DTAB	DYNEX		
RLOAD1	106	104	107	110	102	103			

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
ADEF	Identification number of loads data which defines *A*. [3]	Integer>0	Required
tDEF	Identification number of a **DELAY** or **DELAYS** Bulk Data entry which defines τ.	Integer>0	[4]
qDEF	Identification number of a **DPHASE** or **DPHASES** Bulk Data entry which defines θ.	Integer>0	[4]
CTAB	Identification number of a **TABLEDi** Bulk Data entry which gives *C(f)*.	Integer>0	[4,5]
DTAB	Identification number of a **TABLEDi** Bulk Data entry which gives *D(f)*.	Integer>0	[4,5]
DYNEX	Type of dynamic excitation. [6]	0ôInteger≥7	**0**

Remarks:

1. Dynamic loads sets must be selected in the Case Control packet with the command:

   DLOAD = LID

2. **LID** must be unique for all **DLOAD**, **RLOAD1** and **RLOAD2** data. **RLOAD1** loads may be combined with **RLOAD2** loads only with **DLOAD** data. That is, the **LID** on an **RLOAD1** entry may not be the same as that on an **RLOAD2** entry.
3. **ADEF** may reference any combination of **DAREA**, **DAREAS**, **FORCEi**, **MOMENTi**, **PLOADi**, **LOADC**, **GRAV** and **RFORCE** Bulk Data entries. When **ADEF** references **LOADC** data, **DAREAS** entries with the same set identification and non-zero loads must also exist.
4. If any of τ**DEF**, θ**DEF**, **CTAB**, or **DTAB** are blank or zero, the corresponding τ, θ, *C(f)*, or *D(f)* will be zero.
5. Either **CTAB** or **DTAB** must be present.

7 BULK

6. The table below defines the **DYNEX** dynamic excitation options which are available:

DYNEX	EXCITATION FUNCTION	DYNEX	EXCITATION FUNCTION
0	**FORCEi, MOMENTi,** or **PLOADi**	**4**	Fluid Pressure
1	Enforced Displacement	**5**	Fluid Volumetric Flux
2	Enforced Velocity	**6**	Fluid Volumetric Velocity
3	Enforced Acceleration	**7**	Fluid Volumetric Acceleration

When any enforced motion option is selected (**DYNEX** of **1**, **2**, or **3**), you must apply the dynamic excitation to *s-set* degrees of freedom.

7. When performing substructuring analyses, **DAREAS** data may only reference degrees of freedom in the boundary set of the solution structure.

Bulk Data Entry RLOAD2

Defines a frequency-dependent dynamic load for use in frequency response analyses, of the form:

$$P(f) = AB(f)\, e^{i(C(f) + \theta - 2\pi f\tau)}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
RLOAD2	LID	ADEF	τDEF	θDEF	BTAB	CTAB	DYNEX		
RLOAD2	100	1001	1002	1003	101	201			

Field	*Contents*	*Data Range*	*Default*
LID	Set identification number. [1,2]	Integer>0	Required
ADEF	Identification number of loads data which defines *A*. [3]	Integer>0	Required
tDEF	Identification number of **a DELAY** or **DELAYS** Bulk Data entry which defines τ.	Integer>0	[4]
qDEF	Identification number of a **DPHASE** or **DPHASES** Bulk Data entry which defines θ.	Integer>0	[4]
BTAB	Identification number of a **TABLEDi** Bulk Data entry which defines *B(f)*.	Integer>0	Required
CTAB	Identification number of a **TABLEDi** Bulk Data entry which defines *C(f)*.	Integer>0	[4]
DYNEX	Type of dynamic excitation. [5]	0ôInteger≥7	**0**

Remarks:

1. Dynamic loads sets must be selected in the Case Control packet with the command:

   DLOAD = LID

2. **LID** must be unique for all **DLOAD**, **RLOAD1** and **RLOAD2** data. **RLOAD1** loads may be combined with **RLOAD2** loads only with **DLOAD** data. That is, the **LID** on an **RLOAD1** entry may not be the same as that on an **RLOAD2** entry.

3. **ADEF** may reference any combination of **DAREA**, **DAREAS**, **FORCEi**, **MOMENTi**, **PLOADi**, **LOADC**, **GRAV** and **RFORCE** Bulk Data entries. When **ADEF** references **LOADC** data, **DAREAS** entries with the same set identification and non-zero loads must also exist.

4. If any of τ**DEF**, θ**DEF**, or **CTAB** are blank or zero, the corresponding τ, θ, or *C(f)* will be zero.

7 BULK

5. The table below defines the **DYNEX** dynamic excitation options which are available:

DYNEX	EXCITATION FUNCTION	DYNEX	EXCITATION FUNCTION
0	**FORCEi, MOMENTi,** or **PLOADi**	**4**	Fluid Pressure
1	Enforced Displacement	**5**	Fluid Volumetric Flux
2	Enforced Velocity	**6**	Fluid Volumetric Velocity
3	Enforced Acceleration	**7**	Fluid Volumetric Acceleration

When any enforced motion option is selected (**DYNEX** of **1**, **2**, or **3**), you must apply the dynamic excitation to *s-set* degrees of freedom.

6. With Substructure Analysis, **DAREAS** data may only reference degrees of freedom in the boundary set of the solution substructure.

Bulk Data Entry RROD

Defines a pin-ended rod that is rigid in extension.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RROD	EID	GID1	GID2	DDOF1	DDOF2				
RROD	145	101	102	2					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GID1 **GID2**	GRID point identification numbers of connection points.	Integer>0	Required
DDOF1 **DDOF2**	Single dependent **translational** degree of freedom in the global coordinate system for **GIDi**. [2,3]	DOF Code	[4]

Remarks:

1. Element identification numbers must be unique.
2. The degree of freedom selected to be dependent must have a nonzero component along the axis of the rod.
3. The component **DDOFi** specified are placed in the m-set. Therefore, they must not appear as dependent components in other rigid element definitions, or on Bulk Data entries such as **ASET**, **OMIT**, **SPC** or **SUPORT**, that place them in mutually exclusive sets.
4. Only one of the fields **DDOF1** or **DDOF2** may contain a value. The other must be blank.
5. Rigid elements are ignored in heat transfer analysis.

Bulk Data Entry RSPLINE

Defines multipoint constraints for the interpolation of displacements at GRID points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RSPLINE	EID	D/L	GIDF	GID1	DOF1	GID2	DOF2	GID3	-cont-
-cont-	DOF3	*CONTINUES IN GROUPS OF 2*				GIDL			

RSPLINE	152		1001	1002	123456	1003		1004	+A
+A	16	2	23	2001					

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
D/L	Ratio of the diameter of an elastic tube, which the SPLINE represents, to its length.	Real>0.0	**0.1**
GIDF, GIDL	GRID point identification numbers of the independent, first and last points defining the element. [2]	Integer>0	Required
GIDi	GRID point identification numbers of internal, dependent connection points. [3]	Integer>0	Required
DOFi	Degrees of freedom to be constrained at GRID point **GIDi**. [3]	DOF Code	Required

Remarks:

1. Element identification numbers must be unique.
2. Displacements are interpolated from the equations of an elastic tube passing through the GRID points.
3. A blank entry in **DOFi** indicates that all six degrees of freedom at **GIDi** are independent. A non-blank entry selects the dependent components at **GIDi**. By definition all six components of GRID points **GIDF** and **GIDL** are independent, therefore no field is provided to select their components. For the example shown, GRID points **1001**, **1003** and **2001** are independent; **1002** has six dependent degrees of freedom; and **1004** and **2** each have two.
4. The component **DDOFi** specified are placed in the m-set. Therefore, they must not appear as dependent components in other rigid element definitions, or on Bulk Data entries such as **ASET**, **OMIT**, **SPC** or **SUPORT**, that place them in mutually exclusive sets.
5. Rigid elements are ignored in heat transfer analysis.

7 BULK

Bulk Data Entry RTRPLT

Defines a rigid triangular plate.

Format and Example:

1	2	3	4	5	6	7	8	9	10
RTRPLT	EID	GID1	GID2	GID3	IDOF1	IDOF2	IDOF3		-cont-
-cont-	DDOF1	DDOF2	DDOF3						

RTRPLT	107	151	152	153	1236	3	3		+A
+A	6								

Field	*Contents*	*Data Range*	*Default*
EID	Element identification number. [1]	Integer>0	Required
GIDi	GRID point identification numbers of connection points.	Integer>0	Required
IDOFi	Independent degrees of freedom in the global coordinate system for the element at GRID point **GIDi**.	DOF Code	[2,3]
DDOFi	Dependent degrees of freedom in the global coordinate system.	DOF Code	[3,4]

Remarks:

1. Element identification numbers must be unique.
2. The total number of components in **IDOFi** must equal six and they must collectively be capable of representing any general rigid body motion of the element.
3. By default, all of the degrees of freedom not in **IDOFi** will be made dependent.
4. The component **DDOFi** specified are placed in the m-set. Therefore, they must not appear as dependent components in other rigid element definitions, or on Bulk Data entries such as **ASET**, **OMIT**, **SPC** or **SUPORT**, that place them in mutually exclusive sets.
5. Rigid elements are ignored in heat transfer analysis.

7 BULK

Bulk Data Entry SECTAX

Defines a sector of a model containing CONEAX, TRAPAX or TRIAAX elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SECTAX	ID	RID	R	$\phi 1$	$\phi 2$				

SECTAX	1	2	3.0	30.0	40.0				

Field	*Contents*	*Data Range*	*Default*
ID	Sector identification number. [1]	Integer>0	Required
RID	Ring identification number defined by a **RINGAX** Bulk Data entry.	Integer>0	Required
R	Effective radius.	Real>0.0	Required
fi	Azimuthal limits of the sector in degrees.	Real	Required

Remarks:

1. **SECTAX** identification numbers must be unique with respect to all other point identification numbers.
2. This entry is allowed only if an **AXIC** Bulk Data entry is also present.

7 BULK

Bulk Data Entry SEQEP

Allows resequencing of the EXTRA points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SEQEP	EPID1	SEQ1	EPID2	SEQ2	EPID3	SEQ3	EPID4	SEQ4	
SEQEP	2378	14.6	23	1.9.6	37	4			

Field	*Contents*	*Data Range*	*Default*
EPIDi	EXTRA point identification numbers.	Integer	Required
SEQi	Sequenced identification number. [1,2]	[1]	Required

Remarks:

1. The **SEQi** are of the form:

   ***a*[.*b*[.*c*[.*d*]]]**

   where ***a*** is an integer value of up to 4 digits, and ***b***, ***c***, and ***d*** are single digits. The single digits are used only when the new sequence number is inserted between two existing *sequential* identification numbers.

2. The **SEQi** numbers must be unique and may not be the same as identification numbers of EXTRA points whose sequence is not being changed. EXTRA point identification numbers may not be referenced more than once.

3. This entry may be used to explicitly define the formation sequence of the EXTRA points of the structural model, in such a way as to optimize bandwidth. This is essential for efficient solutions by the displacement method.

Bulk Data Entry SEQGP

Allows resequencing of the GRID and SCALAR points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SEQGP	GID1	SEQ1	GID2	SEQ2	GID3	SEQ3	GID4	SEQ4	
SEQGP	5	27.8	101	3.6.2	102	1.9	66	127	

Field	*Contents*	*Data Range*	*Default*
GIDi	GRID or SCALAR point identification numbers.	Integer>0	Required
SEQi	Sequence identification number. [1,2]	[1]	Required

Remarks:

1. The **SEQi** are of the form:

   ***a*[.*b*[.*c*[.*d*]]]**

   where ***a*** is an integer value of up to 4 digits, and ***b***, ***c***, and ***d*** are single digits. The single digits are used only when the new sequence number is inserted between two existing *sequential* identification numbers.

2. The **SEQi** numbers must be unique and may not be the same as identification numbers of GRID or SCALAR points whose sequence is not being changed. GRID or SCALAR point identification numbers may not be referenced more than once.

3. This entry may be used to explicitly define the formation sequence of the GRID or SCALAR points of the structural model, in such a way as to optimize bandwidth. This is essential for efficient solutions by the displacement method.

4. UAI/*NASTRAN* automatically creates **SEQGP** data internally for structural and heat transfer models to mimimize the solution time. The use of **SEQGP** Bulk Data entries may cause a less efficient solution to occur. Refer to the Executive Control command **SEQUENCE** for more information.

7 BULK

Bulk Data Entry SET1

Defines a set of structural GRID points.

Format and Example:

1	2	3	4	**5**	6	7	8	9	10
SET1	LABEL	G1	G2	G3	G4	G5	G6	G7	-cont-
-cont-	G8	*CONTINUES WITH LIST OF GRID ID'S*							-cont-

SET1	ELEV1	101	105	108	110	111	120	131	

Field	*Comments*	*Data Range*	*Default*
LABEL	Identification label of GRID point list. [1]	Character	Required
Gi	GRID point identification numbers. [2]	Integer>0	Required

Remarks:

1. The **SET1** GRID point list is referenced by **SPLINEi** Bulk Data entries.
2. A minimum of two GRID points are required for linear spline, and three GRID points for surface splines.

7 BULK

Bulk Data Entry SETI

Defines an integer set to be used for defining, for example, groups of GRID points or elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SETI	SID	IVAL1	IVAL2	IVAL3	*CONTINUES WITH LIST OF VALUES*				-CONT-
SET1	101	1	THRU	100	BY	2	201	203	-CONT-
-CONT-	219	301	THRU	399					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1,2]	Integer>0	Required
IVALi	Integer value to be included in set **SID**. [3]	Integer>0	Required

Remarks:

1. The set identification number, **SID**, must be unique among all **SETI** (including its synonyms **GPSET**, **GLIST**, **ELSET** and **ELIST**), **SETR** and **SETOP** Bulk Data entries.
2. The set identification number, **SID**, is referenced by other Bulk Data entries.
3. The list of values may include any number of **THRU** and **THRU/BY** clauses as described in the introduction to this Chapter.
4. For user convenience, this Bulk data entry has several synonyms to make the overall Bulk Data packet easier to read. **GPSET**, **GLIST**, **ELSET** and, **ELIST** may be used instead of **SETI**.

7 BULK

New: V20.1

Bulk Data Entry SETOP

Defines a new integer set by performing operations on two or more other integer **SET**s.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SETOP	SID	OPER	SID1	SID2	*CONTINUES WITH LIST OF VALUES*				-CONT-
SETOP	101	UNION	101	201	301				

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1,2]	Integer>0	Required
OPER	Specifies the operation to be performed on the selected sets. You may choose from **UNION** or **INTER-**section. [2]	Character	Required
SIDi	Set identification numbers on which **OPER** will be performed to create **SID**. [3]	Integer>0	Required

Remarks:

1. The set identification number, **SID**, must be unique among all **SETOP**, **SETI** (including its synonyms **GPSET**, **GLIST**, **ELSET** and **ELIST**) and **SETR** Bulk Data entries.
2. The set identification number, **SID**, is referenced by other Bulk Data entries
3. The **UNION** operation results in the set X where:

   $X = \{ x \mid x \in \text{of } \textbf{SOME SIDi} \}$

   Similarly, the **INTER**section results in:

   $X = \{ x \mid x \in \text{of } \textbf{ALL SIDi} \}$

7 BULK

New: V20.1

Bulk Data Entry SETR

Defines a set of real numbers. This set may be used for defining, for example, axial stations along a **BMFORCE** element.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SETR	SID	RVAL1	RVAL2	RVAL3	*CONTINUES WITH LIST OF VALUES*				CONT
SETR	101	1.0	THRU	5.0	BY	0.2	17.5	20.3	-CONT-
-CONT-	24.3	30.0	THRU	32.0	BY	0.1			

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1,2]	Integer>0	Required
RVALi	Real value to be included in set **SID**. [3]	Real	Required

Remarks:

1. The set identification number, **SID**, must be unique among all **SETR**, **SETI** (including its synonyms **GPSET**, **GLIST**, **ELSET** and **ELIST**) and **SETOP** Bulk Data entries.
2. The set identification number, **SID**, is referenced by other Bulk Data entries
3. The list of values may include any number of **THRU** and **THRU/BY** clauses as described in the introduction to this Chapter. If there is no **BY** term, then the increment is assumed to be **1.0**.

7 BULK

New: V20.1

Bulk Data Entry SHOCK

Defines a Response Spectra Shock Response load.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SHOCK	LID	GID	COMP	DYNEX	DMPTYP				-cont-
-cont-	DVAL1	TID1	DVAL2	TID2	*CONTINUES IN GROUPS OF 2*				-cont-

SHOCK	101	999	2						+A
+A	0.01	101	0.02	201					

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
GID	GRID point identification number.	Integer>0	Required
COMP	Component code.	DOF Code	Required
DYNEX	Type of dynamic excitation. [2]	Integer { **1** **2** **3** }	Required
DMPTYP	Type of damping. [3]	Character { **G** **FRACTION** **PERCENT** **Q** }	**G**
DVALi	Modal damping value. [3]	Real≠0.0	Required
TIDi	Identification number of a **TABLED1** Bulk Data entry which defines the peak excitation as a function of frequency for the specified modal damping value **DVALi**.	Integer>0	Required

Remarks:

1. Shock spectra loads must be selected in the Case Control packet with the command:

   SHOCK = LID

2. The table defines the **DYNEX** dynamic excitation options which are available:

DYNEX Value	EXCITATION FUNCTION	**DYNEX** Value	EXCITATION FUNCTION	DYNEX Value	EXCITATION FUNCTION
1	Enforced Displacement	**2**	Enforced Velocity	**3**	Enforced Acceleration

3. The modal damping values must be defined in either of the four available types: **PERCENT**age of critical damping; **FRACTION** of critical damping; twice the fraction of critical damping, **G**; or the reciprocal of **G**, called damping amplification factor, **Q**.

Mod: V20.1

Bulk Data Entries SKIPON, SKIPOFF

Disables and enables the processing of Bulk Data entries.

Formats:

1	2	3	4	5	6	7	8	9	10
SKIPON									

SKIPOFF									

Field	*Contents*	*Data Range*	*Default*

There is no additional input for these entries.

Remarks:

1. The processing of Bulk Data entries is suspended when a **SKIPON** entry is encountered in the Bulk Data packet. All following entries are echoed but not processed until a **SKIPOFF** entry is encountered. Echoing of skipped entries may be controlled by the **ECHOOFF** and **ECHOON** entries.
2. As many pairs of **SKIPON**/**SKIPOFF** entries may be used as desired.

Bulk Data Entry SLOAD

Defines static loads applied to SCALAR points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SLOAD	LID	SPID1	F1	SPID2	F2	SPID3	F3		
SLOAD	16	2	5.9	17	-6.2	11	-7.83		

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1]	Integer>0	Required
SPIDi	SCALAR point identification numbers.	Integer>0	Required
Fi	Load values.	Real	Required

Remarks:

1. Load sets must be selected in the Case Control packet with the command:

   LOAD = LID

7 BULK

Bulk Data Entry SPC

Defines sets of single-point constraints and static enforced displacements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SPC	SID	GID1	DOF1	$\delta 1$	GID2	DOF2	$\delta 2$		
SPC	103	133	1	1.25	106	1	-2.4		

Field	*Contents*	*Data Range*	*Default*
SID	Single-point constraint set identification number. [1]	Integer>0	Required
GIDi	GRID or SCALAR point identification numbers.	Integer>0	Required
DOFi	List of degrees of freedom.	DOF Code	Required
di	Value of enforced displacement for all components designated by **GIDi** and **DOFi**. [2]	Real	**0.0**

Remarks:

1. Single-point constraint sets must be selected in the Case Control packet with the command:

   SPC = SID

2. Enforced displacements in dynamics are defined using **TLOADi** and **RLOADi** Bulk Data entries. They may not be defined using the **SPC** Bulk Data entry.

3. The component **DOFi** specified are placed in the s-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SUPORT** or **OMIT**, that place them in mutually exclusive sets.

4. **SPC** degrees of freedom may be redundantly specified as permanent constraints on **GRID** Bulk Data entries.

5. In Nonlinear Steady State Heat Transfer analysis, all boundary condition degrees of freedom which have prescribed temperature values must be specified using both **SPC** and **TEMP** or **TEMPD** Bulk Data entries. The boundary condition values which are used by **UAI/*NASTRAN*** are those specified on the **TEMP** or **TEMPD** entries. These Bulk Data entries must also be selected with the **TEMP(ESTIMATED), TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control command.

Bulk Data Entry SPC1

Defines sets of single-point constraints.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
SPC1	SID	DOF	*GRID ID LIST*						-cont-
SPC1	999	123456	1	10	20	THRU	30	35	

Field	*Contents*	*Data Range*	*Default*
SID	Single-point constraint set identification number. [1]	Integer>0	Required
DOF	List of degrees of freedom.	DOF Code	Required
GRID ID LIST	List of one or more GRID or SCALAR point identification numbers.	Integer>0	Required

Remarks:

1. Single-point constraint sets must be selected in the Case Control packet with the command:

   SPC = SID

2. The component **DOF** specified are placed in the s-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SUPORT** or **OMIT**, that place them in mutually exclusive sets.

3. **SPC** degrees of freedom may be redundantly specified as permanent constraints on **GRID** Bulk Data entries.

4. Enforced displacements are not available with this entry.

5. In Nonlinear Steady State Heat Transfer analysis, all boundary condition degrees of freedom which have prescribed temperature values must be specified using both **SPC** and **TEMP** or **TEMPD** Bulk Data entries. The boundary condition values which are used by **UAI/*NASTRAN*** are those specified on the **TEMP** or **TEMPD** entries. These Bulk Data entries must also be selected with the **TEMP(ESTIMATED), TEMP(MATERIAL)** or **TEMP(BOTH)** Case Control command.

7 BULK

Bulk Data Entry SPCADD

Defines a single-point constraint set as the union of single-point constraint sets defined with **SPC** or **SPC1** Bulk Data entries.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SPCADD	SID	*SET ID LIST*							-cont-

SPCADD	202	191	201	215	257	3	794	801	+A
+A	914								

Field	*Contents*	*Data Range*	*Default*
SID	Single-point constraint set identification number. [1]	Integer>0	Required
SET ID LIST	List of one or more single-point constraint set identification numbers defined with **SPC** or **SPC1** entries. [2,3]	Integer> **SETi≠SID**	Required

Remarks:

1. Single-point constraint sets must be selected in the Case Control packet with the command:

   SPC = SID

2. The ***SET ID LIST*** values must be unique, and they may not be the identification numbers of single-point constraint sets defined by other **SPCADD** entries.

7 BULK

Bulk Data Entry SPCAX

Defines a single-point constraint set for axisymmetric harmonic models.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SPCAX	SID	RID	HID	DOF	δ				
SPCAX	103	104	4	13	4.2				

Field	*Contents*	*Data Range*	*Default*
SID	Single-point constraint set identification number. [1]	Integer>0	Required
RID	Ring identification number.	Integer>0	Required
HID	Harmonic identification number.	Integer>0 or Harmonic Code	Required
DOF	List of degrees of freedom.	DOF Code	Required
d	Enforced displacement value.	Real	**0.0**

Remarks:

1. Single-point constraint sets must be selected in the Case Control packet with the command:

   SPC = SID

2. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

3. The component **DOF** specified are placed in the s-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SUPORT** or **OMIT**, that place them in mutually exclusive sets.

7 BULK

Bulk Data Entry SPCD

Defines enforced displacement values for static analysis.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SPCD	LID	GID1	DOF1	$\delta 1$	GID2	DOF2	$\delta 2$		

SPCD	100	32	436	-2.6	5		2.9		

Field	*Contents*	*Data Range*	*Default*
LID	Static load set identification number. [1]	Integer>0	Required
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom. [2]	DOF Code	Required
di	Value of enforced displacement for all components designated by **GIDi** and **DOFi**. [3]	Real	Required

Remarks:

1. Enforced displacements are selected like loads with the Case Control command:

   LOAD = LID

2. A component referenced on an **SPCD** entry must also be referenced by a selected **SPC** or **SPC1** Bulk Data entry.

3. Values of δi will override the values specified on an **SPC** Bulk Data entry, if the **LOAD** set is requested.

4. **SPCD** data may be combined with other loads using the **LOAD** Bulk Data entry.

7 BULK

Bulk Data Entry SPCS

Defines a single-point constraint set for a specified Basic Substructure.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SPCS	SID	SNAME	GID1	DOF1	GID2	DOF2	GID3	DOF3	-cont-
-cont-	GID4	DOF4	*CONTINUES IN GROUPS OF 2*						-cont-

SPCS	61	WINGR	110	45	119	124	137	456	+A
+A	189	123							

Field	*Contents*	*Data Range*	*Default*
SID	Single-point constraint set identification number. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom.	DOF Code	Required

Remarks:

1. Single-point constraint sets must be selected in the Case Control packet with the command:

   SPC = SID

2. A single **GIDi,DOFi** pair may not specify all component degrees of freedom for a connected GRID point when only some of the degrees of freedom of the GRID point have been connected, or when some have been disconnected with a **RELES** entry. The connected degrees of freedom must be referenced separately from those that were not connected.

3. The component **DOFi** specified are placed in the s-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SUPORT** or **OMIT**, that place them in mutually exclusive sets.

7 BULK

Bulk Data Entry SPCS1

Defines a set of single-point constraints for a specified Basic Substructure.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SPCS1	SID	SNAME	DOF	*GRID ID LIST*					-cont-

SPCS1	116	FUSLG	1236	1101	1102	1105	THRU	1110	+A
+A	1121	1130	THRU	1140					

Field	*Contents*	*Data Range*	*Default*
SID	Single-point constraint set identification number. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
DOF	List of degrees of freedom. [2]	DOF Code	Required
GRID ID LIST	List of one or more GRID or SCALAR point identification numbers. [2]	Integer>0	Required

Remarks:

1. Single-point constraint sets must be selected in the Case Control packet with the command:

   SPC = SID

2. A single **GIDi,DOF** pair may not specify all component degrees of freedom for a connected GRID point when only some of the degrees of freedom of the GRID point have been connected, or when some have been disconnected with a **RELES** entry. The connected degrees of freedom must be referenced separately from those that were not connected.

3. The component **DOF** specified are placed in the s-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SUPORT** or **OMIT**, that place them in mutually exclusive sets.

7 BULK

Bulk Data Entry SPCSD

Defines enforced static displacements for a Basic Substructure.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SPCSD	LID	SNAME	GID1	DOF1	δ1	GID2	DOF2	δ2	-cont-
SPCSD	27	LWING	965	3	3.6				

Field	*Contents*	*Data Range*	*Default*
LID	Static load set identification number. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GIDi	GRID or SCALAR point identification number. [2]	Integer>0	Required
DOFi	List of degrees of freedom. [2,3]	DOF Code	Required
di	Value of enforced displacement for all components designated by **GIDi** and **DOFi**.	Real	**0.0**

Remarks:

1. Enforced displacements are selected like loads with the Case Control command:

   LOAD = LID

2. A single **GIDi,DOFi** pair may not specify all component degrees of freedom for a connected GRID point when only some of the degrees of freedom of the GRID point have been connected, or when some have been disconnected with a **RELES** entry. The connected degrees of freedom must be referenced separately from those that were not connected.

3. A component referenced on an **SPCSD** entry must also be referenced by selected **SPCS** or **SPCS1** data.

7 BULK

Bulk Data Entry SPLINE1

Defines a surface spline for interpolating out-of-plane motion of a structural GRID to an aerodynamic GRID for aeroelastic problems.

Format and Example:

1	2	3	4	**5**	6	7	8	9	10
SPLINE1	EID	CAERO	BOX1	BOX2	SETG	DZ	METHOD	NC	-cont-
-cont-	NS	FC	FS						

SPLINE1	101	101			11	0.0	FINITE		

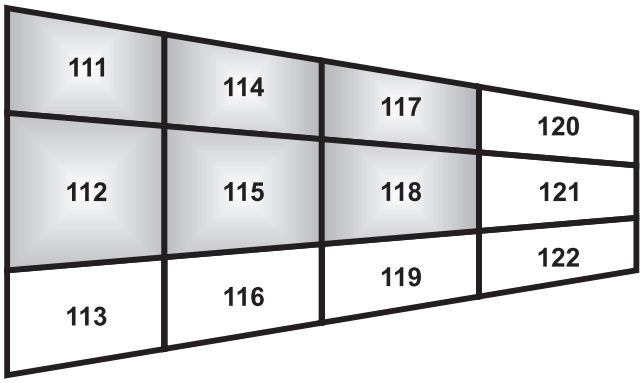
Field	*Comments*	*Data Range*	*Default*
EID	Spline identification number. [1]	Integer>0	Required
CAERO	Identification label of a **CAERO** panel whose aero boxes are interpolated using this spline [2]	Character	Required
BOX1	First box whose motions are interpolated using this spline. [3]	Integer>0 or **FIRST**	**FIRST**
BOX2	Last box whose motions are interpolated using this spline. [3]	Integer>0 or **LAST**	**LAST**
SETG	Identification number of a **SETi** Bulk Data entry that lists the structural GRID points to which the spline is attached.	Integer>0	Required
DZ	Linear attachment flexibility. [4]	Real≥0.0	**0.0**
METHOD	**INFINITE** for the infinite plate spline, and **FINITE** for the finite surface spline.	Keyword { **INFINITE** / **FINITE** }	**INFINITE**
NC	Number of equal chordwise segments in the finite surface finite element mesh.	Integer>0	**10**
NS	Number of equal spanwise segments in the finite surface finite element mesh.	Integer>0	**10**
FC	Margin of spline region as decimal fraction of the chordwise length of the finite element mesh.	Real>0.0	**0.01**
FS	Margin of spline region as decimal fraction of the spanwise length of the finite element mesh.	Real>0.0	**0.01**

Remarks:

1. The spline **EID** is used only for error messages and it is not related to the macroelement identification number.
2. The spline plane is assumed to be the **CAERO** macroelement plane.

7 BULK

3. The interpolated points (k-set) are defined by aero cells. The figure below illustrates the cells for which u_k is interpolated if **BOX1=111** and **BOX2 = 118**.



To include all **CAERO** boxes, set the fields **BOX1** and **BOX2** to **FIRST** and **LAST**, respectively, or leave them blank.

4. The attachment flexibility (units of area) is used for smoothing the interpolation. If **DZ=0.0**, the spline will pass through all deflected GRID points. If **DZ** is much greater than the area of the spline, a least-squares plane will be fit between the points. Intermediate values provide smoothing.

Input Data Entry: SPLINE2

Defines a beam spline for interpolating panels and bodies for steady and unsteady aeroelastic analyses.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
SPLINE2	EID	MACROID	BOX1	BOX2	SETG	DZ	DTOR	CID	-cont-
-cont-	DTHX	DTHY							

SPLINE2	1000	5000	5000	5100	10	0.	1.0	4	+ABC
+BC	-1.								

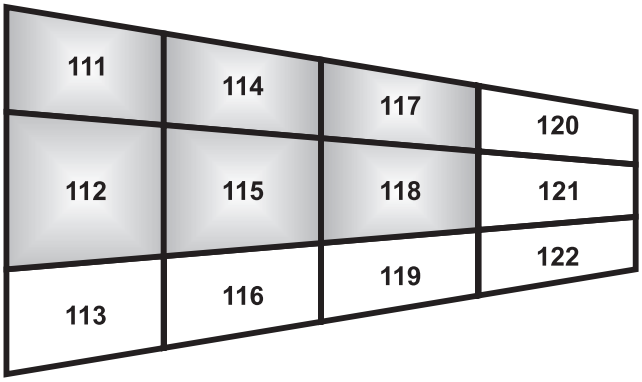
Field	*Contents*	*Data Range*	*Default*
EID	Spline identification number. [1]	Integer > 0	Required
MACROID	The identification number of a **CAERO1** aerodynamic macroelement to be splined. [2,3]	Integer > 0	Required
BOX1	The identification number of the first box on the macroelement to be interpolated using this spline. [4]	Integer > 0 or **FIRST**	**FIRST**
BOX2	The identification number of the last box on the macroelement to be interpolated using this spline. [4]	Integer > 0 or **LAST**	**LAST**
SETG	The identification of a **SETi** entry which lists the structural GRID points to which the spline is attached	Integer > 0	Required
DZ	Linear attachment flexibility.	Real ≥ 0.0	**0.0**
DTOR	Torsional flexibility, $^{EI}/_{GJ}$ (use 1.0 for bodies)	Real ≥ 0.0	**0.0**
CID	Rectangular coordinate system which defines the y-axis of the spline.	Integer > 0	**0**
DTHX,DTHY	Rotational attachment flexibility. **DTHX** is for rotation about the x-axis; not used for bodies. **DTHY** is for rotation about the y-axis; used for slope of bodies. [5]	Real	**0.0**

Remarks:

1. The **SPLINE2 EID** must be unique with respect to all other **SPLINE**i data entries, it is used only for error messages.
2. The interpolation points (k-set) will be defined by aero-cells.
3. For panels, the spline axis is the projection of the y-axis of coordinate system **CID** onto the plane of the panel. For bodies, the spline axis is parallel to the x-axis of the aerodynamic coordinate system.

7 BULK

4. The interpolated points (k-set) are defined by aero cells. The figure below illustrates the cells for which u_k is interpolated if **BOX1=111** and **BOX2 = 118**.



To include all **CAERO** boxes, set the fields **BOX1** and **BOX2** to **FIRST** and **LAST**, respectively, or leave them blank.

5. The flexibilities are used for smoothing. Zero attachment flexibilities will imply rigid attachment, i.e., no smoothing. Negative values of **DTHX** and /or **DTHY** will imply no attachment.

Bulk Data Entry SPOINT

Defines SCALAR points of the structural model.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
SPOINT	*SCALAR POINT ID LIST*								

SPOINT	108	1294	234	235	100				

Field	*Contents*	*Data Range*	*Default*
SCALAR POINT ID LIST	List of one or more SCALAR point identification numbers. [1,2,3]	Integer>0 [2] **SIDF<SIDL**	Required

Remarks:

1. SCALAR point identification numbers must be unique with respect to all other GRID, SCALAR, and fluid points.
2. The maximum value for the SCALAR Point identification number is one tenth of the largest integer that can be represented on the **UAI*/NASTRAN*** host computer.
3. SCALAR points which appear on a spring, damping or mass connection entry need not appear on this entry. This entry is used primarily to define SCALAR points appearing in single or multipoint constraint equations.

7 BULK

Bulk Data Entry SUPAX

Defines a set of component degrees of freedom sufficient to constrain free body motion of axisymmetric harmonic models.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SUPAX	RID1	HID1	DOF1	RID2	HID2	DOF2			

SUPAX	104	3	3						

Field	*Contents*	*Data Range*	*Default*
RIDi	Ring identification number.	Integer>0	Required
HIDi	Harmonic identification number.	Integer>0 or Harmonic Code	Required
DOFi	List of degrees of freedom.	DOF Code	Required

Remarks:

1. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.
2. The component **DOFi** specified are placed in the r-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SPC** or **OMIT**, that place them in mutually exclusive sets.

7 BULK

Bulk Data Entry SUPORT

Defines a set of component degrees of freedom sufficient to constrain free-body motion.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SUPORT	GID1	DOF1	GID2	DOF2	GID3	DOF3	GID4	DOF4	

SUPORT	116	126	145	51					

Field	*Contents*	*Data Range*	*Default*
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom.	DOF Code	Required

Remarks:

1. The component **DOFi** specified are placed in the r-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SPC** or **OMIT**, that place them in mutually exclusive sets.

7 BULK

Bulk Data Entry SUPORTS

Defines a set of component degrees of freedom in a Basic Substructure, sufficient to constrain free-body motion of a solution structure.

Format and Example:

1	2	3	4	5	6	7	8	9	10
SUPORTS	SNAME	GID1	DOF1	GID2	DOF2	GID3	DOF3		
SUPORTS	WING	101	123	257	156				

Field	*Contents*	*Data Range*	*Default*
SNAME	Basic Substructure name.	Name	Required
GIDi	GRID or SCALAR point identification numbers. [1]	Integer>0	Required
DOFi	List of degrees of freedom. [1]	DOF Code	Required

Remarks:

1. All points referenced must exist in Basic Substructure **SNAME** and the component **DOF** specified must exist in the final solution structure.
2. The component **DOFi** specified are placed in the r-set. Therefore, they must not appear on Bulk Data entries such as **MPC**, **SPC** or **OMIT**, that place them in mutually exclusive sets.

7 BULK

Bulk Data Entry TABDMP1

Defines structural modal damping as a tabular function of frequency of the form:

$$g(f) = \mathbf{gi}(\mathbf{fi})$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABDMP1	TID	XAXIS	YAXIS	DMPTYP					-cont-
-cont-	f1	d1	f2	d2	*CONTINUES IN GROUPS OF 2*				-cont-
TABDMP1	101	LOG	LOG						+A
+A	1.0	.10	100.	.30	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number. [1]	Integer>0	Required
XAXIS	Specifies the method used for interpolation for x-axis data. [2]	Character { **LINEAR** / **LOG** }	**LINEAR**
YAXIS	Specifies the method used for interpolation for y-axis data. [2]	Character { **LINEAR** / **LOG** }	**LINEAR**
DMPTYP	Type of damping data. [3]	Character { **G** / **FRACTION** / **PERCENT** / **Q** }	**G**
fi	Frequency values in cycles per unit time. [4,5,6,7]	Real≠0.0	Required
di	Damping values. [2,3]	Real	Required

Remarks:

1. Structural modal damping tables must be selected in the Case Control packet with the command:

   SDAMP = TID

2. Values of structural modal damping are found by linear or logarithmic interpolation within, and extrapolation outside the table frequency range, using the last two **fi** entries at the appropriate table end. The method of interpolation is selected with the **XAXIS** and **YAXIS** fields. At jump points the average of the two **di** values is used.

3. The modal damping values must be defined in either of four available types: **PERCENT**age of critical damping; **FRACTION** of critical damping; twice the fraction of critical damping, **G**; or the reciprocal of **G**, called the damping ampllification factor, **Q**.

4. The **fi** values must be entered in either ascending or descending order.

7 BULK

5. Step functions, or jumps, where two consecutive **fi** entries are identical, but the **di** entries change, are allowed, but may not be entered at the end points.
6. Any **fi,di** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
7. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
8. Structural modal damping is used only in modal dynamic analyses.

Bulk Data Entry TABDMP2

Defines structural modal damping for a list of specific mode numbers or *h-set* degrees of freedom.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABDMP2	TID			DMPTYP	DEFDMP	INDEX			-cont-
-cont-	m1	d1	m2	d2	*CONTINUES IN GROUPS OF 2*				-cont-

TABDMP2	101					MODE			+A
+A	1	.10	3	.30	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number. [1]	Integer>0	Required
DMPTYP	Type of damping data. [2]	Character { **G** / **FRACTION** / **PERCENT** / **Q** }	**G**
DEFDMP	Default damping value to be used if a mode does not appear in the **mi** list specified below. [3]	Real≥0.0	**0.0**
INDEX	Defines the meaning of the Mode Index (**mi**) given below. A value of **MODE** indicates that the indices are mode identification numbers, while a value of **HSET** indicates that the indices are modal degrees of freedom in the *h-set*. [3]	Character { **MODE** / **HSET** }	**MODE**
mi	Mode index interpreted as above. [3]	Integer>0	Required
di	Damping value for **mi**. [2,3]	Real≥0.0	Required

Remarks:

1. Structural modal damping tables must be selected in the Case Control packet with the command:

   SDAMP = TID

2. The modal damping values must be defined in either of four available types: **PERCENT**age of critical damping; **FRACTION** of critical damping; twice the fraction of critical damping, **G**; or the reciprocal of **G**, called the damping ampllification factor, **Q**.
3. Values of structural modal damping are found by selection from the table for a specific mode index or *h-set* degree of freedom. For other modes or degrees of freedom, the damping is set to **DEFDMP**.
4. The **mi** values must be entered in either ascending or descending order.
5. Any **mi,di** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
6. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
7. Structural modal damping is used only in modal dynamic analyses.

Bulk Data Entry TABLEA1

Defines a set of Aerodynamic pressure weighting factors.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLEA1	TID								-cont-
-cont-	BOX1	WT1	BOX2	WT2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLEA1	101								-CONT-
-CONT-	1001	0.9	1002	0.8	1005	0.9	ENDT		

Field	*Comments*	*Data Range*	*Default*
LABEL	Table identification number. [1]	Integer>0	Required
BOXi	Aerodynamic Box identification numbers. [2]	Integer>0	blank
WTi	Aerodynamic pressure weighting factors. [2]	Real	1.0

Remarks:

1. The weighting factor table must be referenced by an **ACORFAC** Bulk Data entry.
2. Aerodynamic Box identification numbers that are not specified will have a weighting factor of 1.0.
3. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.

7 BULK

Bulk Data Entry TABLED1

Defines a tabular function, used in generating frequency-dependent and time-dependent dynamic loads, of the form:

$$y(X) = \mathbf{Yi}(\mathbf{Xi})$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLED1	TID	XAXIS	YAXIS						-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLED1	32								+A
+A	-3.0	6.9	2.0	5.6	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
XAXIS	Specifies the method used for interpolation for x-axis data. [1]	Character { **LINEAR** / **LOG** }	**LINEAR**
YAXIS	Specifies the method used for interpolation for y-axis data. [1]	Character { **LINEAR** / **LOG** }	**LINEAR**
Xi,Yi	Tabular entries. [1,2,3,4,5]	Real	Required

Remarks:

1. Values of **Yi** are found by linear or logarithmic interpolation within, and extrapolation outside the table frequency range, using the last two **Xi** entries at the appropriate table end. The method of interpolation is selected with the **XAXIS** and **YAXIS** fields. At jump points the average of the two **Yi** values is used.
2. The **Xi** must be entered in either ascending or descending order.
3. Step functions, or jumps, where two consecutive **Xi** entries are identical, but the **Yi** entries change, are allowed, but may not be entered at the end points.
4. Any **Xi,Yi** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
5. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.

7 BULK

Bulk Data Entry TABLED2

Defines a parametric tabular function, used in generating frequency-dependent and time-dependent dynamic loads, of the form:

$$y(X) = \mathbf{Yi}(\mathbf{Xi} - \mathbf{C1})$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLED2	TID	C1							-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLED2	115	15.6							+A
+A	1.0	-4.5	2.0	-4.5	2.0	2.8	ENDT		

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
C1	Function parameter.	Real	Required
Xi,Yi	Tabular entries. [1,2,3,4]	Real	Required

Remarks:

1. The **Xi** must be entered in either ascending or descending order.
2. Step functions, or jumps, where two consecutive **Xi** entries are identical, but the **Yi** entries change, are allowed, but may not be entered at the end points.
3. Any **Xi,Yi** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
4. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last entry.
5. Values of the dependent variable, **Y**, are found by linear interpolation within, and linear extrapolation outside the table **X** range using the two **Xi** points nearest the appropriate table end. At jump points the average of the two **Yi** values is used.

7 BULK

Bulk Data Entry TABLED3

Defines a parametric tabular function, used in generating frequency-dependent and time-dependent dynamic loads, of the form:

$$y(X) = \mathbf{Yi}\left(\frac{\mathbf{Xi} - \mathbf{C1}}{\mathbf{C2}}\right)$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLED3	TID	C1	C2						-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLED3	62	126.9	30.0						+A
+A	2.9	2.9	3.6	0.47	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
C1,C2	Function parameters.	Real **C2**≠0.0	Required
Xi,Yi	Tabular entries. [1,2,3,4,5]	Real	Required

Remarks:

1. The **Xi** must be entered in either ascending or descending order.
2. Step functions, or jumps, where two consecutive **Xi** entries are identical, but the **Yi** entries change, are allowed, but may not be entered at the end points.
3. Any **Xi,Yi** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
4. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
5. Values of the dependent variable, **Y**, are found by linear interpolation within, and linear extrapolation outside the table **X** range using the two **Xi** points nearest the appropriate table end. At jump points the average of the two **Yi** values is used.

7 BULK

Bulk Data Entry TABLED4

Defines the coefficients of a power series, used in generating frequency-dependent and time-dependent dynamic loads, of the form:

$$y(X) = \begin{cases} \sum_{i=0}^{N} \mathbf{Ai} \cdot \left(\frac{\mathbf{C3} - \mathbf{C1}}{\mathbf{C2}} \right)^{i} & \text{when } Xi < \mathbf{C3} \\ \sum_{i=0}^{N} \mathbf{Ai} \cdot \left(\frac{Xi - \mathbf{C1}}{\mathbf{C2}} \right)^{i} & \text{when } \mathbf{C3} \le Xi \le \mathbf{C4} \\ \sum_{i=0}^{N} \mathbf{Ai} \cdot \left(\frac{\mathbf{C4} - \mathbf{C1}}{\mathbf{C2}} \right)^{i} & \text{when } Xi > \mathbf{C4} \end{cases}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLED4	TID	C1	C2	C3	C4				-cont-
-cont-	A0	A1	A2	A3	A4	*CONTINUES*			-cont-

TABLED4	28	0.0	1.0	0.0	100.0				+A
+A	-3.0	6.9	2.0	5.6	0.0				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
Cj	Function parameters.	Real **C2**≠0.0 **C3<C4**	Required
Ai	Coefficient entries. [1]	Real	Required

Remarks:

1. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
2. There are *N+1* entries in the table.

7 BULK

Bulk Data Entry **TABLEM1** (Structural Analysis)

Defines a tabular function, used for generating temperature dependent material properties, of the form:

$$y(X) = \mathbf{Yi}(\mathbf{Xi})$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLEM1	TID								-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLEM1	32								+A
+A	-3.0	6.9	2.0	5.6	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
Xi,Yi	Tabular entries. [1,2,3,4]	Real	Required

Remarks:

1. The **Xi** values must be entered in either ascending or descending order.
2. Step functions, or jumps, where two consecutive **Xi** entries are identical, but the **Yi** entries change, are allowed, but may not be entered at the end points.
3. Any **Xi,Yi** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
4. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
5. Values of the dependent variable, **Y**, are found by linear interpolation within, and linear extrapolation outside the table **X** range using the two **Xi** points nearest the appropriate table end. At jump points the average of the two **Yi** values is used.

7 BULK

Bulk Data Entry TABLEM1 (Heat Transfer Analysis)

Defines a tabular function, used for generating temperature dependent material properties, of the form:

$$y(x) = Z \cdot \mathbf{Yi}(\mathbf{Xi})$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLEM1	TID								-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLEM1	32								+A
+A	-3.0	6.9	2.0	5.6	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
Xi,Yi	Tabular entries. [1,2,3,4,5]	Real	Required

Remarks:

1. The **Xi** values must be entered in either ascending or descending order.
2. Step functions, or jumps, where two consecutive **Xi** entries are identical, but the **Yi** entries change, are allowed, but may not be entered at the end points.
3. Any **Xi,Yi** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
4. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
5. Values of the dependent variable, **Y**, are found by linear interpolation within, and linear extrapolation outside the table **X** range using the two **Xi** points nearest the appropriate table end. At jump points the average of the two **Yi** values is used. *Z* is supplied from the corresponding **MATi** Bulk Data entry.

Bulk Data Entry TABLEM2

Defines a parametric tabular function, used in generating temperature dependent material properties of the form:

$$y(X) = Z \cdot \mathbf{Yi}(\mathbf{Xi} - \mathbf{C1})$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLEM2	TID	C1							-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLEM2	115	15.6							+A
+A	1.0	-4.5	2.0	-4.5	2.0	2.8	ENDT		

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
C1	Function parameter.	Real	Required
Xi,Yi	Tabular entries. [1,2,3,4,5]	Real	Required

Remarks:

1. The **Xi** values must be entered in either ascending or descending order.
2. Step functions, or jumps, where two consecutive **Xi** entries are identical, but the **Yi** entries change, are allowed, but may not be entered at the end points.
3. Any **Xi,Yi** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
4. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
5. Values of the dependent variable, **Y**, are found by linear interpolation within, and linear extrapolation outside the table **X** range using the two **Xi** points nearest the appropriate table end. At jump points the average of the two **Yi** values is used. *Z* is supplied from the corresponding **MATi** Bulk Data entry.

7 BULK

Bulk Data Entry TABLEM3

Defines a parametric tabular function, used in generating temperature dependent material properties of the form:

$$y(X) = Z \cdot \mathbf{Yi}\left(\frac{\mathbf{Xi} - \mathbf{C1}}{\mathbf{C2}} \right)$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLEM3	TID	C1	C2						-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF 2*				-cont-

TABLEM3	62	126.9	30.0						+A
+A	2.9	2.9	3.6	0.47	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Positive Integer	Required
C1,C2	Function parameters.	Real **C2**≠0.0	Required
Xi,Yi	Tabular entries. [1,2,3,4]	Real	Required

Remarks:

1. The **Xi** values must be entered in either ascending or descending order.
2. Step functions, or jumps, where two consecutive **Xi** entries are identical, but the **Yi** entries change, are allowed, but may not be entered at the end points.
3. Any **Xi,Yi** pair is ignored if the string **SKIP** appears in either of the two fields used for the pair.
4. The end of the table is indicated by entering the string **ENDT** in either of the two fields following the last value entry.
5. Values of the dependent variable, **Y**, are found by linear interpolation within, and linear extrapolation outside the table **X** range using the two **Xi** points nearest the appropriate table end. At jump points the average of the two **Yi** values is used. *Z* is supplied from the corresponding **MATi** Bulk Data entry.

7 BULK

Bulk Data Entry TABLEM4

Defines coefficients of a power series, used in generating temperature dependent material properties, of the form:

$$y(X) = \begin{cases} Z \cdot \sum_{i=0}^{N} \mathbf{Ai} \cdot \left(\frac{\mathbf{C3} - \mathbf{C1}}{\mathbf{C2}} \right)^{i} & \text{when } Xi < \mathbf{C3} \\ Z \cdot \sum_{i=0}^{N} \mathrm{Ai} \cdot \left(\frac{Xi - \mathbf{C1}}{\mathbf{C2}} \right)^{i} & \text{when } \mathbf{C3} \le Xi \le \mathbf{C4} \\ Z \cdot \sum_{i=0}^{N} \mathrm{Ai} \cdot \left(\frac{\mathbf{C4} - \mathbf{C1}}{\mathbf{C2}} \right)^{i} & \text{when } Xi > \mathbf{C4} \end{cases}$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLEM4	TID	C1	C2	C3	C4				-cont-
-cont-	A0	A1	A2	A3	*CONTINUES IN GROUPS OF 2*				-cont-

TABLEM4	28	1.0	0.0	100.	200.				+A
+A	-3.	6.7	2.0	5.6	ENDT				

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
Ci	Function parameters.	Real **C2**≠0.0 **C3<C4**	Required
Ai	Coefficient entries. [1,2]	Real	Required

Remarks:

1. The end of the table is indicated by placing the string **ENDT** in either of the two fields following the last value entry.
2. *Z* is supplied from the corresponding **MATi** Bulk Data entry.
3. There are *N+1* entries in the table.

7 BULK

Bulk Data Entry TABLENL

Defines a nonlinear *stress-strain* curve for all nonlinear structural elements, or a *force-deflection* curve for the ELASNL, BUSH elements and the soil component of the PILE element. The tabular forms are:

$$\sigma(\varepsilon) = \textbf{SCALE} \cdot \textbf{Yi}(\textbf{Xi}) \text{ or}$$

$$F(\delta) = \textbf{SCALE} \cdot \textbf{Yi}(\textbf{Xi})$$

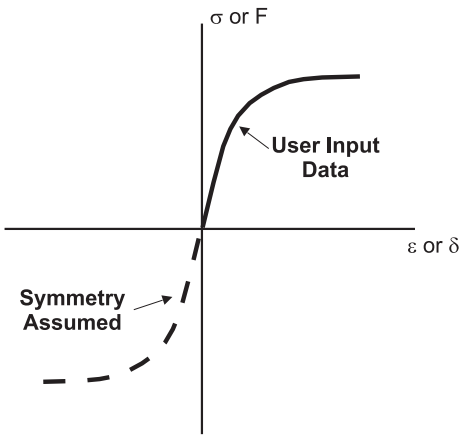
Format and Example:

1	2	3	4	5	6	7	8	9	10
TABLENL	TID	EU	SCALE	HYSTYP	YIELD	ϕ	β	EP	-cont-
-cont-	X1	Y1	X2	Y2	*CONTINUES IN GROUPS OF TWO*				-cont-

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
EU	Plastic unloading modulus.	Real	[2,7]
SCALE	Function scale factor.	Real	**1.0**
HYSTYP	Type of hysteretic material behavior. [8]	Character { **CYCLIC** **NONC** **ISOT** **KINE** **COMB** }	**ISOT**
YIELD	Yield stress value. [9]	Real	[9]
ϕ	The angle of internal friction, in degrees. [10]	0.0≤Real<45.0	**0.0**
β	Combination factor. [11]	0.0≤Real≤1.0	**0.5**
EP	Elastic modulus used in the plastic range.	Real	[2]
Xi,Yi	Tabular entries. [1,12,13,14,15,16,17,18]	Real	Required

Remarks:

1. In all cases, the data curves are assumed skew-symmetric about the vertical axis if your input begins at **Xi**≥**0.0** as shown in the adjacent figure.
2. For Plastic materials, the unloading modulus is defined by **EU**. If **EU** is blank the unloading modulus will be the Young's modulus, *E*. The default value for **EP** is *E* before the first unloading and **EP=EU** thereafter.
3. For Nonlinear Elastic materials, **EU** may be **0.0** or blank if the **NLELAST** option is specified on the **MAT1NL** entry referencing this table. For the ELASNL, BUSH, or PILE (soil component) element, **EU** must be **0.0**. In this case, both loading and unloading follow your defined curve.



7 BULK

4. For the ELASNL and BUSH elements, and the soil component of the PILE element, **EU=0.0** selects Nonlinear Elastic behavior, otherwise the behavior is Plastic.

5. For Nonlinear Elastic materials and for other materials with a **HYSTYP** option of **NONC**, the stress-strain or force-deflection data may be defined in the negative region of the strain or deflection. This will produce unsymmetric behavior in tension and compression.

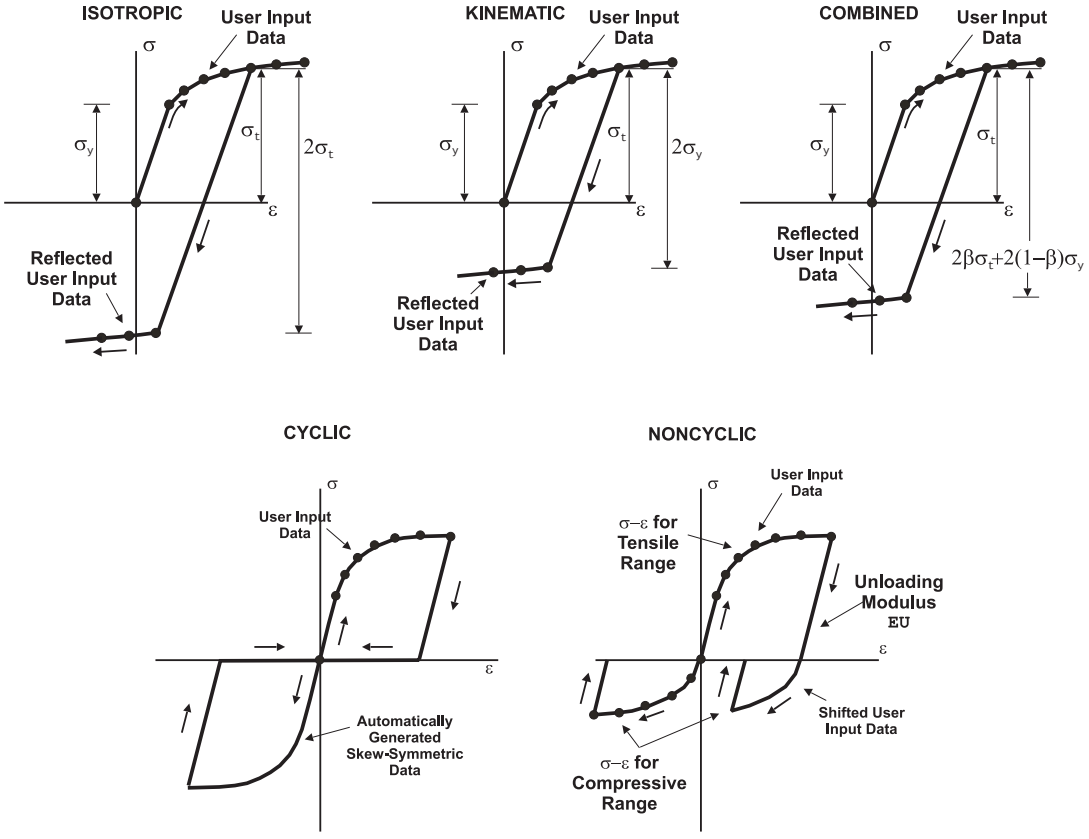
6. Combined Nonlinear Elastic and Plastic materials are only available with structural elements, and are requested with the **EPCOMB** option on the **MAT1NL** entry. In this case the behavior is nonlinear elastic until the yield stress value is reached for the first time, thereafter it becomes plastic.

7. For combined Nonlinear Elastic and Plastic materials, the default value for **EU** is calculated as the **YIELD** value divided by the strain corresponding to the **YIELD**. For more information, see section 16.4.5 of the User's Guide.

8. For Plastic materials, there are five types of hysteretic material behavior, **HYSTYP**, which are:

   - **ISOT** - Isotropic hardening
   - **KINE** - Kinematic hardening
   - **COMB** - Combined isotropic and kinematic hardening
   - **CYCLIC** - Cyclic loading exhibiting a "gap" effect
   - **NONC** - Noncyclic loading

   For Combined Nonlinear Elastic and Plastic materials, only **ISOT**, **KINE**, and **COMB** are available. **CYCLIC** and **NONC** behavior are available only for ELASNL, ROD, BAR, BEAM, and PILE elements. The five material behaviors are illustrated in the following figures. For additional information, see Chapter 16 of the ***User's Guide***.



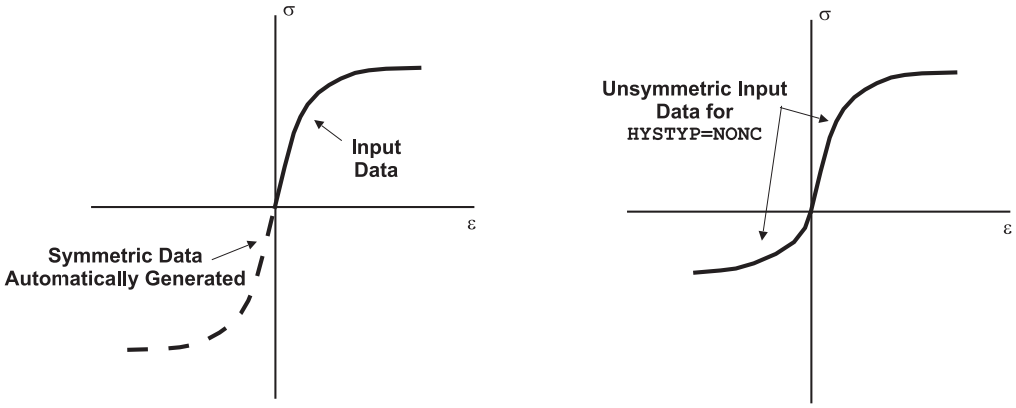
9. For Plastic materials, the yield value is the **Yi** value at the first positive **Xi** value in the table. For Combined Elastic and Plastic materials (the **EPCOMB** option on **MAT1NL** Bulk Data entry), the **YIELD** value is required. The **YIELD** value normally represents the yield stress, except when the yield criterion (**YC**) selected on the corresponding **MAT1NL** entry is **MC** or **DP**, in which case it represents cohesion in stress units.

10. The angle of internal friction is used in conjunction with the cohesion value used for yield criteria **MC** or **DP**, as entered on the **MAT1NL** entry.

11. The combination factor, β, is used with the **COMB** hardening rule and controls the level of combination. A value of **0.0** will result in **KINE**matic hardening, while **1.0** represents **ISOT**ropic hardening.

12. The **Xi** must be entered in ascending order.

13. The tabular entries must form a continuous function, which represents a stress/cohesion-strain curve, except for the nonlinear spring properties in the ELASNL and BUSH elements, and the soil portion of the PILE element, where it represents a force-deflection curve.



14. Values of the dependent variable, **Y**, are found by linear interpolation within, and linear extrapolation outside the table **X** range using the two **Xi** points nearest the appropriate table end.

15. If **HYSTYP** is **ISOT**, **KINE**, **COMB**, or **CYCLIC**, then the **Xi** values must be zero or positive. In this case, symmetric tabular data for **Xi<0.0** are automatically generated, i.e. the **Yi** values are the negative of the input **Yi** values, as shown in the left portion of the figure below.

    For **HYSTYP=NONC**, the input values of **Xi** may be either positive or negative. This results in unsymmetric tabular data as shown in the right portion of the figure. However, if all **Xi** values are non-negative, then the symmetric reflection will be automatically generated.

16. Except at **X=0.0**, the slope at any **Xi** is the average of the slopes immediately before and after that point. At **X=0.0**, the slope is set to the slope immediately after **X=0.0**, and must be equal to the value of Young's modulus on the corresponding **MAT1** Bulk Data entry.

17. Any **Xi,Yi** pair may be ignored by placing the string **SKIP** in either of the two fields used for that pair.

18. The end of the table is indicated by placing the string **ENDT** in either of the two fields following the last value entry.

Bulk Data Entry TABRND1

Defines power spectral density factors for use in random analysis, as a tabular function of frequency of the form:

$$PSD(f) = \mathbf{gi}\left(\mathbf{fi}\right)$$

Format and Example:

1	2	3	4	5	6	7	8	9	10
TABRND1	TID	XAXIS	YAXIS						-cont-
-cont-	f1	g1	f2	g2	*CONTINUES IN GROUPS OF 2*				-cont-

TABRND1	101	LOG	LOG						+A
+A	.1	1.5	100.	7.5	10000.	4.0	ENDT		

Field	*Contents*	*Data Range*	*Default*
TID	Table identification number.	Integer>0	Required
XAXIS	Specifies the method used for interpolation for x-axis data. [1]	Character { **LINEAR** / **LOG** }	**LINEAR**
YAXIS	Specifies the method used for interpolation for y-axis data. [1]	Character { **LINEAR** / **LOG** }	**LINEAR**
fi	Frequency values in cycles per unit time. [1,2,3,4]	Real>0.0	Required
gi	Power Spectral Density values. [1,3,4]	Real	Required

Remarks:

1. Values of the Power Spectral Density function are found by linear or logarithmic interpolation within, and extrapolation outside the table frequency range, using the last two **fi** entries at the appropriate table end. The method of interpolation is selected with the **XAXIS** and **YAXIS** fields. At jump points the average of the two **gi** values is used.
2. The **fi** must be entered in ascending order.
3. Step functions, or jumps, where two consecutive **fi** entries are identical, but the **gi** entries change, are allowed, but may not be entered at the end points.
4. The end of the table is indicated by the string **ENDT** in either of the two fields following the last value entry.

7 BULK

Bulk Data Entry TEMP

Defines a temperature set at GRID points.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TEMP	SID	GID1	T1	GID2	T2	GID3	T3		
TEMP	104	191	234.6	150	374.6				

Field	*Contents*	*Data Range*	*Default*
SID	Temperature set identification number. [1,2,3]	Integer>0	Required
GIDi	GRID point identification numbers.	Integer>0	Required
Ti	Temperatures.	Real	Required

Remarks:

1. Temperature sets must be selected in the Case Control packet with the command:

   TEMP = SID

   They are used in calculations involving equivalent thermal loading, temperature-dependent material properties and stress data recovery.

2. **SID** must be unique with respect to all other **LOAD** type data if the

   TEMP(LOAD)

   command appears in the Case Control packet.

3. In Transient Heat Transfer analyses, non-zero initial temperatures defined on this Bulk Data entry must be selected in the Case Control packet with the command:

   IC = SID

4. If thermal effects are requested, all elements must have a temperature field defined either directly by **TEMPP1**, **TEMPP2**, **TEMPP3** or **TEMPRB** data, or indirectly as the average of the connected GRID point temperatures defined by **TEMP** or **TEMPD** data. Directly defined element temperatures always take precedence over the average of GRID point temperatures.

5. If the element material is temperature-dependent, its properties are evaluated at the average temperature. Average element temperatures are obtained as a simple average of the connecting GRID point temperatures when element temperature data are not defined.

Bulk Data Entry TEMPAX

Defines a temperature set for an axisymmetric harmonic model.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TEMPAX	SID	RID	ϕ	TEMP					
TEMPAX	106	14	65.0	165.					

Field	*Contents*	*Data Range*	*Default*
SID	Temperature set identification number. [1,2]	Integer>0	Required
RID	Ring identification number. [3]	Integer>0	Required
f	Azimuthal angle in degrees. [3]	Real	Required
TEMP	Temperature value.	Real	Required

Remarks:

1. Temperature sets must be selected in the Case Control packet with the command:

   TEMP = SID

   They are used in calculations involving equivalent thermal loading, temperature-dependent material properties and stress data recovery.

2. **SID** must be unique with respect to all other **LOAD** type data if the command

   TEMP(LOAD)

   appears in the Case Control packet.

3. At least two different angles are required for each **SID** and **RID** to specify the subtended angle over which the temperature applies.

4. This entry is allowed only if an **AXIC** entry is also present in the Bulk Data packet.

Bulk Data Entry TEMPD

Defines a temperature default for all GRID points of the structural model for which temperatures have not been explicitly defined.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TEMPD	SID1	T1	SID2	T2	SID3	T3	SID4	T4	
TEMPD	102	276.5							

Field	*Contents*	*Data Range*	*Default*
SIDi	Temperature set identification numbers. [1,2,3]	Integer>0	Required
Ti	Default temperature values.	Real	Required

Remarks:

1. Temperature sets must be selected in the Case Control packet with the command:

   TEMP = SID

   They are used in calculations involving equivalent thermal loading, temperature-dependent material properties and stress data recovery.

2. **SID** must be unique with respect to all other **LOAD** type entries if the command

   TEMP(LOAD)

   appears in the Case Control packet.

3. In Transient Heat Transfer analyses, non-zero initial temperatures defined on this Bulk Data entry must be selected in the Case Control packet with the command:

   IC = SID

4. If thermal effects are requested, all elements must have a temperature field defined either directly by **TEMPP1**, **TEMPP2**, **TEMPP3** or **TEMPRB** entries, or indirectly as the average of the connected GRID point temperatures defined by **TEMP** or **TEMPD** entries. Directly defined element temperatures always take precedence over the average of GRID point temperatures.

5. If the element material is temperature-dependent, its properties are evaluated at the average temperature. Average element temperatures are obtained as a simple average of the connecting GRID point temperatures when element temperature data are not defined.

Bulk Data Entry TEMPP1

Defines a temperature field for the plate and shell elements of the form:

$$T(z) = \mathbf{TBAR} + \mathbf{T'} \cdot z$$

Format and Examples:

1	2	3	4	5	6	7	8	9	10
TEMPP1	SID	EID1	TBAR	T′	T1	T2			-cont-
-cont-	*ELEMENT ID LIST*								-cont-

TEMPP1	2	24	62.0	10.0	57.0	67.0			+A
+A	1	THRU	19	30	THRU	100			

Field	*Contents*	*Data Range*	*Default*
SID	Temperature set identification number. [1,2,3]	Integer>0	Required
EIDi	Element identification numbers.	Integer>0	Required
TBAR	Average temperature over the cross-section. [4]	Real	Required
T′	Effective linear thermal gradient. [5]	Real	**0.0**
T1,T2	Temperature values at stress recovery fibers.	Real	[6]
ELEMENT ID LIST	List of one or more element identification numbers.	Integer>0	Required

Remarks:

1. Temperature sets must be selected in the Case Control packet with the command:

   TEMP = SID

   They are used in calculations involving equivalent thermal loading, temperature-dependent material properties and stress data recovery.

2. **SID** must be unique with respect to all other load data if the commnad

   TEMP(LOAD)

   appears in the Case Control packet.

3. In Transient Heat Transfer analyses, non-zero initial temperatures defined on this Bulk Data entry must be selected in the Case Control packet with the command:

   IC = SID

4. The *average* temperature, **TBAR**, for a homogeneous plate with volume V is:

   $$\mathbf{TBAR} = \frac{1}{V}\int_V T\, dV$$

   TBAR is assumed constant over the area of the element.

7 BULK

5. For a temperature field other than a constant gradient, the *effective gradient* for a homogeneous plate is:

$$\mathbf{T}' = \frac{1}{I}\int T(z)\, z\, dz$$

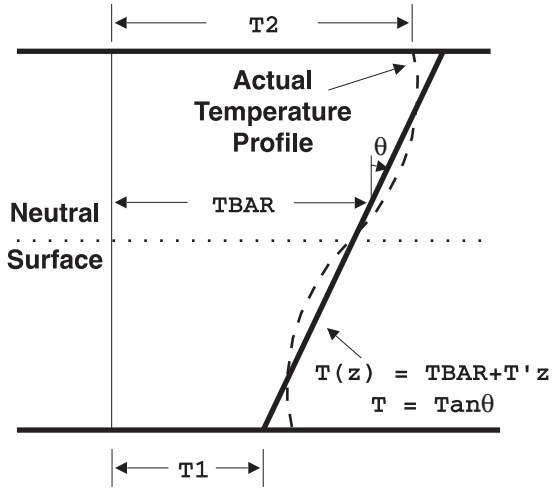
where I is the bending inertia, and z is the fiber distance from the neutral surface in the positive normal direction. **T′** is not used for membrane behavior.

6. If not specified, **T1** and **T2** are calculated using **TBAR**, **T′**, and the stress recovery fiber distance values.

7. If the element material is temperature-dependent, its properties are evaluated at the average temperature **TBAR**.

8. If thermal effects are requested, all elements must have a temperature field defined either directly by **TEMPP1**, **TEMPP2**, **TEMPP3** or **TEMPRB** entries, or indirectly as the average of the connected GRID point temperatures defined by **TEMP** or **TEMPD** entries. Directly defined element temperatures always take precedence over the average of GRID point temperatures.

9. The figure defines the input temperature profile.



Bulk Data Entry TEMPP2

Defines a temperature field for the plate and shell elements as an average temperature and thermal moments.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
TEMPP2	SID	EID1	TBAR	MX	MY	MXY	T1	T2	-cont-
-cont-	*ELEMENT ID LIST*								

TEMPP2	2	36	66.8						-cont-
-cont-	1	2	3	40	THRU	50			

Field	*Contents*	*Data Range*	*Default*
SID	Temperature set identification number. [1,2,3]	Integer>0	Required
EIDi	Element identification numbers.	Integer>0	Required
TBAR	Average temperature over the cross-section. [4]	Real	Required
Mij	Resultant thermal moments per unit width in element coordinate system. [5]	Real	Required
T1,T2	Temperature values at stress recovery fibers.	Real	Required
ELEMENT ID LIST	List of one or more element identification numbers.	Integer>0	Required

Remarks:

1. Temperature sets must be selected in the Case Control packet with the command:

   TEMP = SID

   They are used in calculations involving equivalent thermal loading, temperature-dependent material properties and stress data recovery.

2. **SID** must be unique with respect to all other load data if the command

   TEMP(LOAD)

   appears in the Case Control packet.

3. In Transient Heat Transfer analyses, non-zero initial temperatures defined on this Bulk Data entry must be selected in the Case Control packet with the command:

   IC = SID

7 BULK

4. The *average* temperature, **TBAR**, for a homogeneous plate with volume V is:

$$\mathbf{TBAR} = \frac{1}{V}\int_V T\,dV$$

TBAR is assumed constant over the area of the element.

5. The thermal moments in the element coordinate system may be calculated from the equation:

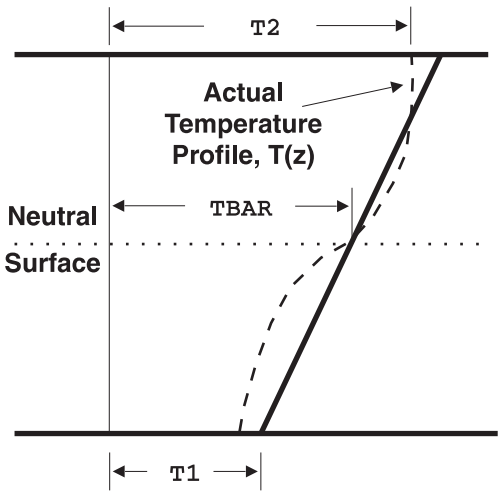
$$\begin{Bmatrix} \mathbf{MX} \\ \mathbf{MY} \\ \mathbf{MXY} \end{Bmatrix} = -\int G_e\,\alpha_e\,T(z)\,z\,dz$$

where the integration is performed over the bending material properties specified by **MID2** on the **PSHELL** entry, in the element coordinate system, G_e is the elastic coefficient matrix, α_e are the material thermal expansion coefficients, and $T(z)$ is the temperature at z, the distance from the neutral surface in the element coordinate system.

6. The temperature dependent material properties are evaluated at the average temperature T. If a property varies with depth, an effective value must be used which satisfies the desired elastic and stress relationships. The temperatures at the fiber distances may be changed to compensate for local differences in material properties and produce correct stresses.

7. If thermal effects are requested all elements must have a temperature field defined either directly by **TEMPP1**, **TEMPP2**, **TEMPP3** or **TEMPRB** entries, or indirectly as the average of the connected GRID point temperatures defined by **TEMP** or **TEMPD** entries. Directly defined element temperatures always take precedence over the average of GRID point temperatures.

8. The figure defines the input temperature profile.



Bulk Data Entry TEMPP3

Defines a temperature field for the plate and shell elements of the tabular form:

$$T(z) = \mathbf{Ti}(\mathbf{zi})$$

Format and Examples:

1	2	3	4	5	6	7	8	9	10
TEMPP3	SID	EID1	z0	T0	z1	T1	z2	T2	-cont-
-cont-	z3	T3	z4	T4	z5	T5	z6	T6	-cont-
-cont-	z7	T7	z8	T8	z9	T9	z10	T10	-cont-
-cont-	*ELEMENT ID LIST*								-cont-

TEMPP3	17	1	0.0	32.9	.1	43.4	.2	45.0	+A
+A	.3	60.0	.4	90.					+B
+B									+C
+C	2	10	20	30	40	50	60	70	+D
+D	100	THRU	200						

Field	*Contents*	*Data Range*	*Default*
SID	Temperature set identification number. [1,2,3]	Integer>0	Required
EID1	Element identification number.	Integer>0	Required
z0	Position of the bottom surface with respect to an arbitrary reference plane.	Real	Required
zi	Positions through the thickness relative to the arbitrary reference plane. [4]	Real	Required
T0	Temperature at the bottom surface.	Real	Required
Ti	Temperature at position **zi**.	Real	Required
ELEMENT ID LIST	List of one or more element identification numbers.	Integer>0	Required

Remarks:

1. Temperature sets must be selected in the Case Control packet with the command:

   TEMP = SID

   They are used in calculations involving equivalent thermal loading, temperature-dependent material properties and stress data recovery.

2. **SID** must be unique with respect to all other load data if the commnad

   TEMP(LOAD)

   appears in the Case Control packet.

7 BULK

3. In Transient Heat Transfer analyses, non-zero initial temperatures defined on this Bulk Data entry must be selected in the Case Control packet with the command:

   IC = SID

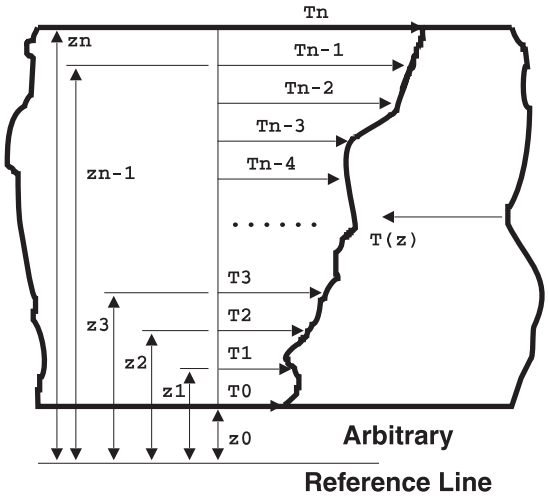
4. **zi** must be in increasing order with the last entered value corresponding to the top surface.

5. Data on **TEMPP3** entries are processed to simulate a linear distribution of temperature through the element thickness. The temperatures defined at the two element surfaces are used in element stress recovery calculations.

6. If thermal effects are requested, all elements must have a temperature field defined either directly with **TEMPP1**, **TEMPP2**, **TEMPP3**, or **TEMPRB** entries, or indirectly as the average of the connected GRID point temperatures defined with **TEMP** or **TEMPD** entries. Directly defined element temperatures always take precedence over the average of GRID point temperatures.

7. If the element material is temperature dependent, its properties are evaluated at the average temperature over the depth which is calculated from the simulated linear distribution.

8. The figure defines the input temperature profile.



7 BULK

Bulk Data Entry TEMPRB

Defines a temperature field for the BAR, BEAM, ROD, TUBE and CONROD elements.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TEMPRB	SID	EID1	TBARA	TBARB	T'1A	T'1B	T'2A	T'2B	-cont-
-cont-	TCA	TDA	TEA	TFA	TCB	TDB	TEB	TFB	-cont-
-cont-	*ELEMENT ID LIST*								-cont-

TEMPRB	105	102	77.2	56.5	0.0	23.5			+A
+A	62.5	85.0	39.2		42.	82.	25.		+B
+B	1002	1003	105	457					

Field	*Contents*	*Data Range*	*Default*
SID	Temperature set identification number. [1,2,3]	Integer>0	Required
EID1	Element identification number.	Integer>0	Required
TBARj	Average temperature over the cross-sectional area at End **j**.	Real	Required
T'ij	Effective linear gradient in direction **i** at End **j**. [4]	Real	**0.0**
Tkj	Temperatures at point **k** as defined on the property entries at end **j**. [4]	Real	**0.0**
ELEMENT ID LIST	Element identification numbers in a range definition.	Integer>0	Required

Remarks:

1. Temperature sets must be selected in the Case Control packet with the command:

   TEMP = SID

   They are used in calculations involving equivalent thermal loading, temperature-dependent material properties and stress data recovery.

2. **SID** must be unique with respect to all other load data if the command

   TEMP(LOAD)

   appears in the Case Control packet.

3. In Transient Heat Transfer analyses, non-zero initial temperatures defined on this Bulk Data entry must be selected in the Case Control packet with the command:

   IC = SID

4. The **T'ij** and **Tkj** fields are only used for the BAR and BEAM elements. If at least one nonzero or nonblank **Tkj** is present, the point temperatures given are used for stress recovery. If no **Tkj** values are given, linear temperature gradients are assumed for stresses.

5. If thermal effects are requested, all elements must have a temperature field defined either directly with **TEMPP1**, **TEMPP2**, **TEMPP3** or **TEMPRB** entries, or indirectly as the average of the connected GRID point temperatures defined by **TEMP** or **TEMPD** entries. Directly defined element temperatures always take precedence over the average of GRID point temperatures.
6. If the element material is temperature dependent, the material properties are evaluated at the average temperature of End A and End B.

Bulk Data Entry TF

Defines a transfer function of the form:

$$(\mathbf{B0} + \mathbf{B1}\, p + \mathbf{B2}\, p^2)\, u_d + \sum_i (\mathbf{A0}(i) + \mathbf{A1}(i)\, p + \mathbf{A2}(i)\, p^2)\, u_i = 0$$

which may be used as an alternate means of direct matrix input.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TF	SID	DGID	DDOF	B0	B1	B2			-cont-
-cont-	GID1	DOF1	AO(1)	A1(1)	A2(1)				-cont-
-cont-	GID2	DOF2	AO(2)	A1(2)	A2(2)				-cont-

TF	510	432	4	3.2	6.7	15.2			+A
+A	298	1	2.3	9.4	4.7				
+B	299	3	1.1	0.0	2.5				

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
DGID	Identification number of the dependent GRID, SCALAR or EXTRA point. [2,3]	Integer>0	Required
DDOF	Single dependent degree of freedom. [2,3]	DOF Code	Required
GIDi	Identification numbers of independent GRID, SCALAR or EXTRA points.	Integer>0	Required
DOFi	Single independent degree of freedom.	DOF Code	Required
Bj,Aj(i)	Transfer function coefficients.	Real	Required

Remarks:

1. Transfer Function sets must be selected in the Case Control packet with the command:

   TFL = SID

2. In dynamic response analyses, u_d is placed in the *e-set*. Therefore, **DGID** must reference an EXTRA point and **DDOF** must be left blank.
3. The constraint relation given by this function requires that the dependent degree of freedom not otherwise belong to the dependent set.
4. The matrix elements defined by this entry are added to the dynamic matrices for the problem.

7 BULK

Bulk Data Entry TIC

Defines initial displacements and velocities for Direct or Modal Transient Response analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TIC	SID	GID	DOF	U0	V0				
TIC	102	745	4	4.5	-6.0				

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1,2,3]	Integer>0	Required
GID	GRID, SCALAR or EXTRA point identification number for *d-set* (physical coordinate) input, or Mode Number for *h-set* (modal coordinate) input.	Integer>0	Required
DOF	Single degree of freedom for *d-set* input or 1 for *h-set* input.	DOF Code	Required
U0	Initial displacement value.	Real	**0.0**
V0	Initial velocity value.	Real	**0.0**

Remarks:

1. Transient initial condition sets must be selected in the Case Control packet with the command:

   IC = SID

   Initial conditions are the sum of all data from **TIC**, **TICRV**, and **TICTV** Bulk Data entries.

2. Initial conditions for components not specified on **TIC**, **TICRV**, and **TICTV** entries are assumed zero. Therefore, if an **IC** command is not present in the Case Control packet, all initial conditions are assumed zero.

3. Options on the Case Control command **IC** may be used to cause the automatic calculation of initial conditions when applied loads are non-zero at t=0. Also, initial conditions may also be computed automatically for Modal Transient Response analyses using Bulk Data entry **PARAM,NRB**.

7 BULK

Bulk Data Entry TICRV

Defines rotational (and translational) velocity initial conditions for the structural model for Direct or Modal Transient Response analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TICRV	SID	GID	CID	A	V1	V2	V3	GID1	-cont-
-cont-	*GRID ID LIST*								

TICRV	2	100	0	1.3	1.0	0.0	0.0	1	
	THRU	946							

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1,2,3]	Integer>0	Required
GID	GRID identification number to define a point on the axis of rotation. [4]	Integer≥0	**0**
CID	Coordinate system identification number in which **V** is defined.	Integer≥0	**0**
A	The value of rotational velocity.	Real	Required
Vi	Components of a vector, in coordinate system **CID**, defining the direction of rotation.	Real	**1.,0.,0.**
GID1, ***GRID ID LIST***	Initial GRID point, and list of one or more GRID points, which will have velocity initial conditions computed.	Integer>0	Required

Remarks:

1. Transient initial condition sets must be selected in the Case Control packet with the command:

   IC = SID

   Initial conditions are the sum of all data from **TIC**, **TICRV**, and **TICTV** Bulk Data entries.

2. Initial conditions for components not specified on **TIC**, **TICRV**, and **TICTV** entries are assumed zero. Therefore, if an **IC** command is not present in the Case Control packet, all initial conditions are assumed zero.

3. Options on the Case Control command **IC** may be used to cause the automatic calculation of initial conditions when applie loads are non-zero at t=0. Also, initial conditions may also be computed automatically for Modal Transient Response analyses using Bulk Data entry **PARAM,NRB**.

4. If **GID** is not input, the origin of the basic coordinate system is used.

7 BULK

Bulk Data Entry TICS

Defines initial displacements and velocities for a Basic Substructure in direct transient response analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TICS	SID	SNAME	GID	DOF	U0	V0			
TICS	102	SPAR	745	4	4.5	-6.0			

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
SNAME	Basic Substructure name.	Name	Required
GID	GRID, SCALAR or EXTRA point identification number.	Integer>0	Required
DOF	List of Degrees of freedom. [2]	DOF Code	Required
U0	Initial displacement value. [3]	Real	**0.0**
V0	Initial velocity value. [3]	Real	**0.0**

Remarks:

1. Transient initial condition sets must be selected in the Case Control packet with the command:

   IC = SID

2. Initial conditions may only be applied to the analysis degrees of freedom in the solution substructure.
3. Initial conditions for components not specified on **TICS** entries will be assumed zero. Therefore, if an **IC** command is not present in the Case Control packet, all initial conditions are assumed zero.
4. **TICS** data are used in the substructure SOLVE operation.

7 BULK

Bulk Data Entry TICTV

Defines translational velocity initial conditions for the structural model for Direct or Modal Transient Response analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TICTV	SID		CID	A	V1	V2	V3	GID1	-cont-
-cont-	*GRID ID LIST*								

TICTV	1000		1	10.0	1.0	0.0	0.0	1	
	THRU	3399							

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1,2]	Integer>0	Required
CID	Coordinate system identification number in which **V** is defined.	Integer≥0	**0**
A	The value of translational velocity.	Real	Required
Vi	Components of a vector, in coordinate system **CID**, defining the direction of translational velocity.	Real	**1.,0.,0.**
GID1, ***GRID ID LIST***	Initial GRID point, and list of one or more GRID points, which will have velocity initial conditions computed.	Integer>0	Required

Remarks:

1. Transient initial condition sets must be selected in the Case Control packet with the command:

   IC = SID

   Initial conditions are the sum of all data from **TIC**, **TICRV**, and **TICTV** Bulk Data entries.

2. Initial conditions for components not specified on **TIC**, **TICRV**, and **TICTV** entries are assumed zero. Therefore, if an **IC** command is not present in the Case Control packet, all initial conditions are assumed zero.

3. Options on the Case Control command **IC** may be used to cause the automatic calculation of initial conditions when applie loads are non-zero at t=0. Also, initial conditions may also be computed automatically for Modal Transient Response analyses using Bulk Data entry **PARAM,NRB**.

Bulk Data Entry TLOAD1

Defines a time-dependent dynamic load of the form:

$$P(t) = A \cdot F(t - \tau)$$

for use in transient response analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TLOAD1	LID	ADEF	τDEF	DYNEX	FTAB				

TLOAD1	101	201	301		4				

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
ADEF	Identification number of loads data which defines *A*. [3,4]	Integer>0	Required
tDEF	Identification number of a **DELAY** or **DELAYS** set which defines τ.	Integer≥0	[5]
DYNEX	Type of dynamic excitation. [6]	0ôInteger≥7	**0**
FTAB	Identification number of a **TABLEDi** entry which gives *F*(t-τ).	Integer>0	Required

Remarks:

1. Dynamic load sets must be selected in the Case Control packet with the command:

   DLOAD = LID

2. **LID** must be unique for all **DLOAD**, **TLOAD1**, and **TLOAD2** Bulk Data entries. **TLOAD1** loads may be combined with **TLOAD2** loads only by using a **DLOAD** Bulk Data entry.

3. For structural analyses, **ADEF** may reference any combination of **DAREA**, **DAREAS**, **FORCEi**, **MOMENTi**, **PLOADi**, **LOADC**, **GRAV** and **RFORCE** Bulk Data entries. When **ADEF** references **LOADC** data, **DAREAS** entries with the same set identification and non-zero loads must also exist.

4. For heat transfer analyses, **ADEF** may reference any combination of **QHBDY**, **QBDY1**, **QBDY2**, **QVECT**, and **QVOL** Bulk Data entries. The referenced **QVECT** data entry may also contain references to functions of time, and therefore **A**, as defined by **ADEF**, may be a function of time.

5. If τ**DEF** is blank or zero, τ will be zero.

7 BULK

6. The table defines the **DYNEX** dynamic excitation options which are available:

DYNEX	EXCITATION FUNCTION	DYNEX	EXCITATION FUNCTION
0	**FORCEi, MOMENTi,** or **PLOADi**	**4**	Fluid Pressure
1	Enforced Displacement	**5**	Fluid Volumetric Flux
2	Enforced Velocity	**6**	Fluid Volumetric Velocity
3	Enforced Acceleration	**7**	Fluid Volumetric Acceleration

When any enforced motion option is selected (**DYNEX** of **1**, **2**, or **3**), you must apply the dynamic excitation to *s-set* degrees of freedom.

Bulk Data Entry TLOAD2

Defines a time-dependent dynamic load of the form:

$$P(t) = \begin{cases} 0 & \text{when} \quad \bar{t} < 0 \ \text{ or } \ \bar{t} > \mathbf{t2}-\mathbf{t1} \\ A\,\bar{t}^{\,\mathbf{B}}\,e^{\,\mathbf{C}\bar{t}}\cos(2\pi\,\mathbf{f}\,\bar{t}+\theta) & \text{when} \quad 0 \le \bar{t} \le \mathbf{t2}-\mathbf{t1} \end{cases}$$

$$\text{where} \quad \bar{t} = t - \mathbf{t1} - \tau$$

for use in transient response analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TLOAD2	LID	ADEF	τDEF	DYNEX	t1	t2	f	θ	-cont-
-cont-	C	B							

TLOAD2	101	401	402		2.5	6.5	9.5	35.2	+A
+A	3.0	1.0							

Field	*Contents*	*Data Range*	*Default*
LID	Load set identification number. [1,2]	Integer>0	Required
ADEF	Identification number of loads data which defines *A*. [3,4]	Integer>0	Required
tDEF	Identification number of a **DELAY** or **DELAYS** set which defines τ. [5]	Integer>0	**0**
DYNEX	Type of dynamic excitation. [6]	0ôInteger≥7	**0**
t1	Time constant.	Real≥0.0	**0.0**
t2	Time constant.	Real>**t1**	Required
f	Frequency in cycles per unit time.	Real>0.0	Required
q	Phase angle in degrees.	Real	**0.0**
C	Exponential Coefficient.	Real	**0.0**
B	Growth coefficient.	Real	**0.0**

Remarks:

1. Dynamic load sets must be selected in the Case Control packet with the command:

   DLOAD = LID

2. **LID** must be unique for all **DLOAD**, **TLOAD1**, and **TLOAD2** Bulk Data entries. **TLOAD1** loads may be combined with **TLOAD2** loads only by using a **DLOAD** Bulk Data entry.

7 BULK

3. For structural analyses, **ADEF** may reference any combination of **DAREA**, **DAREAS**, **FORCEi**, **MOMENTi**, **PLOADi**, **LOADC**, **GRAV** and **RFORCE** Bulk Data entries. When **ADEF** references **LOADC** data, **DAREAS** entries with the same set identification and non-zero loads must also exist.

4. For heat transfer analyses, **ADEF** may reference any combination of **QHBDY**, **QBDY1**, **QBDY2**, **QVECT**, and **QVOL** Bulk Data entries. The referenced **QVECT** data entry may also contain references to functions of time, and therefore **ADEF** may be a function of time.

5. If τ**DEF** is blank or zero, τ will be zero.

6. The table defines the **DYNEX** dynamic excitation options which are available:

DYNEX	EXCITATION FUNCTION	DYNEX	EXCITATION FUNCTION
0	**FORCEi, MOMENTi,** or **PLOADi**	**4**	Fluid Pressure
1	Enforced Displacement	**5**	Fluid Volumetric Flux
2	Enforced Velocity	**6**	Fluid Volumetric Velocity
3	Enforced Acceleration	**7**	Fluid Volumetric Acceleration

When any enforced motion option is selected (**DYNEX** of **1**, **2**, or **3**), you must apply the dynamic excitation to *s-set* degrees of freedom.

Bulk Data Entry TRANS

Defines the Basic Coordinate System of a component substructure relative to the Basic Coordinate System of the combined substructure.

Format and Example:

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

TRANS	101		0.0	0.0	0.0	-0.5	1.0	10.0	+A
+A	0.0	10.0	0.5						

Field	*Contents*	*Data Range*	*Default*
CID	Coordinate system identification number. [1,2]	Integer>0	Required
Ai,Bi, Ci	Coordinates, in the Basic Coordinate System of the combined substructure, of points defining the orientation of the Basic Coordinate System of a component substructure. [3]	Real	Required

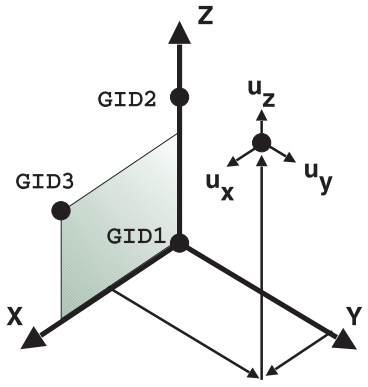
Remarks:

1. Transformation sets for a whole substructure must be selected in the Substructure Control packet as part of the COMBINE operation by using the subcommand:

   TRANS = CID

2. The value of **CID** must be unique with respect of all other **TRANS** Bulk Data entries.

3. The coordinates of three points **A**, **B**, **C** must be expressed in the Basic Coordinate System of the resultant combined substructure. Point **A** defines the location of the origin of the Basic Coordinate System of the component substructure. Point **B** defines the location of a point on the z-axis, and point **C** defines the location of a point in the positive x-side of the xz-plane, as illustrated.



4. Transformation of individual GRID points in a substructure prior to combining them is requested with **GTRAN** Bulk Data entries which, in turn, reference **TRANS** entries.

7 BULK

Bulk Data Entry TSTEP

Defines time step intervals at which the solution will be generated in transient response analyses.

Format and Example:

1	2	3	4	5	6	7	8	9	10
TSTEP	SID	N1	DT1	NO1					-cont-
-cont-		N2	DT2	NO2					-cont-

TSTEP	103	12	.001	3					+A
+A		9	0.01	1					

Field	*Contents*	*Data Range*	*Default*
SID	Set identification number. [1]	Integer>0	Required
Ni	Number of time steps.	Integer>0	Required
DTi	Time increment.	Real>0.0	Required
NOi	Output Increment. [2]	Integer>0	Required

Remarks:

1. The time step must be selected in the Case Control packet with the command:

   TSTEP = SID

2. Output will be generated at each **NOi** time step.

7 BULK

Bulk Data Entry USET

Assigns degrees of freedom to a user set.

Format and Example:

1	2	3	4	5	6	7	8	9	10
USET	SET	GID1	DOF1	GID2	DOF2	GID3	DOF3		

USET	U3	101	123	201	126				
USET	ZEROU3	123456	105	113					

Field	*Contents*	*Data Range*	*Default*
SET	User set name. [1]	Character { [ZERO]U1 [ZERO]U2 [ZERO]U3 [ZERO]U4 [ZERO]U5 [ZERO]U6 [ZERO]U7 [ZERO]U8 }	Required
GIDi	GRID or SCALAR point identification number.	Integer>0	Required
DOFi	List of degrees of freedom.	DOF Code	Required

Remarks:

1. If the **ZERO** keyword is appended to the set name, then the refereced degrees of freedom will be ***excluded*** from the set. In addition to the default names **U1-U8**, you may also specify a new user set name that you have defined with a **DEFUSET** Bulk Data entry.
2. The user set names are used in some DMAP modules such as **VEC**, **UPARTN**, and **UMERGE**.

7 BULK

Bulk Data Entry USET1

Assigns degrees of freedom to a user set.

Format and Examples:

1	2	3	4	5	6	7	8	9	10
USET1	SET	DOF	*GRID ID LIST*						-cont-

USET1	U3	123456	101	THRU	126	BY	2		
USET1	ZEROU3	123456	105	113					

Field	*Contents*	*Data Range*	*Default*
SET	User set name. [1]	Character { **[ZERO]U1** **[ZERO]U2** **[ZERO]U3** **[ZERO]U4** **[ZERO]U5** **[ZERO]U6** **[ZERO]U7** **[ZERO]U8** }	Required
DOF	List of degrees of freedom.	DOF Code	Required
GRID ID LIST	List of one or more GRID or SCALAR point identification numbers.	Integer>0	Required

Remarks:

1. If the **ZERO** keyword is prepended to the set name, then the refereced degrees of freedom will be ***excluded*** from the set. In addition to the default names **U1-U8**, you may also specify a new user set name that you have defined with a **DEFUSET** Bulk Data entry.
2. The user set names are used in some DMAP modules such as **VEC**, **UPARTN**, and **UMERGE**.

7 BULK

Bulk Data Entry VIEW

Defines shading and subelement mesh for radiation exchange calculations.

Format and Example:

1	2	3	4	5	6	7	8	9	10
VIEW	VID	SHDR	SHDE	NB	NG	DLIN			
VIEW	100	NO	YES	4	8	0.7			

Field	*Contents*	*Data Range*	*Default*
VID	**VIEW** identification number.	Integer>0	Required
SHDR	Indicates that the element can shade other elements.	Character { **YES** } { **NO** }	NO
SHDE	Indicates that the element can be shaded by other elements.	Character { **YES** } { **NO** }	NO
NB	Subelement mesh size in the B-direction [3].	Integer>0	**1**
NG	Subelement mesh size in the G-direction [3].	Integer>0	**1**
DLIN	Displacement of a surface perpendicular to the active side of the surface. [4]	Real	**0.0**

Remarks:

1. **VIEW** entries must be referenced by **CHBDY** entries in order to be used in the computation of radiation exchange coefficients.
2. The shading flags should be used with caution. Computer time may be saved if the user can identify a surface which cannot cause shading between any other surfaces by setting **SHDR=NO**. Similarly identifying surfaces which cannot be shaded by setting **SHDE=NO**, will also save computer time. If in doubt, all surfaces must be labeled **SHDR=YES** and **SHDE=YES**.
3. The use of data **NB** and **NG** is a function of the HBDY element type.
4. The **DLIN** field is used only for HBDY **LINE** elements.

7 BULK

Bulk Data Entry VIEWOP

Options to control the operations performed with the automatic calculation of radiation view factors

Format and Example:

1	2	3	4	5	6	7	8	9	10
VIEWOP	OUT	METHOD	RMAX	SHADE	SUM	PREC	TOL		
VIEWOP	PRINT	FDIFF	0.1	NO	YES	HIGH	3.0		

Field	*Contents*	*Data Range*	*Default*
OUT	Ouput option. [1]	Character { **PRINT** **PUNCH** **BOTH** **NONE** }	NONE
METHOD	View factor computation method. [2,3]	Character { **FDIFF** **CONT** }	[3]
RMAX	Threshold value to be used in automatically selecting a view factor computation method. [3]	Real>0.0	**0.1**
SHADE	Shading computation selector. [4]	Character { **YES** **NO** }	**YES**
SUM	Requests a summary of the shading conditions and subelement divisions for each HBDY element.	Character { **YES** **NO** }	NO
PREC	Specifies the precision of **RADMTX** Bulk Data output. [1,5]	Character { **STANDARD** **HIGH** }	STANDARD
TOL	Used in checking the planarity of HBDY AREA4 elements. The value specifies an allowable deviation from planarity as illustrated in the figure. [6]	Real >0.0	**1.0**

Remarks:

1. You may use the **ASSIGN** Executive Control command to define a logical file with **USE=BULK** if you use the **PUNCH** or **BOTH** options. **RADLST** and **RADMTX** Bulk Data will then be written to file BULK. Alternately, you may use the automatically assigned default file.
2. You may select the finite difference method, **FDIFF**, or the contour integration method, **CONT**.
3. If **METHOD** is blank view factors between two subelements, *i* and *j*, are computed based on the threshold **RMAX**. The contour integration method, **CONT**, is selected only if:

$$\frac{A_j}{d_{ij}^2} > \textbf{RMAX}$$

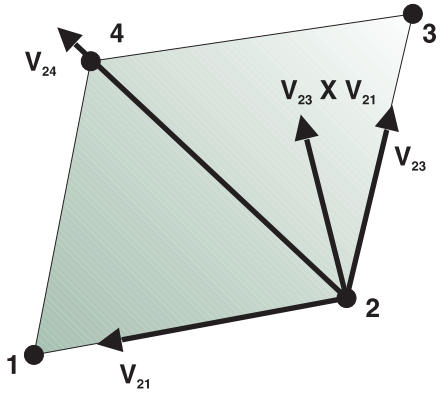
where A_j is the area of subelement *j*, and d_{ij} is the distance between subelement *i* and *j*. The default value of **RMAX** is **0.1**. If the above relationship is not satisfied the finite difference, **FDIFF**, method is selected.

7 BULK

4. If you select **YES**, then the shading specified on the **VIEW** Bulk Data entry for each HBDY element is used.

5. The **STANDARD** option produces standard precision Bulk Data entry images on file BULK, whereas the **HIGH** option produces high precision (16 characters per field) Bulk Data entries.

6. The value of **TOL** is used to check the degree of planarity of the HBDY element as described next. As illustrated in the figure below, vector $V_{23} \times V_{21}$ is defined as the normal to the HBDY element. Vector $\boldsymbol{V}_{24}$ measures the warpage in the element by passing through opposite GRID points as illustrated. The element is within the allowed tolarance if the angular difference between the normal and this vector is **TOL** degrees of 90 degrees.



7 BULK

Chapter 8

DIRECT MATRIX ABSTRACTION

Each of the solution algorithms, or Rigid Formats, available in **UAI/*NASTRAN*** is defined by a program written in the special language DMAP, the Direct Matrix Abstraction Program. It is the DMAP language which provides the great flexibility of the **UAI/*NASTRAN*** software system. You may use the ALTER feature, described in Chapter 2, to modify a Rigid Format, or you may perform an entire sequence of operations using your own DMAP program.

8.1 DMAP INSTRUCTIONS

There are two types of DMAP instructions. These are called ***Executive instructions*** and ***Functional module instructions***. The Functional modules are further grouped as:

- ❒ Structural Modules
- ❒ Matrix Modules
- ❒ Database Modules
- ❒ Utility Modules

Structural modules are those that relate to specific finite element operations, such as element matrix generation and assembly, and to engineering data manipulation, such as stress recovery and geometry processing. ***Matrix modules*** perform numerous efficient mathematical operations including equation solving, matrix algebra and other functions. ***Database modules*** allow you to import from, and export to an ***eBase*** database. ***Utility modules*** perform general purpose functions such as printing data and allowing external interfaces to **UAI*/NASTRAN***. The Executive instructions control the sequence of DMAP operations and aid in the data flow within the program. This chapter discusses the Executive Instructions and the Matrix and Utility modules which are summarized in Table 8-1.

Table 8-1. DMAP MODULES FOR GENERAL USE

MODULE CLASS	MODULE NAMES				
EXECUTIVE INSTRUCTIONS	BEGIN	CHKPNT	COND	END	EQUIV
	EXIT	FILE	JUMP	LABEL	PRECHK
	PURGE	REPT	SAVE		
MATRIX MODULES	ADD	ADD5	DECOMP	DIAGONAL	FBS
	MERGE	MPYAD	MPY3	PARTN	SMPYAD
	SOLVE	TRNSP	UMERGE	UMERGE1	UPARTN
DATABASE MODULES	DBIN	DBOUT	DBPARM	SOFI	SOFO
	SOFUT				
UTILITY MODULES	APPEND	COPY	INPUTT1	INPUTT2	INPUTT4
	LAMX	MATGEN	MATMOD	MATGPR	MATPCH
	MATPRN	MATPRT	OUTPUT1	OUTPUT2	OUTPUT3
	OUTPUT4	PARAM	PARAML	PARAMR	POST
	PRTPARM	SCALAR	SETVAL	SWITCH	TABPCH
	TABPRT	TABPT	TRAILER	VEC	VECPLOT

8 DMAP

8.2 DATA FLOW IN UAI/*NASTRAN*

The most important feature of DMAP is the ability to transfer data from one module to the next. This is accomplished by the **UAI/*NASTRAN*** Engineering Database Management System, ***eBase***. All data within **UAI/*NASTRAN*** are organized into ***Database Entities***. An entity contains either a table, such as a grid point identification number and its spatial coordinates, or a matrix stored in a special ***packed*** form for minimizing storage.

There are two classes of entities. Those that are created automatically from the input data stream, called ***Preface*** entities, and those created by DMAP modules. You, as a DMAP programmer, may give any name to an entity, but the preface entity names are reserved. This means that they must be called by the particular name that has been assigned to them. The Preface entities are shown in Table 8.2.

Every entity has a status. The status may be ***generated***, ***not generated*** or ***purged***. An entity is generated when it contains data that is available for input into a subsequent module. Not generated means that the entity does not exist but may be created as output from a Functional module. If an entity is purged, it is not available for generation and may not be used for input or output in a module. There are two Executive instructions which modify the entity status: **PURGE** and **EQUIV**.

Table 8-2. PREFACE eBase ENTITY NAMES

ENTITY NAME	DATA CONTENTS
ALTER	DMAP Alter data.
AXIC	Axisymmetric modeling data.
BDAT	Internal Bulk Data.
BULKDATA	Sorted, expanded Bulk Data file.
CASECC	Case Control data.
DIT	Direct input tables.
DMAP	Rigid Format data.
DYNAMICS	Dynamic analysis data.
EDT	Element deformation data.
EPT	Element property data.
GEOM1	Grid and coordinate data.
GEOM2	Element connection data.
GEOM3	Load data.
GEOM4	Constraint data.

ENTITY NAME	DATA CONTENTS
GEOM5	Design sensitivity data.
IFPFILE	Parameter data.
MATPOOL	Direct input matrix data.
MED	Restart data.
MPT	Material property data.
OPTDICT	Checkpoint data.
OSCAR	Compiled DMAP data.
PCDB	Structure plot control data.
SOLUTION	Executive Control data.
STEPCC	Nonlinear step control.
XYCDB	X-Y plot control data.
IMPCTRL	IMPORT control data.
ARCHPRM	ARCHIVE control data.
ARCHCHK	ARCHIVE control data.

8.3 DMAP INSTRUCTION SYNTAX

The syntax of the Functional module instructions and Executive instructions is different. These differences are described in this section.

8.3.1 Syntax of Functional Module Instructions

The general syntax of a Functional module instruction is:

```
MODNAME ilist / olist / parmlist $
```

where:

- MODNAME is the Functional module name
- *ilist* is a list of input entity names
- *olist* is a list of output entity names
- *parmlist* is a list of parameters

Note that each section of the instruction is separated by a slash (**/**). The dollar sign (**$**) denotes the end of an instruction. It is not required unless the statement ends with a slash (**/**).

Input and Output Entity Lists

The input and output entity name lists have the same form:

```
name1,name2,name3,...,namen
```

The number of input and output entities required by a Functional module is predefined, and the correct number must be entered for each. An entity name may be omitted if it is not needed for a particular application. All entity references beyond the last needed entity may be omitted.

The Parameter Specification List

Parameters may serve many purposes in a DMAP program. They may pass data values into, or out of, a module, or they may be used as flags to control the computational flow within the module or the DMAP program. There are two allowable forms of the parameter section of the DMAP instruction. The first explicitly states the attributes of the parameters, while the second is a simplified specification. The general form of the formal parameter section is:

```
/ type , mode , parm /
```

where the allowable parameter specifications are:

type	V	Parameter value is variable and may be changed by a module during execution.
	C	Value is constant and may not be changed.
	S	Parameter is of type **V** and it will be saved at the completion of the current module.
mode	Y	Initial parameter value may be specified using a **PARAM** Bulk Data entry.
	N	Initial value may not be specified.
parm	PNAME	PNAME is a character name selected to represent a given parameter symbolically.
	PNAME=***value***	Sets the value of the symbolic parameter PNAME.
	value	Is the actual value for an unnamed parameter.

The *type* operand defines whether the parameter is a variable, **V**, or a constant, **C**. A special type, **S,** denotes a variable parameter that is automatically saved. (See the description of the SAVE instruction.) The *mode* operand defines whether a parameter may, **Y**, or may not, **N**, be changed by entering a **PARAM** Bulk Data entry into the Bulk Data packet. *parm* specifies the parameter name, and optionally, its value.

Each parameter has a specific arithmetic type. These types, with examples of each, are shown in the table below:

NUMERIC TYPE	EXAMPLES		
INTEGER	7	-2	0
REAL	-3.6	2.13+5	-3.1-4
CHARACTER	VAR01	STRING	B3R56
REAL DOUBLE PRECISION	2.5D1	-.32D-3	.4D03
COMPLEX	(-1.0,-3.5)	(-1.+3,2.1-3)	
COMPLEX DOUBLE PRECISION	(1.23D-4,3.061D-2)		

The arithmetic type of each module parameter is defined within **UAI/*NASTRAN***. When using DMAP, the type that is input must match this definition.

The simplified parameter specifications may be used in certain frequently occurring instances. When you are specifying a constant value, then you may omit the *type* and *mode*:

> */C,N,value/* may be written as */value/*

When the constant is a character string, then the special syntax:

```
/'string'/
```

is used. Similarly, when you enter a parameter name of *type* **V** and *mode* **N**, it is only necessary to enter the name. This is also true if you are initializing the paramter value at the same time:

```
/V,N,PNAME/  may be written as  /PNAME/
    and
/V,N,PNAME=value/  may be written as  /PNAME=value/
```

Finally, you may select the predefined default value for a parameter by entering two successive slashes:

```
/(default_value)/  may be written as  //
```

Note that if you use this form and the parameter does not have a default value that you will receive an error message.

8.3.2 Syntax of Executive Instructions

Each Executive instruction has its own format which is generally open-ended, meaning that the number of module arguments is not prescribed. Executive instructions are divided into two categories:

- **Control instructions: BEGIN**, **COND**, **END**, **EXIT**, **JUMP**, **LABEL** and **REPT** which control the order in which DMAP instructions are executed.

- **File instructions: CHKPNT**, **EQUIV**, **FILE**, **PRECHK**, **PURGE**, and **SAVE** which aid the **UAI*/NASTRAN*** Executive System in allocating files, interfacing between Functional modules, and in Restarting a problem.

The rules associated with the Executive instructions are distinct for each instruction and are discussed individually in Section 8.5.

8.4 EXAMPLES OF DMAP

This section contains several examples of DMAP programs. They illustrate many of the features used for creating useful utility programs, particularly for matrix operations.

8.4.1 Solving Matrix Equations

Assume that the constrained symmetric matrix $\boldsymbol{K}_{ll}$ and the load vector $\boldsymbol{P}_l$ are defined with **DMI** Bulk Data. It is desired to perform the following matrix operations:

$$\boldsymbol{U}_l = \boldsymbol{K}_{ll}^{-1}\,\boldsymbol{P}_l$$

$$\boldsymbol{R} = \boldsymbol{K}_{ll}\,\boldsymbol{U}_l - \boldsymbol{P}_l$$

$$\partial\boldsymbol{U} = \boldsymbol{K}_{ll}^{-1}\,\boldsymbol{R}$$

$$\boldsymbol{U} = \boldsymbol{U}_l + \partial\boldsymbol{U}$$

and then print matrix $\boldsymbol{U}$. A typical DMAP program to do this is:

```
BEGIN $
SOLVE  KLL,PL/UL/1/1 $
MPYAD  KLL,UL,PL/R/0/1/-1 $
SOLVE  KLL,R/DU/1 $
ADD    UL,DU/U $
MATPRN U// $
END    $
```

This DMAP uses the **SOLVE** module twice. This means that $\boldsymbol{K}_{ll}$ must be decomposed twice, a potentially costly operation. An alternate DMAP using the symmetric decomposition module, **DECOMP**, eliminates this problem:

```
BEGIN $
DECOMP KLL/LLL,ULL $
FBS    LLL,ULL,PL/UL/1/1 $
MPYAD  KLL,UL,PL/R/0/1/-1 $
FBS    LLL,ULL,R/DU $
ADD    UL,DU/U $
MATPRN U// $
END    $
```

8.4.2 Looping in DMAP Programs

Suppose that given a square matrix $\boldsymbol{Q}$, it is desired to compute $\boldsymbol{Q}^r$. It is assumed that the value of the integer power, $r>1$, is input using **PARAM** Bulk Data and that $\boldsymbol{Q}$ is entered using **DMI** Bulk Data. One possible DMAP to accomplish this is:

```
BEGIN $
MATPRN Q// $
PARAM  //'SUB'/RR/V,Y,R=2/1 $
COPY   Q/P $
LABEL  TOP
MPYAD  Q,P//PP/0 $
SWITCH P,PP// $
REPT   TOP,RR $
MATPRN PP// $
END    $
```

The DMAP loop has been identified in the program. The loop is executed **RR** times and a matrix multiply and **SWITCH** are performed each time.

8.4.3 Partitioning Operations and ALTERs

One of the functions of the structural module SMP1 is to partition the *f-set* stiffnes matrix into the *a-set* and *o-set* for Guyan Reduction:

$$\boldsymbol{K}_{ff} = \begin{bmatrix} \bar{\boldsymbol{K}}_{aa} & \boldsymbol{K}_{ao} \\ \boldsymbol{K}_{oa} & \boldsymbol{K}_{oo} \end{bmatrix}$$

$$\boldsymbol{G}_o = -\boldsymbol{K}_{oo}^{-1}\,\boldsymbol{K}_{oa}$$

$$\boldsymbol{K}_{aa} = \bar{\boldsymbol{K}}_{aa} + \boldsymbol{K}_{oa}^{T}\,\boldsymbol{G}_o$$

These operations may be performed directly in DMAP without using **SMP1**. To do this, Rigid Format 1 is ALTERed (see Chapter 9) and **SMP1** is replaced by DMAP:

```
ALTER  85,85
   UPARTN USET,KFF/KOO,,KOA,KAAB/'F''O''A' $
   CHKPNT KOO,KOA,KAAB $
   SOLVE  KOO,KOA/GO/1/-1 $
   CHKPNT GO $
   MPYAD  KOA,GO,KAAB/KAA/1 $
   CHKPNT KAA $
ENDALTER
```

This ALTER subpacket is placed in the Executive Control packet. The **ALTER 85,85** command removes the old **SMP1** module and replaces it with the DMAP that follows. **ENDALTER** terminates the subpacket. A reason to perform such an operation is to allow the Checkpointing of data blocks more often than is possible if **SMP1** is used.

8.4.4 Testing and Branching with DMAP

Suppose that the following matrix operations are to be performed:

$$\boldsymbol{X} = \begin{Bmatrix} \boldsymbol{AB}+\boldsymbol{C} & \text{if } k<0 \\ (p\boldsymbol{A}+\boldsymbol{B})^T & \text{if } k=0 \\ \boldsymbol{A}^2\boldsymbol{C}^{-1} & \text{if } k>0 \end{Bmatrix}$$

It is again assumed that the matrices ***A, B*** and ***C*** have been defined using **DMI** Bulk Data, and that the real constant, *p*, and the integer constant, *k*, have been specified with **PARAM** Bulk Data. The input and output results are then printed. The following DMAP will accomplish this:

```
BEGIN $
PARAM  //'NOP'/V,N,TRUE=-1 $
MATPRN A,B,C// $
COND   ONE,K $
PARAM  //'NOT'/V,N,CHOOSE/V,Y,K $
COND   THREE,CHOOSE $
JUMP   TWO $
LABEL  ONE $
MPYAD  A,B,C/X/0 $
JUMP   ALLDONE $
LABEL  TWO $
ADD    A,B/Y/C,Y,P=(0.0,0.0) $
TRNSP  Y/X2 $
EQUIV  X2,X/TRUE $
JUMP   ALLDONE $
LABEL  THREE $
SOLVE  C/Z $
MPYAD  A,Z/W/0 $
MPYAD  A,W/X3/0 $
EQUIV  X3,X/TRUE $
LABEL  ALLDONE $
MATPRN X// $
END $
```

Here, the **COND** and **JUMP** Executive instructions have been used to create separate DMAP to perform each of the operations depending upon the value of *k*.

8.5 DMAP MODULE DESCRIPTIONS

The remainder of this chapter provides you with the detailed description of the Executive instructions and Matrix and Utility modules that are available for use. The modules are listed alphabetically. For each module, all arguments are shown and described by six items as summarized below:

Name

Provides the symbolic name of the module argument as given in the DMAP command syntax.

Mode

Indicates whether the NAME represents INPUT data or is an OUTPUT result of the module.

Argument Type

Specifies the type of NAME. These types include MATRIX for matrix database entities, TABLE for table entities, ANY ENTITY if either type of entity may be used, and PARAM for parameters. Occasionally a module may require a matrix entity with a specified topology. These are indicated by terms such as SQUARE MATRIX. A LABEL is a special character string which defines a DMAP language label statement, much like a labeled FORTRAN statement. Refer to the descriptions of the DMAP statements COND and LABEL in Section 8 .5.

Numeric Type

Specifies the type of numeric data that a matrix or parameter may be. Codes include INT, for integer, RSP and RDP for real, single and real, double precision, CSP and CDP for complex, single and complex, double precision. The specifier ANY indicates that any numeric type is valid, and N/A indicates that a numeric type is not applicable to the NAME. CHARACTER means that a general character string may be input, from 1 to 8 characters in length.

Description

Defines the meaning of NAME and descriptions of any options that may be available.

Default

Specifies what value, if any, NAME will have if you omit it from the DMAP instruction.

Module: ADD

Performs the matrix addition:

$$X = \alpha A \otimes \beta B$$

where α and β are scalar multipliers, and where $\otimes$ can be any of the operators: +, *, ÷ or "overwrite"

DMAP Instruction Syntax:

```
ADD   A,B / X / ALPHA / BETA / OPT / 'TYPE' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A, B	INPUT	MATRIX	ANY	Distinct addend matrix entity names. [1]	Required
X	OUTPUT	MATRIX	[2]	Resultant matrix entity name. [3]	Required
ALPHA	INPUT	PARAM	CSP	Scalar coefficient of **A**.	(1.0,0.0)
BETA	INPUT	PARAM	CSP	Scalar coefficient of **B**.	(1.0,0.0)
OPT	INPUT	PARAM	INT	Operator option: 0: Term by term addition 1: Term by term multiplication 2: Term by term division (A_{ij} / B_{ij}) 3: $X_{ij} = \alpha A_{ij}$ if $\lvert B_{ij} \rvert = 0$ $X_{ij} = \beta B_{ij}$ if $\lvert B_{ij} \rvert \neq 0$	0
TYPE	INPUT	PARAM	CHAR	Type of output matrix: **REAL** - Output is a real matrix **COMPLEX** - Output is a complex matrix Blank - Type of output matrix is the maximum of the data types for **A**, **B**, **ALPHA**, and **BETA**.	Blank

Remarks:

1. **A** or **B** may be purged. If they are both purged, the matrix sum will be assumed null.
2. The type and precision of **X** is the maximum of the types of **A**, **B**, α, β unless overridden by the **TYPE** parameter. It is illegal to request a **REAL** output matrix if any of the input matrices are complex or if any of the input coefficients have nonzero imaginary parts.
3. **X** cannot be purged.
4. The size of **X** is the size of **A** if **A** is present. Otherwise it is that of **B**.
5. All operators, other than the default add operator, require the presence of both **A** and **B**.

8 DMAP

Example:

1. Perform the operation

   $\overline{K_{gg}} = KX_{gg} + 3.0 * KY_{gg}$

   using the **ADD** module:

```
ADD      KXGG,KYGG / KGGBAR / / (3.0,0.0) $
```

2. Convert real matrix REAL to complex matrix COMPLEX using a real, scalar parameter FACTOR:

```
ADD      REAL, / COMPLEX / FACTOR / / 0 / 'COMPLEX' $
```

Module: ADD5

Performs the matrix addition:

$$\boldsymbol{X} = \alpha_1\boldsymbol{A}_1 + \alpha_2\boldsymbol{A}_2 + \alpha_3\boldsymbol{A}_3 + \alpha_4\boldsymbol{A}_4 + \alpha_5\boldsymbol{A}_5$$

where α_i are scalar multipliers.

DMAP Instruction Syntax:

```
ADD5  A1,A2,A3,A4,A5 / X / ALPHA1 / ALPHA2 / ALPHA3 /
      ALPHA4 / ALPHA5 / 'TYPE' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
Ai	INPUT	MATRIX	ANY	Distinct addend matrix entity names. [1]	At least one required
X	OUTPUT	MATRIX	[2]	Resultant matrix entity name. [3]	Required
ALPHAi	INPUT	PARAM	CSP	Scalar coefficient of **Ai**.	(1.0,0.0)
TYPE	INPUT	PARAM	CHAR	Type of output matrix: **REAL** - Output is a real matrix **COMPLEX** - Output is a complex matrix Blank - Type of output matrix is the maximum of the data types for **Ai**, and **ALPHAi**.	Blank

Remarks:

1. Any of the input matrices may be purged. If they are all purged, the matrix sum will be assumed null.
2. The type and precision of **X** is the maximum of the types of **Ai** and **ALPHAi** unless overridden by the **TYPE** parameter. It is illegal to request a **REAL** output matrix if any of the input matrices are complex or if any of the input coefficients have nonzero imaginary parts.
3. **X** cannot be purged.
4. The size of **X** is the size of the first nonpurged input matrix.

Example:

1. Perform the operation

   $$\boldsymbol{K} = \boldsymbol{K}_1 + 2.0 * \boldsymbol{K}_2 + (4.5,1.0) * \boldsymbol{K}_3$$

   using the **ADD5** module:

```
ADD5     K1,K2,K3 / K / / (2.0,0.0) / (4.5,1.0) $
```

8 DMAP

Module: APPEND

Appends data blocks to one another.

DMAP Instruction Syntax:

```
APPEND      IN1,IN2,IN3,IN4,IN5 / OUT1,OUT2,OUT3,OUT4,OUT5 /
            APPFLG / NZREC / TYPEB / PREC / ROWS  $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
INi	INPUT	ANY ENTITY	ANY	Input entity to be copied or appended to output entity.	At least one required
OUi	OUTPUT	ANY ENTITY	[2]	Resultant entity name.	Required to match input
APPFLG	INPUT	PARAM	INT	If **+1**: Output entities are copies of input entities. If not **+1**: Input entities are appended to output entities.	Required
NZREC	INPUT	PARAM	INT	If >0 and the input entity is purged, **NZREC** is the number of null records to be appended to the corresponding output entity.	0
TYPEB	INPUT	PARAM	CHAR	Must be either **'MATRIX'** or **'TABLE'**. If the input entity is purged, and the corresponding output entity is empty, the output entity will be created with this type.	Required
PREC	INPUT	PARAM	CHAR	Must be one of **'PREC'**, **'SINGLE'** or **'DOUBLE'**. If the input entity is purged, and the corresponding output entity is null, and **TYPEB** is **'MATRIX'**, the output matrix will be of this precision.	Required
ROWS	INPUT	PARAM	INT	If the input entity is purged and the output entity is of type **'MATRIX'**, the output matrix will have this number of rows.	Required

Executive Instruction: BEGIN

Indicates the beginning of a DMAP program.

DMAP Instruction Syntax:

```
BEGIN [ { GO   } ] [ , ERR = level ] [ , { NOLIST } ] [ , { NODECK } ]
      [ { NOGO } ]                     [ { LIST   } ]   [ { DECK   } ]

                                  [ , { NOREF  } ] [ , { NOWARN } ] $
                                  [ { REF   } ]    [ { WARN  } ]
```

Option	*Type*	*Description*	*Default*
{ GO / NOGO }	CHAR	Requests DMAP program compilation and execution, or termination of **UAI/*NASTRAN*** after compilation.	**GO**
ERR=*level*	INT	Defines the error *level* at which suspension of execution will occur. *level=0* selects the warning error level, *level=1* the potentially fatal error level, and a *level=2* the fatal error level. [1]	**2**
{ LIST / NOLIST }	CHAR	Selects or deselects DMAP program listing option.	**NOLIST**
{ DECK / NODECK }	CHAR	Selects or deselects writing the DMAP program to the PUNCH file.	**NODECK**
{ REF / NOREF }	CHAR	Selects or deselects creating a DMAP cross-reference listing.	**NOREF**
{ WARN / NOWARN }	CHAR	Selects or deselects warning message print. If **WARN** is selected, then warning and potentially fatal error messages are printed, otherwise they are not.	**NOWARN**

Remarks:

1. **BEGIN** is a non-executable DMAP instruction which is used only by the DMAP compiler for information purposes.
2. A **BEGIN** instruction is required when selecting **APP DMAP** in the Executive Control packet. This is followed by DMAP instructions up to and including the **END** statement.
3. The use of **BEGIN** implicitly selects all compiler defaults.

8 DMAP

Executive Instruction: CHKPNT

Requests that database entities be written on the Checkpoint data file to enable the problem to be Restarted with a minimum of redundant processing.

DMAP Instruction Syntax:

```
CHKPNT      D1,D2,...,DN $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
Di	INPUT	ANY ENTITY	N/A	Database entity names to be written to the Checkpoint data file. [1]	At least one required

Remarks:

1. An entity to be Checkpointed must have been referenced in a previous PURGE, EQUIV or functional module instruction.
2. **CHKPNT** cannot be the first instruction of a DMAP loop.
3. Database entities generated by the Input File Processor (including **DMI**s and **DTI**s) should not be Checkpointed since they are always regenerated on Restart.
4. Checkpointing only takes place when a new Checkpoint data file is **ASSIGN**ed as shown below and the Executive Control command:

   ASSIGN NPTP=*phys_name*,**SAVE,USE=CHKPNT**
   CHKPNT YES

   appears in the Executive Control packet. Otherwise, the **CHKPNT** instructions are ignored.
5. For each entity that is successfully Checkpointed, an entry is made in the Checkpoint dictionary file.
6. For entities that have been purged or equivalenced, an entry is made in the Checkpoint dictionary file to this effect. In these cases entities are not written to the Checkpoint file.
7. You may use the **PRECHK** instruction for an automated **CHKPNT** capability.

Executive Instruction: COND

Alters the normal order of execution of DMAP modules by conditionally transferring program control to a specified location in the DMAP program.

DMAP Instruction Syntax:

```
COND  LABEL,ENABLE $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
LABEL	N/A	LABEL	CHAR	Name of a label to which control will be transferred.	Required
ENABLE	N/A	PARAM	INT	Parameter enabling the transfer. If **ENABLE** is negative, then the transfer occurs, otherwise control passes to the next DMAP statement.	Required

Remarks:

1. Only forward transfers are allowed. See the **REPT** instruction for backward transfers.
2. See the **LABEL** instruction for defining location labels.

Example:

1. Jump to the label named **LAB001** based on the value of parameter **IFGOOD**:

```
COND      LABOO1,IFGOOD $
```

8 DMAP

Module: COPY

Generates a physical copy of a database entity.

DMAP Instruction Syntax:

```
COPY  DBIN / DBOUT / ENABLE $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBIN	INPUT	ANY ENTITY	N/A	Name of the entity to be copied. [1]	Required
DBOUT	OUTPUT	ANY ENTITY	N/A	The name of the new copy of **DBIN**.	Required
ENABLE	INPUT	PARAM	INT	Parameter enabling the **COPY**. If **ENABLE** is negative, then the copy is performed, otherwise the instruction is ignored.	**-1**

Remarks:

1. The input entity, **DBIN**, may not be purged.

Example:

1. Copy the contents of entity **MYDATA** to entity **YOURDATA** based on the value of parameter **DOCOPY**:

```
COPY   MYDATA / YOURDATA / DOCOPY $
```

Module: DBIN

Reads, or imports, an ***eBase*** entity from an ARCHIVE database into a DMAP entity.

DMAP Instruction Syntax:

```
DBIN  / DB / 'LUN' / 'NAME' / INDEX1 / INDEX2 / INDEX3 / 'DNAME1' /
              'DNAME2' / DNAME3' / 'DNAME4 ' / 'DNAME5' /  $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DB	INPUT	ANY ENTITY	CHAR	Name of the data block entity to be imported. [1]	Required
LUN	INPUT	PARAM	CHAR	The name of the ARCHIVE database containing **DB**. [2]	**ARCHIVE**
NAME	INPUT	PARAM	CHAR	Base name of the of **DB** on the ARCHIVE database . [3]	**ENTITY**
INDEXi	INPUT	PARAM	INT	Subscript values to be applied to **DB** on the ARCHIVE database . [3,4]	
DNAMEi	INPUT	PARAM	CHAR	Names of up to five directory levels defining the location of **DB**. [3,4]	None

Remarks:

1. The input entity, **DB**, may not be purged.
2. Your Executive Control packet must **ASSIGN** a database with **USE=ARCHIVE** to import an entity.
3. The name of entity **DB** on the ARCHIVE database is constructed from a combination of the base name, subscripts, and subdirectories. The general form of such a name is:

   /DNAME1/DNAME2/DNAME3/DNAME4/DNAME5/DB(INDEX1,INDEX2,INDEX3)

4. None of the intervening subscripts or directory names may be undefined or blank. In other words, an entity may not have subscripts **(0,0,3)** or a directory path such as **/one//three//five**.
5. Note that **DBIN** and **DBOUT** may also be used to import and export Substructuring data from an SOF database. See Chapter 19 of the ***User's Guide*** for additional information.

8 DMAP

New: V20.1

Example:

1. Import the matrix **MYMAT/STIFFNESS/KGG(1,1)** which is stored in database **mydata**, into the entity **KGG** after line 147 of the DMAP Solution Sequence:

   Executive Control:

```
ASSIGN MYDATA,NEW,USE=ARCHIVE
```

   DMAP ALTER:

```
ALTER 147
DBIN     /KGG//'MYDATA'/'KGG'/1/1//'MYMAT'/'STIFFNESS' $
CEND
```


New: V20.1

Module: DBOUT

Writes, or exports, a DMAP data block entity from a DMAP Solution Sequence to an ARCHIVE database.

DMAP Instruction Syntax:

```
DBOUT    DB // 'LUN' / 'NAME' / INDEX1 / INDEX2 / INDEX3 / 'DNAME1' /
              'DNAME2' / DNAME3' / 'DNAME4 ' / 'DNAME5' /  $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DB	INPUT	ANY ENTITY	CHAR	Name of the data block entity to be exported. [1]	Required
LUN	INPUT	PARAM	CHAR	The name of the ARCHIVE database which will contain **DB**. [2]	**ARCHIVE**
NAME	INPUT	PARAM	CHAR	Base name of the of **DB** on the ARCHIVE database . [3]	**ENTITY**
INDEXi	INPUT	PARAM	INT	Subscript values to be applied to **DB** on the ARCHIVE database . [3,4]	
DNAMEi	INPUT	PARAM	CHAR	Names of up to five directory levels defining the location of **DB**. [3,4]	None

Remarks:

1. The input DMAP data block entity, **DB**, may not be purged.
2. Your Executive Control packet must **ASSIGN** a database with **USE=ARCHIVE** to export a DMAP data block entity.
3. The name of **DB** on the ARCHIVE database is constructed from a combination of the base name, subscripts, and subdirectories. The general form of such a name is:

   /DNAME1/DNAME2/DNAME3/DNAME4/DNAME5/DB(INDEX1,INDEX2,INDEX3)

4. None of the intervening subscripts or directory names may be undefined or blank. In other words, an entity may not have subscripts **(0,0,3)** or a directory path such as **/one//three//five**.
5. Note that **DBIN** and **DBOUT** may also be used to import and export Substructuring data from an SOF database. See Chapter 19 of the ***User's Guide*** for additional information.

8 DMAP

New: V20.1

Example:

1. Export the matrix **KGG** after line 147 of the DMAP Solution Sequence to the database **mydata** giving it the name **MYMAT/STIFFNESS/MYKGG(1,1)**:

   Executive Control:

   ```
   ASSIGN MYDATA,OLD,USE=ARCHIVE
   ```

   DMAP ALTER:

   ```
   ALTER 147
   DBOUT    KGG//'MYDATA''MYKGG'/1/1//'MYMAT''STIFFNESS' $
   CEND
   ```


New: V20.1

Module: DBPARM

Saves, or restores, DMAP Solution Sequence **PARAM**eters to an ARCHIVE database.

DMAP Instruction Syntax:

```
DBPARM      // 'OPTION' / 'LUN' / 'NAME1' / 'NAME2' / 'NAME3' /
               'NAME4' / 'NAME5' / 'NAMEP' / INDEX1 / INDEX2 /
               INDEX3 / 'DNAME1' / 'DNAME2' / 'DNAME3' / 'DNAME4' /
               'DNAME5 $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
OPTION	INPUT	PARAM	CHAR	Option to be performed: **SAVE**: To save parameters **RESTORE**: To restore parameters	Required
LUN	INPUT	PARAM	CHAR	The name of the ARCHIVE database which will contain the parameter entity **NAMEP**. [1]	**ARCHIVE**
NAMEi	INPUT	PARAM	CHAR	Names of parameters to be saved on the ARCHIVE database . [2]	[2]
NAMEP	INPUT	PARAM	CHAR	The name of the entity which will contain the parameter values. [3]	Required
INDEXi	INPUT	PARAM	INT	Subscript values to be applied to **NAMEP** on the ARCHIVE database . [3,4]	None
DNAMEi	INPUT	PARAM	CHAR	Names of up to five directory levels defining the location of **NAMEP**. [3,4]	None

Remarks:

1. Your Executive Control packet must **ASSIGN** a database with **USE=ARCHIVE** to save or restore parameters.
2. If all of the **NAMEi** are blank, and **OPTION** is **SAVE**, then **ALL** of the DMAP parameters will be saved to, or restored from, the ARCHIVE database. This is **NOT** the case if **OPTION** is **RESTORE**. In this case, you must explicitly name the parameters that will be restored.
3. The name of **NAMEP** on the ARCHIVE database is constructed from a combination of the base name, subscripts, and subdirectories. The general form of such a name is:

   /DNAME1/DNAME2/DNAME3/DNAME4/DNAME5/NAMEP(INDEX1,INDEX2,INDEX3)

4. None of the intervening subscripts or directory names may be undefined or blank. In other words, an entity may not have subscripts **(0,0,3)** or a directory path such as **/one//three//five**.

8 DMAP

New: V20.1

Example:

1. Save parameters **P1**, **P2** and **P3** after line 147 of the DMAP Solution Sequence to the database **MYDATA** giving it the name **MYPARM/PARMS(1,1)**:

   Executive Control:

   ```
   ASSIGN MYDATA,NEW,USE=ARCHIVE
   ```

   DMAP ALTER:

   ```
   ALTER 147
   DBPARM   //'SAVE''MYDATA'/P1/P2/P3///'PARMS'/1/1//'MYPARM' $
   CEND
   ```

2. Restore the parameters **P1** and **P3**, saved in the previous example, after line 147 of the DMAP Solution Sequence.

   Executive Control:

   ```
   ASSIGN MYDATA,OLD,USE=ARCHIVE
   ```

   DMAP ALTER:

   ```
   ALTER 147
   DBPARM   //'RESTORE''MYDATA'/P1/P3////'PARMS'/1/1//'MYPARM' $
   CEND
   ```


New: V20.1

Module: DECOMP

Decomposes a square matrix **A** into upper and lower triangular factors **U** and **L**.

$$A \Rightarrow L\,U$$

DMAP Instruction Syntax:

```
DECOMP      A,USET,EQEXIN / L,U / KSYM / CHLSKY /
            MINDIAG / DET / POWER / SING / 'SET' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	SQUARE MATRIX	ANY	Name of the matrix entity to be decomposed.	Required
USET	INPUT	TABLE	N/A	Name of the displacement set definition table.	[1]
EQEXIN	INPUT	TABLE	N/A	Name of the external to internal GRID point equivalence table.	[1]
L,U	OUTPUT	MATRIX	ANY	Lower and upper triangular factors of **A**.. [2]	Required
KSYM	INPUT	PARAM	INT	Symmetry option: **-1**: Automatically determine type of decomposition **0**: Perform unsymmetric decomposition **+1**: Perform symmetric decomposition	**-1**
CHLSKY	INPUT	PARAM	INT	Cholesky decomposition selector [3]: **0**: Do not use Cholesky method **1**: Use Cholesky method	**0**
MINDIAG	OUTPUT	PARAM	CSP	The minimum diagonal term of **U**.	Required
DET	OUTPUT	PARAM	CSP	Scaled value of the determinant of **A**.	Required
POWER	OUTPUT	PARAM	INT	Integer power of 10 by which **DET** must be multiplied to get the actual value of the determinant of **A**.	Required
SING	OUTPUT	PARAM	INT	Singularity flag: **0**: **A** is not singular **1**: **A** is singular	Required
SET	INPUT	PARAM	CHAR	The name of the displacement set corresponding to the matrix **A**. [4]	Required

Remarks:

1. **USET** and **EQEXIN** are optional and are used when singularities are found in the matrix being decomposed. The tables allow the module to translate the singular column numbers into more easily understood values referencing GRID points and degrees of freedom.
2. Non-standard triangular factor matrix entities are used to improve the efficiency of the back substitution process in module **FBS**. The matrix utility modules should be cautiously employed when dealing with non-standard matrix entities.

3. The Cholesky method requires that **A** be positive definite. If this option is selected, the resulting factor, which will be written as **U**, cannot be input to **FBS**.
4. The **SET** parameter is required if **USET** and **EQEXIN** are input. The legal values of this parameter are:
   A, D, E, F, G, H, L, M, N, O, P, R, and **S.**
5. Variable parameters output from functional modules must be **SAVE**d if they are to be subsequently used. See the **SAVE** instruction.

Examples:

1. Decompose matrix **A** into its upper and lower triangular factors:

```
DECOMP   A,, / L,U / / MINDIAG / DET / POWER / SING $
```

2. Decompose the matrix **KGG** which is a structural matrix of g-size. Input the appropriate tables to allow the module to report matrix singularities using GRID point identification data:

```
DECOMP   KGG,USET,EQEXIN / L,U / / MINDIAG / DET / POWER /
         SING / 'G' $
```

Module: DIAGONAL

Extracts the real part of the diagonal of a matrix and stores the terms as a column vector, square symmetric matrix, or diagonal matrix. These terms may be raised to a specified power. Optionally, the output vector or matrix can contain all terms, not just the diagonals, of the input matrix raised to a power.

DMAP Instruction Syntax:

```
DIAGONAL    MATIN / MATOUT / OPTION / POWER $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
MATIN	INPUT	MATRIX	ANY	Name of a square or diagonal matrix entity to be copied. [1]	Required
MATOUT	OUTPUT	MATRIX	ANY	The name of the resulting matrix entity whose contents are based on **OPTION**.	Required
OPTION	INPUT	PARAM	CHAR	The extraction option: **COLUMN**: Produces a column vector **DIAGONAL**: Produces a diagonal matrix **SQUARE**: Produces a square matrix **WHOLE**: Produces an output matrix containing every term of **A** raised to the requested power	**COLUMN**
POWER	INPUT	PARAM	RSP	Exponent to which the real part of each element is raised. A value of **0.0** will cause the output matrix to contain values of **1.0**. If **POWER** is not 1.0 or 2.0, the desired terms of **A** must be non-negative or an error will occur.	**1.0**

Remarks:

1. The precision of the output matrix matches the precision of the input matrix. If **OPTION** is **WHOLE** and **POWER** is **1.0**, the absolute value of each term of **MATIN** will be returned.
2. Modules **DIAGONAL** and **MPYAD** will correctly process matrices created with the **DIAGONAL** option.

Examples:

1. Extract the diagonal of matrix **A** and store it in a column vector called **ADIAG**:

```
DIAGONAL    A / ADIAG / 'COLUMN' $
```

2. Raise all of the terms of matrix **B** to the power 2.5 and save as **B2P5**:

```
DIAGONAL    B / B2P5 / 'WHOLE' / 2.5 $
```

Executive Instruction: END

Indicates the end of a user-supplied DMAP program.

DMAP Instruction Syntax:

```
END $
```

Remarks:

1. The **END** instruction also acts as an implied **EXIT** instruction.
2. The **END** instruction is required whenever **APP DMAP** is selected.

Executive Instruction: EQUIV

Assigns one or more equivalent entity names, or alias, to an existing entity so that the entity may be referenced by several equivalent names.

DMAP Instruction Syntax:

```
EQUIV       PDB1,list1 / ENABLE1 / PDB2,list2 / ENABLE2 ...$
            listi ⇒ SDB1,SDB2,...
```

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
PDBi	INPUT	ANY ENTITY	N/A	Primary entity name [1,2]	Required
SDBi	INPUT	ANY ENTITY	N/A	Secondary entity name [1]	One required
ENABLEi	INPUT	PARAM	INT	Parameters enabling the operation: [3] <**0**: Perform the equivalence ≥**0**: Break the eqivalence	**-1**

Remarks:

1. Each entity name must appear within the DMAP sequence. The first entity name in each group is known as the ***primary entity*** and the second and subsequent entity names become equivalent to the primary (depending on the associated parameter value). These equivalenced entities are known as ***secondary entities***. The number of entity names **SDBi** prior to each parameter **ENABLEi** and the number of such groups in a particular instruction may be variable.
2. An **EQUIV** instruction may appear at any time as long as the primary entity name has been previously defined.
3. The entity names in each group are made equivalent if the value of the associated parameter is negative. If a number of entities are already equivalenced and the parameter value is non-negative, the equivalence is broken and the entity names again become unique. If the entities are not equivalenced and the parameter value is non-negative, no action is taken.
4. If an equivalence is to be performed at all times it is not necessary to specify a parameter name. For example:

```
EQUIV    DB1,DB2 // DB3,DB4 $
```

8 DMAP

Executive Instruction: EXIT

Conditionally terminates the execution of the DMAP program.

DMAP Instruction Syntax:

```
EXIT NTIME $
EXIT $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
NTIME	N/A	CONST	INT	Specifies the number of times the instruction is to be ignored before terminating the program.	**0**

Remarks:

1. **EXIT** may appear anywhere within the DMAP sequence.

Module: FBS

Solves the matrix equation:

$$\boldsymbol{L\,U\,X = \pm B}$$

where **L** and **U** are the lower and upper triangular factors of a matrix previously obtained from module **DECOMP**.

DMAP Instruction Syntax:

```
FBS   L,U,B / X / SYM / SIGN $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
L	INPUT	MATRIX	ANY	Name of nonstandard lower triangular factor matrix entity.	Required
U	INPUT	MATRIX	ANY	Name of nonstandard upper triangular factor matrix entity resulting from an unsymmetric decomposition.	None
B	INPUT	MATRIX	ANY	Name of matrix entity on right-hand-side.	Identity matrix
X	OUTPUT	MATRIX	ANY	Name of the solution matrix entity.	Required
SYM	INPUT	PARAM	INT	Symmetry option: **-1**: Automatically determine type of **FBS**. **0**: Perform unsymmetric **FBS**. **+1**: Perform symmetric **FBS**.	**-1**
SIGN	INPUT	PARAM	INT	Sign of **B** matrix: **-1**: Negative **+1**: Positive	**1**

Example:

1. Solve the two systems of equations:

   $\boldsymbol{A\,X_1 = B}$ and
   $\boldsymbol{A\,X_2 = -C}$

   using the **DECOMP** and **FBS** modules:

```
DECOMP   A / L,U / / / MINDIAG / DET / POWER / SING $
FBS      L,U,B / X1 $
FBS      L,U,C / X2 / / -1 $
```

8 DMAP

Executive Instruction: FILE

Defines special characteristics of an entity.

DMAP Instruction Syntax:

```
FILE        DB1=oplist / DB2=oplist / .... / DBn=oplist $
            oplist ⇒ [SAVE,][APPEND]
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	INPUT	ANY ENTITIES	N/A	Name of any database entities. [1]	Required
oplist	INPUT	KEYWORD	CHAR	Selects the file status as **SAVE**, **APPEND**, or both. [2,3]	One required

Remarks:

1. An entity name may only appear in a single **FILE** statement. Entities created by the **UAI/*NASTRAN*** preface may not appear in a **FILE** declaration.
2. The **SAVE** option indicates that the entity is to be saved while looping in a DMAP program.
3. Output entities which are generated within a DMAP loop are normally rewritten during each pass through the loop, unless the entity is declared **APPEND** in a **FILE** statement, in which case data may be added to the entity on successive passes through the DMAP loop.

Module: INPUTT1

Recovers database entities from an *internal* user file that was previously created by utility module **OUTPUT1**.

DMAP Instruction Syntax:

```
INPUTT1     / DB1,DB2,DB3,DB4,DB5 / POS / 'LFNAME' / 'LABEL' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	OUTPUT	ANY ENTITIES	N/A	Names of the entities to be recovered from file **LFNAME**.	At least one required
POS	INPUT	PARAM	INT	File positioning parameter. [1]	**0**
LFNAME	INPUT	PARAM	CHAR	The logical file name from which entities are read. [2]	**INP1**
LABEL	INPUT	PARAM	CHAR	Label which was written on the file by module **OUTPUT1**. [1]	**XXXXXXXX**

Remarks:

1. The table below describes the relationships of the **POS** and **LABEL** parameters and the action taken on **LFNAME** when they are selected:

POS	File Positioning Action	File Label Action
+n	Skips forward n database entities before reading.	Not checked.
0	Reads entities starting at the current position. The current position for the first use of a file is at the **LABEL**, in which case the **LABEL** counts as one entity.	Not checked.
-1	Rewinds before reading, and positions file past **LABEL**.	Checked.
-3	Prints entity names and then rewinds before reading.	Checked, warning check made.
-5	Searches for the **first** version of entity **DBi** requested. If any entity is not found, job execution terminates.	Checked.
-6	Searches for **final** version of entity **DBi** requested. If any entity is not found, job execution terminates.	Checked.
-7	Same as **-5**, except that if an entity is not found, a warning message is issued and execution continues.	Checked.
-8	Same as **-6**, except that if an entity is not found, a warning message is issued and execution continues.	Checked.

2. You must use the **ASSIGN** Executive Control command to define the logical file name **LFNAME** with the parameter **USE=INPUTT1**.
3. Files used by module **INPUTT1** must be created with module **OUTPUT1**.

Examples:

1. Read entities **A** and then **B** from file **INP1** starting from the current position of the file. Assume this is the first module to manipulate **INP1**, so that the file is automatically positioned at the beginning of the **LABEL**. In such cases, **POS** must be set to either **1** to skip past the label or **-1** to rewind the file and position it at the beginning of the first entity, **A**.

```
INPUTT1  / A,B,,, / $
```

2. Rewind **INP3** and check the label.

```
INPUTT1  / ,,,, / -1 / 'INP3' $
```

3. Starting from the current position, skip forward 4 entities on **INP2** and read the next five entities into **A, B, C, D** and **E**. Do not check the file label.

```
INPUTT1  / A,B,C,D,E / 4 / 'INP2' $
```

4. Request a complete list of entity names and a warning check of the file label. Then, perform the same operation as in Example 3.

```
INPUTT1  / ,,,, / -3 / 'INP2' $
INPUTT1  / A,B,C,D,E / 4 / 'INP2' $
```

5. For more examples using both **INPUTT1** and **OUTPUT1** see the description of module **OUTPUT1**.

Module: INPUTT2

Recovers database entities from an unformatted *external* user file, written by FORTRAN I/O functions, that was previously created either by utility module **OUTPUT2** or by your own FORTRAN program which is external to **UAI/*NASTRAN***.

DMAP Instruction Syntax:

```
INPUTT2     / DB1,DB2,DB3,DB4,DB5 / POS / 'LFNAME' / 'LABEL' / 'MODE' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	OUTPUT	ANY ENTITIES	N/A	Names of the entities to be recovered from file **LFNAME**.	At least one required
POS	INPUT	PARAM	INT	File positioning parameter. [1]	**0**
LFNAME	INPUT	PARAM	CHAR	Logical file name on which entities are written. [2]	Required
LABEL	INPUT	PARAM	CHAR	Label to be written on the file. [1]	**XXXXXXXX**
MODE	INPUT	PARAM	CHAR	Specifies the data storage mode. [3]	**SPARSE**

Remarks:

1. The file positioning and labelling options are given in the table below:

POS	File Positioning Action	File Label Action
+n	Skips forward n database entities before reading.	Not checked.
0	Reads entities starting at the current position. The current position for the first use of a file is at the **LABEL**, in which case the **LABEL** counts as one entity.	Not checked.
-1	Rewinds before writing, and positions file past **LABEL**.	Checked.
-3	Prints entity names and then rewinds before reading.	Checked, warning check made.
-5	Searches for the **first** version of entity **DBi** requested. If any entity is not found, job execution terminates.	Checked.
-6	Searches for **final** version of entity **DBi** requested. If any entity is not found, job execution terminates.	Checked.
-7	Same as **-5**, except that if an entity is not found, a warning message is issued and execution continues.	Checked.
-8	Same as **-6**, except that if an entity is not found, a warning message is issued and execution continues.	Checked.

2. You must use the **ASSIGN** Executive Control command to define the logical file name **LFNAME** with the parameter **USE=INPUTT2**.

3. The data storage modes are **SPARSE** and **FULL**. The mode must match that used when the file was created by module **OUTPUT2** or your FORTRAN program.

Examples:

1. Read entities **XX** and **YY** from file **MYIN2** starting from the current position of the file. Assume this is the first module to manipulate **MYIN2**, so that the file is automatically positioned at the beginning of the file label.

```
INPUTT2  / XX,YY,,, / / 'MYIN2' $
```

2. Rewind logical file **INFILE** and check the label.

```
INPUTT2  / ,,,, / -1 / 'INFILE' $
```

3. Starting from the current position, skip forward 2 entities on **MATFILE** and read the next four entities into **MAT1, MAT2, MAT3** and **MAT4**. Do not check the file label.

```
INPUTT2  / MAT1,MAT2,MAT3,MAT4, / 2 / 'MATFILE' $
```

4. Request a complete list of entity names and a warning check of the file label. Then, perform the same operation as in Example 3.

```
INPUTT2  / ,,,, / -3 / ' MATFILE' $
INPUTT2  / MAT1,MAT2,MAT3,MAT4, / 2 / 'MATFILE' $
```

Module: INPUTT3

Reads matrix data from an *external* user file, written by Fortran I/O functions, that was previously created in the Rockwell International data format and stores the matrices on the database.

DMAP Instruction Syntax:

```
INPUTT3 /DB1,DB2,DB3,DB4,DB5 / POS / 'LFNAME' / ERROR / TEST / PREC $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	OUTPUT	MATRIX	N/A	Names of the matrix entities to be recovered from file **LFNAME**.	At least one required
POS	INPUT	PARAM	INT	File positioning parameter. [1]	**1**
LFNAME	INPUT	PARAM	CHAR	File name from which entities are read. [2]	**USER1**
ERRFLG	INPUT	PARAM	INT	Specifies termination procedure upon error. [3]	**1**
TEST	INPUT	PARAM	INT	Specifies checking of matrix names. [4]	**0**
PREC	INPUT	PARAM	INT	Desired precision of data as stored on the database. [5]	**0**

Remarks:

1. The file positioning options, **POS**, are given in the table below:

POS	File Positioning Action
0	Begin reading from current file position.
-1	Rewind file and then read.
-2	Begin reading from the current file position, and when complete, rewind file.
-3	Rewind file, read, and when complete rewind the file again.

2. You must use the **ASSIGN** Executive Control command to define the logical file name **LFNAME** with the parameter **USE=INPUTT3**.
3. When the **ERRFLG** value is set to 1, **UAI*/NASTRAN*** will terminate execution if any requested matrix is not found on the input file. Setting this value to 0 will allow the program to continue.
4. When the **TEST** value is set to **1**, all matrix names on the file must match the **UAI*/NASTRAN*** names specified as **DBi** in the module calling sequence, or an error will be detected. Setting **TEST** equal **0** allows the matrix names to be changed.
5. The **PREC** value controls how matrix data are stored on the database according to the table below:

PREC	Result
0	The base precision used by **UAI*/NASTRAN*** for this computer will be used.
1	The output will be single precision regardless of the computer type.
2	The output will be double precision regardless of the computer type.

Module: INPUTT4

Recovers matrix database entities from a formatted or unformatted *external* user file, written by FORTRAN I/O functions, that was previously created either by utility module **OUTPUT4** or by your own FORTRAN program which is external to **UAI/*NASTRAN***.

DMAP Instruction Syntax:

```
INPUTT4     /DB1,DB2,DB3,DB4,DB5/ 'LFNAME' / POS / 'MODE' // MAXREC /
            'COMP' / 'PREC' / DPLACES $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	OUTPUT	MATRIX	N/A	Names of the matrix entities to be recovered from file **LFNAME**.	At least one required
LFNAME	INPUT	PARAM	CHAR	File name from which entities are read. [1]	Required
POS	INPUT	PARAM	INT	File positioning parameter. [2]	**0**
MODE	INPUT	PARAM	CHAR	Specifies storage mode of the matrices. [3]	**SPARSE**
MAXREC	INPUT	PARAM	INT	Maximum length of records written. [4]	**0**
COMP	INPUT	PARAM	CHAR	Specifies a compatibility mode of operation. [5]	**UAI**
PREC	INPUT	PARAM	CHAR	Specifies the numeric precision of the matrices that will written to the RUNDB database [6]: **INPUT**: Determined from file **LFNAME** **SINGLE**: Single precision **DOUBLE**: Double precision	**INPUT**
DPLACES	INPUT	PARAM	INT	Number of decimal places that were when writing real numbers on the Fortran output file. [7]	**9**

Remarks:

1. You must use the **ASSIGN** Executive Control command to define the logical file name **LFNAME** with the parameters **USE=INPUTT4** and **TYPE=FORMATTED** or **TYPE=BINARY**.
2. The file positioning options, **POS**, are given in the table below:

POS	File Positioning Action
0	Begin reading from current file position.
-1	Rewind file and then read.
-2	Begin reading from the current file position, and when complete, rewind file.
-3	Rewind file, read, and when complete rewind the file again.

3. The matrix storage modes are **SPARSE** and **FULL**. The **MODE** must match that used when the file was created by **OUTPUT4** or your FORTRAN program.
4. The maximum record length, **MAXREC**, must agree with that used when the file was created by **OUTPUT4** or your FORTRAN program.

5. The following file structure compatibility modes are available: **UAI** selects the file structure for **UAI/*NASTRAN*** Versions 11.6 and later. This format works for all size matrices, including matrices with more than 65535 rows and/or columns; **UAIOLD** selects the file structure which is compatible with **UAI/*NASTRAN*** Version 11.5 and earlier; **MSCSMALL** or **MSC** selects the file structure which is compatible with MSC/NASTRAN for matrices smaller than 65535 rows and/or columns; and **MSCBIG** for all size matrices, including matrices with more than 65535 rows and/or columns.
6. The **PREC** parameter defines the numeric precision that will be used when writing the file **LFNAME** ***ONTO*** the RUNDB database. It is ***NOT*** the precision of the data in **LFNAME**.
7. To define the FORTRAN format for reading real data you specify the number of decimal places, **DPLACES**. The format is then defined by:

   1P,*r*E*w*.*d*

   where:

   ***d* = DPLACES**
   ***w* = DPLACES + 7**
   r* = 80/*w

   For example, the default value of **9** results in the format:

   1P,5E16.9

8. The format of the FORTRAN files processed by **INPUTT4** are discussed in the **OUTPUT4** module description.

Executive Instruction: JUMP

Alters the normally sequential order of execution of DMAP modules by unconditionally transferring program control to a specified location in the DMAP program.

DMAP Instruction Syntax:

```
JUMP LABNAM $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
LABNAM	INPUT	LABEL	N/A	Specifies the name of a **LABEL** appearing in the DMAP program to which control will be transferred.	Required

Remarks:

1. Jumps must be forward in the DMAP sequence.
2. For backward jumps, see the **REPT** instruction.

Executive Instruction: LABEL

Assigns a label to a location in the DMAP program so that the location may be referenced by the DMAP instructions **JUMP, COND** and **REPT**.

DMAP Instruction Syntax:

```
LABEL  LABNAM $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
LABNAM	INPUT	LABEL	N/A	Specifies the name of a **LABEL**.	Required

Remarks:

1. The **LABEL** instruction is inserted immediately before the DMAP instruction to be executed after a transfer of control.

Module: LAMX

Converts the **LAMA** and **CLAMA** eigenvalue extraction summary tables to matrices.

DMAP Instruction Syntax:

```
LAMX, ,LAMA / LAMAT / / 'PREC' $
LAMX, ,CLAMA / CLAMAT / / 'PREC' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
LAMA	INPUT	TABLE	N/A	The output from module **SDR1** which contains the updated frequency, generalized mass and generalized stiffness data. [1]	Required
CLAMA	INPUT	TABLE	N/A	The output from module **CEAD** which contains the complex frequency data.	Required
LAMAT	OUTPUT	MATRIX	RSP or RDP	Eigenvalue extraction data in matrix form. [2]	Required
CLAMAT	OUTPUT	MATRIX	CSP or CDP	A diagonal matrix containing the complex eigenvalues. The order of the matrix is equal to the number of eigenvalues extracted.	Required
PREC	INPUT	PARAM	CHAR	Specifies the precision of **LAMAT** or **CLAMAT** as **SINGLE** precision or **DOUBLE** precision.	**SINGLE**

Remarks:

1. These data are updated in **SDR1** to reflect renormalization of modal data to a physical degree of freedom if there has been a dynamic reduction and if **MAX** normalization has been requested. Note that **LAMAX** is output from module **READ** and is compatible with **PHIA** while **LAMA** is output from **SDR1** and is compatible with PHIG.
2. **LAMMAT** consists of 5 columns and n rows, where n is the number of eigenvalues. Each row represents a mode with non-zero generalized mass. The five columns contain the following data:
   - Eigenvalue
   - Radian Frequency
   - Cyclic Frequency (Hz)
   - Generalized Mass
   - Generalized Stiffness

Module: MATGEN

Performs various matrix generation operations.

DMAP Instruction Syntax:

```
MATGEN      IN1,IN2,IN3 / OUT / OPTION / P1 / P2 / P3 / P4 / P5 /
            P6 / P7 / P8 / P9 $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
INi	INPUT	[1]	N/A	Input entities depending on **OPTION**. [1]	Required
OUT	OUTPUT	MATRIX	ANY	Matrix resulting from the selected **OPTION**. [1]	Required
OPTION	INPUT	PARAM	INT	Selects the option to be performed. [1]	Required
Pi	INPUT/ OUTPUT	PARAM	[1]	Parameters depending on **OPTION**. [1]	Required

Remarks:

1. A number of unrelated functions are provided. These are selected by parameter **OPTION** and perform the operations described in the tables below, note that the mode of all entities and parameters is INPUT unless otherwise specified:

OPTION	Description	Entity or Parameter	Meaning
1	Generate a real identity matrix.	**P1**	Order of the matrix.
		P2	Matrix option: **0**: Generate a diagonal matrix **>0**: Generate a skew diagonal matrix.
		P3	Precision flag: **0**: Use machine precision **1**: Single precision **2**: Double precision.
3	Generate a table of records with a fixed, constant value.	**P1**	Number of records to generate.
		P2	Length of each record in words. (Default=100)
		P3	Value to be stored in each word. (Default=0)

OPTION	Description	Entity or Parameter	Meaning
4	Generate a pattern matrix.	P1	Number of columns in OUT.
		P2	Number of rows in OUT.
		P3	Precision flag: 0: Use machine precision 1: Single precision 2: Double precision.
		P4	Number of terms per string (1≤P4≥P2).
		P5	Increment between strings (1≤P5≥P2).
		P6	Row number of first string in column 1 (1≤P6≥P2).
		P7	Increment to first row of subsequent columns (1≤P7≥P2).
		P8	Number of columns before returning to P6 (1≤P8≥P1).
5	Generate a matrix of pseudo-random numbers having values between 0.0 and 1.0.	P1	Number of columns in OUT.
		P2	Number of rows in OUT.
		P3	Precision flag: 0: Use machine precision 1: Single precision 2: Double precision.
		P4	Seed for random number gererator.
		P5	Average value of all random numbers generated. (Output)
6	Generate a partitioning vector.	P1	Number of rows (Required).
		P2,P4,P6,P8	Number of rows of "zero" terms.
		P3,P5,P7,P9	Number of rows of "one" terms.
		The output vector will contain P2 zeros, followed by P3 ones, followed by P4 zeros, an so forth. Any of the values P2-P9 may be zero. If the sum of P2 through P9 is less then output vector size, P1, the remaining terms will be zero. If the sum is greater the P1, the excess terms are ignored.	
7	Generate a null matrix.	P1	Number of rows in OUT.
		P2	Number of columns in OUT.
		P3	Matrix Form flag: 0: If P1 = P2, defaults to 1, otherwise defaults to 2. 1: Square matrix, P1 must equal P2. 2: Rectangular matrix.
		P4	Precision flag: 0: Use machine precision 1: Single precision 2: Double precision.
		P5	Complex flag: 0: Real matrix 1: Complex matrix
9	Generate a transformation between external and inter-nal sequence for g-size matrices.	IN1	Must be entity EQEXIN
		P1	Matrix option: 0: To compute the factor U_{int} = OUT U_{ext} 1: To compute the transposed factor U_{ext} = OUT U_{int}
		P2	Must be parameter LUSET or LUSETN depending on the location in the Rigid Format.

8 DMAP

OPTION	Description	Entity or Parameter	Meaning
101	Generate a g-set partition-ing vector from **BGPDT**.	**IN1**	Must be entity **BGPDT**
		P1	Matrix option: **-1**: To select SCALAR points. **n**: To select component **n**, where 1≤**n**≤6
		P2	Number of terms in the g-set (Required). **Note:** The parameter **LUSET** contains this number in most Rigid Formats.
		P3 (OUTPUT)	The number of selected components found. Set to **-1** if none were found.
		OUT will be a column vector of **P3** rows with a **1.0** in each row corresponding to the selected component.	
102	Generate a rigid body displacement shape matrix **D**.	**IN1**	Must be entity **BGPDT**
		IN2	Must be entity **EQEXIN**
		IN3	Must be entity **CSTM**
		P1	Reference point for computation: **0**: **D** is with respect to the origin of the Basic Coordinate System. **n**: **D** is with respect to GRID point **N**.
		P2	Precision flag: **0**: Use machine precision **1**: Single precision **2**: Double precision.
		OUT is always g-set rows by 6 columns. This matrix may then be partitioned to any other **UAI/*NASTRAN*** displacement set using module VEC to create the required partitioning vector.	

Examples:

1. Generate a double precision identity matrix, called **IDENT**, of order 1000:

```
MATGEN    /IDENT/1//1000//2 $
```

2. Generate a rectangular matrix, **RANDOM**, of random numbers which has 500 rows and 800 columns. Use machine precision:

```
MATGEN    /RANDOM/5/800/500 $
```

3. Generate a partitioning vector, **PVEC**, of length 1000 which has 100 zeros followed by 400 ones, 300 zeros, and 200 ones:

```
MATGEN    /PVEC/6/1000/100/400/300/200 $
```

Module: MATGPR

Prints, or writes as **DMIG** Bulk Data, matrices generated by the structural analysis Rigid Formats with external GRID point identification of each non-zero element.

DMAP Instruction Syntax:

For matrices generated in Rigid Formats 1-5 or matrices generated in Rigid Formats 7-12 prior to module **GKAD** (or **GKAM**):

```
MATGPR      GPL,USET,SIL,M // 'CSET' / 'RSET' / 'PRTOPT' /
            FILTER / FLAG / 'UNIT' / 'PREC' / 'NAME' $
```

For matrices generated in Rigid Formats 7-12 after module **GKAD** (or **GKAM**):

```
MATGPR      GPLD,USETD,SILD,M // 'CSET' / 'RSET' / 'PRTOPT' /
            FILTER / FLAG / 'UNIT' / 'PREC' / 'NAME' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
GPL **GPLD**	INPUT	TABLE	N/A	GRID Point List entity. GRID Point List (Dynamics) entity.	Required
USET **USETD**	INPUT	TABLE	N/A	Displacement Set entity. Displacement Set (Dynamics) entity.	Required
SIL **SILD**	INPUT	TABLE	N/A	Scalar Index List entity. Scalar Index List (Dynamics) entity.	Required
M	INPUT	MATRIX	ANY	Any structural matrix entity.	Blank
CSET	INPUT	PARAM	CHAR	The displacement set which defines the number of columns of the input matrix **M**. [1]	Blank
RSET	INPUT	PARAM	CHAR	The displacement set which defines the number of rows of the input matrix **M**.	[1]
PRTOPT	INPUT	PARAM	CHAR	Print option selector. **NULL**: Only null columns will be identified. **ALL**: Standard print **ALLP**: Standard print with complex numbers converted to Magnitude / Phase format.	**ALL**
FILTER	INPUT	PARAM	RSP	Value of filter. [2]	**0.0**
FLAG	INPUT	PARAM	RSP	Controls the use of **FILTER**. [2]	**0.0**
UNIT	INPUT	PARAM	CHAR	Selects the output device: **PRINT**: Output will be to the print file . **BULK**: Output will be **DMIG** entries written to the BULK file .	**PRINT**

8 DMAP

Arguments (Continued):

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
PREC	INPUT	PARAM	CHAR	When UNIT = **BULK**, selects the Bulk Data format: **STANDARD**: Ouput will be in standard, 8 character **HIGH**: Output will be in the 16 character, high precision format.	**HIGH**
NAME	INPUT	PARAM	CHAR	When UNIT = **BULK**, defines the **DMIG** matrix name.	Name of input matrix **M**

Remarks:

1. You must know which sets corresponds to the rows and columns of the matrix you wish to print. This is usually apparent from the DMAP name of the matrix entity. If the specified set does not match the data a warning message is issued and SCALAR degree of freedom notation is used.

   If **CSET** is left blank, then the corresponding data are considered scalar data as in the case of the columns of a load vector. In this case, the corresponding labeling is simply an integer number beginning with 1 and increasing by 1 for each column.

   If **RSET** is left blank, and **M** is a square matrix, then **RSET** is set equal to **CSET**. If **RSET** is blank and **M** is not a square matrix, then the row data are treated as scalar data as described above.

   The table below presents the allowed values for both **CSET** and **RSET**.

CSET or **RSET** Value	Matrix dimension is same as the set size of:	**CSET** or **RSET** Value	Matrix dimension is same as the set size of:
A	$\mathbf{U}_a$	**N**	$\mathbf{U}_n$
D	$\mathbf{U}_d$	**NE**	ξ_o
E	$\mathbf{U}_e$	**O**	$\mathbf{U}_o$
F	$\mathbf{U}_f$	**P**	$\mathbf{U}_p$
FE	ξ_i	**R**	$\mathbf{U}_r$
G	$\mathbf{U}_g$	**S**	$\mathbf{U}_s$
H	$\mathbf{U}_h$	**SB**	$\mathbf{U}_{sb}$
L	$\mathbf{U}_l$	**SG**	$\mathbf{U}_{sg}$
M	$\mathbf{U}_m$		

2. The table below presents how the **FILTER** and **FLAG** parameters control the operation of module **MATGPR**.

	FLAG = 0.0	**FLAG ≠ 0.0**
FILTER = 0.0	Print all terms of **M**.	Print all terms of **M**.
FILTER > 0.0	Print terms where **|term|≥FILTER**.	Print terms where **term ≥ FILTER** or **term ≤ 0.0**.
FILTER < 0.0	Print terms where **|term|≤FILTER**.	

3. When using this module with **USET** as input, the module must appear after **GP4** since entities generated by **GP4** are required inputs. When using with **USETD** input, the module must appear after **DPD** since entities generated by **DPD** are required inputs.

Module: MATMOD

Operates upon input database matrix entities based on a user specified option.

DMAP Instruction Syntax:

```
MATMOD      IN1,IN2,IN3,IN4,IN5,IN6/OUT1,OUT2/OPTION/IP1/IP2/IP3/
            RP1/RP2/IP4/IP5/IP6/IP7/IP8/'CP1' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
INi	INPUT	[1]	N/A	Input entities depending on **OPTION**.	Required
OUTi	OUTPUT	MATRIX	ANY	Matrix resulting from the selected **OPTION**.	Required
OPTION	INPUT	PARAM	INT	The **MATMOD** option. [1]	Required
IPi	INPUT	PARAM	INT	Parameters depending on **OPTION**. [1]	**0**
RPi	INPUT	PARAM	RSP	Parameters depending on **OPTION**. [1]	**0.0**
CP1	INPUT	PARAM	CHAR	Parameter depending on **OPTION**. [1]	Blank

Remarks:

1. A number of unrelated functions are provided. These are selected by parameter **OPTION** and perform the operations described in the tables below:

OPTION	Description	Entity or Parameter	Meaning
1	Extract a specified column from a matrix.	**IN1**	Matrix from which column will be extracted.
		OUT1	Matrix containing the extracted column.
		IP1	Starting column number to be extracted.
		IP2	Number of columns to be extracted (Default = 1).
		IP3	If 0, then the output matrix is created from the extracted columns. Otherwise, the extracted columns are appended to the existing matrix **OUT1**.
2	Filter small magnitude numbers from a matrix.	**IN1**	Matrix to be filtered.
		OUT1	Copy of **IN1** with terms smaller in magnitude than **RP1** set to zero.
		RP1	Value of filter.
3	Zeroes selected DOF in a *g-size* matrix containing only GRID points. (Scalar points are not supported.)	**IN1**	G-size square or symmetric matrix to be processed.
		OUT1	Copy of **IN1** with selected rows and columns set to zero.
		IP1	List of DOF to be zeroed for each GRID point. For example, '**123**' zeroes the translational degrees of freedom.

OPTION	Description	Entity or Parameter	Meaning
5	Generates a *g-size* matrix where the diagonal 6 x 6 matrix for each GRID point is defined by an input DMI matrix.	IN1	Input DMI matrix of order 6 x 6.
		IN2	SIL entity. If purged, then parameter IP1 must be specified.
		OUT1	*G-size* output matrix where a copy in IN2 has been inserted on the diagonal for each GRID point.
		IP1	Size of output matrix OUT1. This parameter is only used if the input SIL entity is purged.
		IP2	Must be 0.
		IP3	Precision of OUT1, either 1 (single) or 2 (double). The default is the precision of IN1.
7	Find the maximum value in each column of a matrix.	IN1	Matrix for which maximum values will be found.
		OUT1	Output column vector conatining the maximum value in each column of IN1.
8	Normalize a matrix	IN1	Input matrix to be normalized.
		OUT1	Output matrix as normalized version of IN1.
		OUT2	If CP1 equal to 'COL', this is a column vector whose row terms are the normalizers for each column of IN1.
		RP1	Returned as the real part of the largest absolute value in IN1.
		RP2	Returned as the complex part of the largest absolute value in IN1.
		CP1	If blank, matrix IN1 is normalized by the largest absolute value in IN1. If equal to 'COL', each column of IN1 is normalized to the largest absolute value in that column.
10	Computes the conjugate of a complex matrix	IN1	The input complex matrix.
		OUT1	The complex conjugate of IN1.
11	Creates a new BGPDT entity with updated GRID point coordinate values	IN1	The name of a matrix entity containing the new coordinate values (see below).
		IN2	The input BGPDT entity.
		OUT1	The new, updated BGPDT entity.
		IP1	If equal to zero (the default), IN1 contains new absolute coordinates. If IP2 is 1, IN1 contains deformed coordinate locations that are added to the input values.
		IP2	Column number from IN1 containing the new coordinates. (Default=1)
12	Find the null columns of a matrix.	IN1 thru IN6	Matrices to be simultaneously searched for null columns.
		OUT1	Output column vector containing a one in each row where all input matrices have null columns.
		OUT2	Output diagonal matrix containing a one on each diagonal where all input matrices have null columns. This matrix is purged if the matrices have no common null columns.
		IP1	The number of columns which are null in all input matrices. This parameter is -1 if the matrices have no common null columns.
		IP2	The number of matrices to be searched.

OPTION	Description	Entity or Parameter	Meaning
14	Filters numbers from a matrix subject to selected options. (See also option 2 above).	IN1	Matrix to be filtered.
		OUT1	Copy of IN1 containing filtered terms.
		IP1	If IP1 is zero and the resulting matrix OUT1 contains no nonzero terms, it is purged. If IP1 is not zero, then a null matrix is generated. (Default=0).
		IP2	If IP2 is less than zero, then all terms below the diagonal are set to zero. Similarly, if IP2 is greater than zero, then all terms above the diagonal are set to zero.
		IP3	Not used.
		RP1	Absolute filter value (see below). (Default=0.0)
		RP2	Relative filter value (see below). (Default=0.0)
		IP4	Scaling filter value (see below). (Default=0)

The three optional filters are defined as:

- Absolute filter: If $|\,\mathbf{IN1}_{ij}\,| < F_a$ then $\mathbf{OUT1}_{ij} = 0$
- Relative filter: If $\dfrac{|\,\mathbf{IN1}_{ij}\,|}{\sqrt{\mathbf{IN1}_{ii} \times \mathbf{IN1}_{jj}}} < F_r$ then $\mathbf{OUT1}_{ij} = 0$
- Scaling filter: $\mathbf{OUT1}_{ij} = \mathbf{IN1}_{ij} \times \dfrac{1}{10^{\mathbf{IP4}}}$

The default value for all filters is 0.0.

OPTION	Description	Entity or Parameter	Meaning
27	Converts a diagonal matrix (FORM=3) to a symmetric matrix (FORM=6).	IN1	Input diagonal matrix.
		OUT1	Output symmetric matrix.
28	Converts the first column of a square, rectangular or symmetric matrix to the a symmetric diagonal matrix (i.e. FORM=6)	IN1	Input square, rectangular or symmetric matrix.
		OUT1	Output symmetric matrix.
101	Generate an identity matrix.	OUT1	Resulting identity matrix.
		IP1	The number of rows and columns in OUT1.
104	Replace 0.0 terms on the diagonal of a matrix with RP1.	IN1	Input matrix to be modified.
		OUT1	Modified version of matrix IN1.
		RP1	Value to be stored on the diagonal of OUT1 in place of 0.0.
105	Converts a complex matrix to a real matrix	IN1	The input complex matrix.
		OUT1	The resulting real matrix.
		IP1	If set greater than zero, the job will terminate if IN1 does not exist.
		IP2	If less than or equal to zero, OUT1 is the amplitude of IN1. If IP2 is 1, OUT1 is the real part of IN1, and if it is 2, OUT2 is the imaginary part.

Module: MATPCH

Punches matrix entities onto **DMI** Bulk Data entries.

DMAP Instruction Syntax:

```
MATPCH      M1,M2,M3,M4,M5 /// 'N1' / 'N2' / 'N3' / 'N4' / 'N5' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
Mi	INPUT	MATRIX	ANY	Names of matrix entities to be written to the **BULK** file as **DMI** Bulk Data entries. [1]	At least one required
Ni	INPUT	PARAM	CHAR	Continuation field characters for Bulk Data entries for matrix **Mi**. [2]	Blank

Remarks:

1. You must use the **ASSIGN** Executive Control command to define a logical file with the parameter **USE=BULK**.
2. Only the first three characters of each **Ni** are used. These characters must be unique for all matrices which will be used as input to a subsequent **UAI*/NASTRAN*** execution. If an **Ni** is non-blank, the maximum number of Bulk Data images for the i^{th} matrix is 9,999. If **Ni** is blank, any number of images may be created.
3. The **DMI** Bulk Data entries may then be read as ordinary Bulk Data to define the matrix entity in a subsequent **UAI*/NASTRAN*** run.

Examples:

1. Punch matrices **A** and **B** onto **DMI** Bulk Data entries. Let **UAI*/NASTRAN*** automatically generate continuation fields:

```
MATPCH    A,B// $
```

2. Punch matrices **KGG** and **KNN** onto **DMI** Bulk Data entries using continuation prefixes of **KGG** and **KNN**, respectively:

```
MATPCH    KGG,KNN///'KGG'/'KNN' $
```

8 DMAP

Module: MATPRN

Prints general matrix database entities.

DMAP Instruction Syntax:

```
MATPRN      MAT1,MAT2,MAT3,MAT4,MAT5 // 'PRTOPT' / FILTER /
            FLAG / 'PREC' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
MATi	INPUT	MATRIX	ANY	Specifies names of matrix entities to be printed.	At least one required
PRTOPT	INPUT	PARAM	CHAR	Print option selector. **NULL**: Only null columns will be identified. **ALL**: Standard **MATPRN** printout. **ALLP**: Standard **MATPRN** printout with complex numbers converted to Magnitude / Phase format.	**ALL**
FILTER	INPUT	PARAM	RSP	Value of filter. [2]	**0.0**
FLAG	INPUT	PARAM	RSP	Controls the use of **FILTER**. [2]	**0.0**
PREC	INPUT	PARAM	CHAR	Selects the precision used to print the matrix: **SINGLE** uses a FORTRAN format of E13.5 and **DOUBLE** a format of D20.12.	**SINGLE**

Remarks:

1. Any or all input entities may be purged.
2. The table below presents how the **FILTER** and **FLAG** parameters control the operation of module **MATPRN**.

	FLAG = 0.0	**FLAG ≠ 0.0**
FILTER = 0.0	Print all terms of **MATi**.	Print all terms of **MATi**.
FILTER > 0.0	Print terms where **|term|**≥**FILTER**.	Print terms where **term** ≥ **FILTER** or **term** ≤ **0.0**.
FILTER < 0.0	Print terms where **|term|**≤**FILTER**.	

3. If any entity is not a matrix type, a table format will be used for printing.
4. Only non-zero matrix columns will be printed.

8 DMAP

Module: MATPRT

Prints matrix database entities associated with GRID points.

DMAP Instruction Syntax:

```
MATPRT      MAT // RORC / ENABLE $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
MAT	INPUT	MATRIX	ANY	Specifies the name of the matrix entity to be printed.	Required
RORC	INPUT	PARAM	INT	Specifies the storage mode of **MAT**: **0**: Stored by columns **1**: Stored by rows	**0**
ENABLE	INPUT	PARAM	INT	Enables or disables the function: <**0**: Disable print ≥**0**: Enable print	**0**

Remarks:

1. **MATPRT** should not be used if SCALAR or EXTRA points are present. For this case, use **MATPRN**.
2. Neither null columns nor a null, or purged, matrix are printed.

Module: MERGE

Forms matrix *A* from its partitions as generated by module **PARTN**:

$$\begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \Rightarrow A$$

DMAP Instruction Syntax:

```
MERGE        A11,A21,A12,A22,CP,RP / A / SYM $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
Aij	INPUT	MATRIX	ANY	Names of the matrix entities defining the partitions of **A**. [2]	[3]
CP	INPUT	MATRIX	RSP	Column partitioning vector. [4]	[5]
RP	INPUT	MATRIX	RSP	Row partitioning vector. [4]	[5]
A	OUTPUT	MATRIX	[6]	Resulting merged matrix. [2]	Required
SYM	INPUT	PARAM	INT	Symmetric partitioning flag: <**0** A symmetric merge is performed using **CP** for both vectors. ≥**0** An unsymmetric merge is performed using both **CP** and **RP**.	**-1** [5]

Remarks:

1. **MERGE** is the inverse of **PARTN** in the sense that if **A11, A12, A21, A22** were produced by **PARTN** using **RP**, **CP** and **SYM** from **A**, **MERGE** will produce **A**.
2. The sizes of the input partitions must be:

Matrix Partition	Number of Rows	Number of Columns
A11	Number of **0**'s in **RP**	Number of **0**'s in **CP**
A12	Number of **0**'s in **RP**	Number of **1**'s in **CP**
A21	Number of **1**'s in **RP**	Number of **0**'s in **CP**
A22	Number of **1**'s in **RP**	Number of **1**'s in **CP**
A	Number of rows in **RP**	Number of rows in **CP**

3. If a given partition is null, its position in the input list should be left blank.
4. A partitioning vector is simply a column of **0**s, **1**s and **2**s. A **0** indicates that the given row, or column, is a member of the partition, and a **1** indicates it is a member of the second partition. A value of **2** indicates that this position in the partitioning vector is to be ignored. The size of matrix **A.**

5. If **RP** is given and **CP** is not, and **SYM≥0**, then a row partitioning will occur as shown in the examples. If **CP** is given and **RP** is not, and **SYM≥0**, then a column partitioning will occur as shown in the examples. If **SYM<0**, a symmetric partitioning is done, that is **CP** is used for both the column and row partitioning vectors. One of the vectors **CP** or **RP** must be given.
6. The type of **A** is the maximum of the types of **Aij**.

Module: MPY3

Performs an orthogonal matrix triple multiply and add of the form:

$$X = A^T BA + E$$

DMAP Instruction Syntax:

```
MPY3  A,B,E / X / FORM $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	MATRIX	ANY	Name of the pre- and post-multiplier matrix.	Required
B	INPUT	MATRIX	ANY	Name of second multiplier matrix.	Required
E	INPUT	MATRIX	ANY	Name of the optional addend matrix.	Blank
X	OUTPUT	MATRIX	[2]	Name of the resultant matrix.	Required
FORM	INPUT	PARAM	INT	The form of the operation: **0**: Performs: $\mathbf{A^TBA + E}$ **1**: Performs: $\mathbf{A^TB + E}$ **2**: Performs: **BA** + **E**	Required

Remarks:

1. The matrices **A**, **B** and **X** may not be purged.
2. The type of **X** is the maximum of the types of **A, B** and **E**.

8 DMAP

Module: MPYAD

Multiplies two matrices and, optionally, adds a third matrix to the product. Options allow the computation of:

$$X = \pm A\,B \pm C$$
$$X = \pm A^T B \pm C$$

DMAP Instruction Syntax:

```
MPYAD       A,B,C / X / TRAN / SIGNAB / SIGNC / TYPE $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	MATRIX	ANY	Name of the pre-multiplier matrix. [1]	Required
B	INPUT	MATRIX	ANY	Name of post-multiplier matrix. [1]	Required
C	INPUT	MATRIX	ANY	Name of the optional addend matrix.	Blank
X	OUTPUT	MATRIX	[2]	Name of the resultant matrix. [1]	Required
TRAN	INPUT	PARAM	INT	The transpose option: **0**: Perform **AB** **1**: Perform $\mathbf{A^TB}$	0
SIGNAB	INPUT	PARAM	INT	The specification of the **AB** term: **+1**: Use **+AB** **0**: **AB** is not present **-1**: Use **-AB**	**1**
SIGNC	INPUT	PARAM	INT	The specification of the **C** term: **+1**: Use **+C** **0**: **C** is not present **-1**: Use **-C**	**1**
TYPE	INPUT	PARAM	INT	Numeric type of the operation: **1**: Single precision **2**: Double precision 3: Complex single precision 4: Complex double precision	[3]

Remarks:

1. The matrices **A**, **B**, and **X** may not be purged.
2. The type of **X** is the maximum of the types of **A**, **B**, and **C**.
3. The default **PREC** depends upon your **UAI/*NASTRAN*** host computer.

Example:

1. Perform the operation $X=A^TB - C$:

```
MPYAD       A,B,C/X/1/1/-1/1 $
```

Module: NORM

This module performs various column oriented normalization procedures as described below.

DMAP Instruction Syntax:

```
NORM        MATIN, MASS / MATOUT / NCOL / NROW / XNORM / OPT $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
MATIN	INPUT	MATRIX	ANY	Name of the matrix to be normalized. [1]	Required
MASS	INPUT	MATRIX	REAL	"Mass" matrix to control the normalization process.	[1]
MATOUT	OUTPUT	MATRIX	[2]	Name of the resultant matrix.	Required
NCOL	OUTPUT	PARAM	INT	The number of columns of **MATIN**.	Required
NROW	OUTPUT	PARAM	INT	The number of rows of **MATIN**.	Required
XNORM	OUTPUT	PARAM	REAL	The maximum (absolute value) normalizing value over all columns of **MATIN**.	0.0
OPT	INPUT	PARAM	INT	Parameter to select the desired calculations: **1**: Normalize **MATIN** by the maximum element in each column to produce **MATOUT** **2**: Create a single ***column*** matrix **MATOUT** which contains the SRSS value for each ***row*** of **MATIN** [3] **3:** Normalize **MATIN** using the **MASS** matrix [4] [4]	1

Remarks:

1. The matrices **MATIN** and **MATOUT** may not be purged. Matrix **MASS** is used only if **OPT** is **3**.
2. The type and precision of **MATOUT** are the same as the type and precision of **MATIN**.
3. The SRSS calculation is the square root of the sum of the squares of each column value, for each row, of matrix **MATIN**.
4. If **OPT** is **3**, the matrix **MASS** is input and **MATOUT** will be normalized to produce an ***identity diagonal matrix*** from the triple product:

   $[MATIN]^T [MASS] [MATIN]$

   This operation assumes that the vectors of **MATIN** are orthogonal with respect to matrix **MASS**.

8 DMAP

Module: OUTPUT1

Writes database entities and an optional label onto an *export* file suitable for *importing* into a subsequent **UAI/*NASTRAN*** job.

DMAP Instruction Syntax:

```
OUTPUT1     DB1,DB2,DB3,DB4,DB5 / / POS / 'LFNAME' / 'LABEL' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	INPUT	ANY ENTITIES	N/A	Names of the entities to be written to logical file **LFNAME**.	[1]
POS	INPUT	PARAM	INT	File positioning parameter. [1,2]	**0**
LFNAME	INPUT	PARAM	CHAR	Logical file name on which entities will be written. [3]	Required
LABEL	INPUT	PARAM	CHAR	Label to be written on the file. [2]	**XXXXXXXX**

Remarks:

1. You may omit all **DBi** entity names to position the file prior to writing the entities. Caution should be employed when positioning a file with **OUTPUT1** because information may be inadvertently destroyed through improper positioning. Even if no entities are written, **an EOF will be written** at the completion of each use. This effectively destroys any data beyond the current position.
2. The table below describes the relationships of the **POS** and **LABEL** parameters and the action taken on **LFNAME** when they are selected:

POS	File Positioning Action	File Label Action
+n	Skips forward n entities before writing.	Not written
0	Writes entities starting at the current position. The current position for the first use of a file is at the **LABEL** in which case **LABEL** counts as one entity.	Not written
-1	Rewinds before writing. (This is dangerous!) An End-of-File is written at the end of each call to **OUTPUT1**.	Written
-3	Rewinds file, prints entity names and then begins writing after the last entity on the file.	Not written, warning check made

3. You must use the **ASSIGN** Executive Control command to define the logical file name **LFNAME** with the parameter **USE=OUTPUT1**.

8 DMAP

Examples:

1. Write entities **A** and **B** onto file **DATA** starting at the current position of the file:

```
OUTPUT1 A,B,,, / / / 'DATA' $
```

   Note that if this is the first write operation for **LFNAME**, it is recommended that you precede the write operation with:

```
OUTPUT1 ,,,, / / -1 / 'DATA' $
```

   which will automatically label the file and position it at the beginning and after the label.

2. Rewind file **INPUT2** and place the default label on the file. Note that any entities on the file will be destroyed:

```
OUTPUT1,  ,,,, / / -1 / 'INPUT2' $
```

3. Starting from the current position, skip forward 4 entities on file **INPUT1** and write **A, B, C, D**, and **E** as the next five entities.

```
OUTPUT1   A,B,C,D,E / /4 / 'INPUT1' $
```

   Note that the skip positioning feature cannot be used if the current position of **INPUT1** is forward of a just previously written entity end-of-file, or if the file has not been labeled.

4. Consider the following instructions:

```
OUTPUT1,  ,,,, / / -3 / 'MYFILE' $
OUTPUT1   A,B,C,D,E / / 4 / 'MYFILE' $
```

   This is an invalid sequence because the first **OUTPUT1** positions the file at the end of all entities on the file.

5. Use module **INPUTT1** to check for a correct file label on logical file **DATA**, to print a complete list of entity names and to rewind the file. Then, use **OUTPUT1** to skip forward 4 entities and write **A, B, C, D**, and **E**:

```
INPUTT1 / ,,,, / -3 / 'DATA' $
OUTPUT1   A,B,C,D,E // 4 / 'DATA' $
```

6. Perform the following operations: (1) Obtain a list of all entities on **SAVFIL**; (2) Skip past the first four entities, replace the next two with entities **A** and **B**, and retain the next three entities; and (3) Obtain a list of all entities on **SAVFIL** after writing the new data:

`INPUTT1 / ,,,, / -3 /'SAVFIL' $`	Performs the first operation and rewinds **SAVFIL**, the default file.
`INPUTT1 / ,,T1,T2,T3 / 6 /'SAVFIL' $`	Recovers entities 7, 8 and 9. This is necessary because they are effectively destroyed if anything is written before them on **SAVFIL**.
`INPUTT1 / ,,,, / -1 /'SAVFIL' $`	Rewind **SAVFIL**. Note that **INPUTT1** must be used to rewind the file so that the previously written data are not destroyed.
`OUTPUT1 A,B,T1,T2,T3 // 4 /'SAVFIL' $`	Satisfies second objective.
`OUTPUT1, ,,,, // -3 /'SAVFIL' $`	Accomplishes third objective and leaves **SAVFIL** positioned after the ninth file, ready to receive additional entities.

7. Perform the following operations: (1) Write entities **A**, **B** and **C** on **OFILE**; (2) Obtain a list of all entities on **OFILE**; (3) Add entities **D** and **E** to the file; and, finally, (4) Obtain a list of all entities on **OFILE**.

`OUTPUT1 A,B,C,, // -1 / 'OFILE' $`	Performs the first operation.
`OUTPUT1, ,,,, // -3 / 'OFILE' $`	Performs the second operation.
`OUTPUT1 D,E,,, /// 'OFILE' $`	Performs the third operation.
`OUTPUT1, ,,,, // -3 / 'OFILE' $`	Performs the fourth operation.

Module: OUTPUT2

Writes entities and a label onto an *export* file suitable for reading by an external FORTRAN program using unformatted FORTRAN I/O functions, or *importing* into a subsequent **UAI/*NASTRAN*** job.

DMAP Instruction Syntax:

```
OUTPUT2     DB1,DB2,DB3,DB4,DB5 // POS / 'LFNAME' / 'LABEL' /
            MAXREC / 'MODE' / 'ALIAS1' / 'ALIAS2' / 'ALIAS3' /
            'ALIAS4' / 'ALIAS5' /$
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	INPUT	ANY ENTITIES	N/A	Names of the entities to be written on **LFNAME**.	Required
POS	INPUT	PARAM	INT	File positioning parameter. [1]	**0**
LFNAME	INPUT	PARAM	CHAR	Logical file name on which entities will be written. [2]	Required
LABEL	INPUT	PARAM	CHAR	Label to be written on the file. [1]	**XXXXXXXX**
MAXREC	INPUT	PARAM	INT	Maximum length of records written. [3]	[3]
MODE	INPUT	PARAM	CHAR	Specifies the data storage mode. [3,5]	**SPARSE**
ALIASi	INPUT	PARAM	CHAR	Defines an **ALIAS** for **DBi** that is written on the logical file. [4]	The name of **DBi**

Remarks:

1. The table below describes the relationships of the **POS** and **LABEL** parameters and the action taken on **LFNAME** when they are selected:

POS	File Positioning Action	File Label Action
+n	Skips forward n entities before writing.	Not written.
0	Writes entities starting at the current position. The current position for the first use of a file is at the **LABEL**, In which case **LABEL** counts as one entity.	Not written.
-1	Rewinds before writing.	Written.
-3	Rewinds file, prints entity names and then writes after the last entity on the file.	Not written, warning check made.
-9	Writes a final EOF on the file.	Not written.

Caution should be employed positioning a file with **OUTPUT2** because information may be inadvertently destroyed through improper positioning. Even if no entities are written, **an EOF will be written** at the completion of each use. This effectively destroys any data beyond the current position.

2. You must use the **ASSIGN** Executive Control command to define the logical file name **LFNAME** with the parameter **USE=OUTPUT2**. You may combine the output from this module and the **BINARY** output from module **OUTPUT4** on the same file.

3. The maximum record length, **MAXREC**, refers to the length of the FORTRAN physical records. The module will automatically split any **UAI*/NASTRAN*** logical records into the required number of physical records. The default for **MAXREC** depends on your **UAI*/NASTRAN*** host computer.

4. The **ALIASi** parameters can be used to change the entity names that are specified by **DBi**. The **ALIASi** is written to file **LFNAME**. This feature is useful when interfacing with post-processors that expect entities to have names which do not correspond to those in the DMAP program.

5. This module writes the specified entities to **LFNAME** using FORTRAN unformatted WRITE statements. As a result, you must read the data into your application program in exactly the same manner in which it was written. Because of the open-ended nature of the **UAI*/NASTRAN*** data structures, the format used generally contains pairs of FORTRAN records. The first record of the pair defines the number of words in the next record. Special codes are inserted in the data to tell you when you have reached the end of a logical record or the end of a file. The end-of-logical-record is signified by a negative value in the first physical record, and the end-of-file by a zero value in the first physical record.

   The matrix storage modes are **SPARSE** and **FULL**. The **SPARSE** format represents each matrix column as a set of one or more ***strings***. A string contains a ***header*** which is followed by the actual matrix terms contained within the string. In this manner, only the nonzero terms of the matrix are actually stored on the disk file. The **FULL** representation stores all terms in a column. Although this mode may take more disk space because some intermediate zero values are stored, it is easier to use in your own FORTRAN program. There is also a **SPARSE** mode for table entities. The actual data content for the **SPARSE** and **FULL** modes is the same, but the format of records in the **SPARSE** table is compatible with that of the **SPARSE** matrix.

   The output file, **LFNAME**, contains a label file and, for each entity, a file containing two logical header records followed by the contents of the entity. Each entity is comprised of an entity header (for tables only) followed by data records.

The format of the label file is given in the following table:

FORMAT OF THE LABEL FILE							
Entity Record Number	Logical Record Number	Physical Record Number	Physical Record Contents				Notes
			WORD1	WORD 2	...	...	
None	1	1	3				Indicates that the next record contains three words.
		2	MONTH	DAY	YEAR		Each as an integer value.
		3	7				Indicates that the next record contains seven words.
		4	X(1)	X(2)	...	X(7)	Seven words containing the characters: 'NASTRAN FORT TAPE ID CODE - '
		5	2				Indicates that the next record contains two words.
		6	LABEL(1)	LABEL(2)			Two words containing the file LABEL.
		7	-1				Indicates End-of-Logical-Record.
		8	0				Indicates End-of-File.

6. The format for the **SPARSE** storage modes are given in the following tables:

HEADER RECORD Data Format: SPARSE Storage Mode							
Entity Record Number	Logical Record Number	Physical Record Number	Physical Record Contents				Notes
			WORD1	WORD 2	...	...	
None	1	1	2				Indicates that the next record contains two words.
		2	NAME(1)	NAME(2)			Two words containing the file NAME.
		3	-1				Indicates End-of-Logical-Record 1.
	2	4	7				Indicates that the next record contains seven words.
		5	T(1)	T(2)	...	T(7)	Seven-word entity trailer array.
		6	-2				Indicates End-of-Logical-Record 2.

TABLE Data Format: SPARSE Storage Mode							
Entity Record Number	Logical Record Number	Physical Record Number	Physical Record Contents				Notes
			WORD1	WORD 2	...	...	
0	3	7	1				Indicates start of new logical record.
		8	0				Indicates that this is a table record.
		9	LREC				Length of the record in words.
		10	D(1)	D(2)	...	D(LREC)	Record data items.
		11	-3				Indicates End-of-LogicalRecord 3.
1	4	12	1				Indicates start of new logical record.
		13	0				Indicates that this is a table record.
		14	LREC				Length of the record in words.
		15	D(1)	D(2)	...	D(LREC)	Record data items.
		16	-4				Indicates End-of-Logical-Record 4.
2	5	17	1				Zero length record.
		18	0				
		19	-5				
...	...	...	...	...	...	...	
n-1	n+2	Continues with serial physical record numbers	1				Indicates start of new logical record.
			0				Indicates that this is a table record.
			LREC1				This logical record spans two physical records.
			D(1)	D(2)	...	D(LREC1)	
			LREC2				
			D(1)	D(2)	...	D(LREC2)	
			-(n+2)				Indicates End-of-Logical-Record (n+2).
n	n+3	Continues	1				Indicates start of new logical record.
			0				Indicates that this is a table record.
			0				Indicates End-of-File.

8 DMAP

MATRIX Data Format: SPARSE Storage Mode							
Entity Record Number	Logical Record Number	Physical Record Number	Physical Record Contents				Notes
			WORD1	WORD 2	...	...	
0	3	7	1				Indicates start of new logical record.
		8	0				Indicates that this is a table record.
		9	2				Length of the record in words.
		10	NAME(1)	NAME(2)			Entity name.
		11	-3				Indicates End-of-LogicalRecord 3.
1	4	Continues with serial physical record numbers	1				Indicates start of new logical record.
			COLID				Indicates the column number of the matrix.
			NW				**NW** is length of the column in words.
			IROW	D(1)	...	D(NW)	**IROW** is the first nonzero row position followed by the terms in the string.
			-4				Indicates End-of-Logical-Record 4.
...	...	...	...	...	...	...	
n-1	n+2	Continues	1				Indicates start of new logical record.
			COLID				Indicates the column number of the matrix.
			NW1				This logical record contains several strings.
			IROW1	D(1)	...	D(NW1)	
			NW2				
			IROW2	D(1)	...	D(NW2)	
			-(n+2)				Indicates End-of-Logical-Record (n+2).
n	n+3	Continues	1				Indicates start of new logical record.
			0				Indicates that this is a table record.
			0				Indicates End-of-File.

8 DMAP

7. The format for the **FULL** storage modes are given in the following tables:

HEADER RECORD Data Format: FULL Storage Mode							
Entity Record Number	Logical Record Number	Physical Record Number	Physical Record Contents				Notes
			WORD1	WORD 2	...	...	
None	1	1	2				Indicates that the next record contains two words.
		2	NAME(1)	NAME(2)			Two words containing the file **NAME**.
		3	-1				Indicates End-of-Logical-Record 1.
	2	4	8				Indicates that the next record contains eight words.
		5	T(1)	...	T(7)	TYPE	Seven-word entity trailer array and entity type (0 for table, 1 for matrix).
		6	-2				Indicates End-of-Logical-Record 2.

TABLE Data Format: FULL Storage Mode							
Entity Record Number	Logical Record Number	Physical Record Number	Physical Record Contents				Notes
			WORD1	WORD 2	...	...	
0	3	7	LREC				Length of the record in words.
		8	D(1)	D(2)	...	D(LREC)	Record data items.
		9	-3				Indicates End-of-LogicalRecord 3.
1	4	10	-4				Indicates End-of-Logical-Record 4. (Null record)
...	...	...	...	...	...	...	
n	n+3	Continues with serial physical record numbers	LREC1				This logical record spans two physical records.
			D(1)	D(2)	...	D(LREC1)	
			LREC2				
			D(1)	D(2)	...	D(LREC2)	
			-(n+3)				Indicates End-of-Logical-Record (n+3).
		Continues	0				Indicates End-of-File.

MATRIX Data Format: FULL Storage Mode							
Entity Record Number	Logical Record Number	Physical Record Number	Physical Record Contents				Notes
			WORD1	WORD 2	...	...	
0	3	7	2				Length of the record in words.
		8	NAME(1)	NAME(2)			Entity name.
		9	-3				Indicates End-of-Logical-Record 3.
1	4	Continues with serial physical record numbers	NW				NW is length of the column in words
			D(1)	D(2)	...	D(NW)	Record data items.
			-4				Indicates End-of-Logical-Record 4.
...	...	...	...	...	...	...	
n	n+3	Continues	NW1				This logical record spans two physical records.
			D(1)	D(2)	...	D(NW1)	
			NW2				
			D(1)	D(2)	...	D(NW2)	
			-(n+3)				Indicates End-of-Logical-Record (n+3).
		Continues	0				Indicates End-of-File.

8 DMAP

Module: OUTPUT3

Punches matrix entities onto **DMI** Bulk Data entries.

DMAP Instruction Syntax:

```
OUTPUT3     M1,M2,M3,M4,M5 // / 'N1' / 'N2' / 'N3' / 'N4' / 'N5' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
Mi	INPUT	MATRIX	ANY	Names of matrix entities to be written to the **BULK** file as **DMI** Bulk Data entries. [1]	At least one required
Ni	INPUT	PARAM	CHAR	Continuation field characters for Bulk Data entries for matrix **Mi**. [2]	Blank

Remarks:

1. You must use the **ASSIGN** Executive Control command to define a logical file with the parameter **USE=BULK**.
2. Only the first three characters of each **Ni** are used. These characters must be unique for all matrices which will be used as input to a subsequent **UAI*/NASTRAN*** execution. If an **Ni** is non-blank, the maximum number of Bulk Data images for the i^{th} matrix is 9,999. If **Ni** is blank, any number of images may be created.
3. The **DMI** Bulk Data entries may then be read as ordinary Bulk Data to define the matrix entity in a subsequent **UAI*/NASTRAN*** run.

Examples:

1. Punch matrices **A** and **B** onto **DMI** Bulk Data entries. Let **UAI*/NASTRAN*** automatically generate continuation fields:

```
OUTPUT3  A,B// $
```

2. Punch matrices **KGG** and **KNN** onto **DMI** Bulk Data entries using continuation prefixes of **KGG** and **KNN**, respectively:

```
OUTPUT3  KGG,KNN///'KGG'/'KNN' $
```

8 DMAP

Module: OUTPUT4

Writes **UAI/*NASTRAN*** matrix entities to an *export* file using either binary or formatted FORTRAN I/O functions.

DMAP Instruction Syntax:

```
OUTPUT4     DB1,DB2,DB3,DB4,DB5 // 'LFNAME' / POS / 'MODE' //
            MAXREC / PREC / 'COMP' / 'ALIAS1' / 'ALIAS2' / 'ALIAS3' /
            'ALIAS4' / 'ALIAS5' / 'DPLACES' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	INPUT	MATRIX	ANY	Names of matrix entities to be written to file **LFNAME**.	At least one required
LFNAME	INPUT	PARAM	CHAR	Logical file name on which entities **DBi** are written. [1]	Required
POS	INPUT	PARAM	INT	File positioning parameter. [2]	**0**
MODE	INPUT	PARAM	CHAR	Specifies storage mode of the matrices. [3]	**SPARSE**
MAXREC	INPUT	PARAM	INT	Maximum length of records written.	[4]
PREC	INPUT	PARAM	CHAR	Specifies the numeric precision of the matrices: **SINGLE**: Single precision **DOUBLE**: Double precision	[5]
COMP	INPUT	PARAM	CHAR	Specifies a compatibility mode of operation. [6,7]	**UAI**
ALIASi	INPUT	PARAM	CHAR	Specifies matrix entity alias names for **DBi** which are written on file **LFNAME**. [8]	**DBi**
DPLACES	INPUT	PARAM	INT	Number of decimal places used in Fortran output file for real numbers. [9]	**9**

Remarks:

1. You must use the **ASSIGN** Executive Control command to define the logical file name **LFNAME** with the parameters **USE=OUTPUT4** and **TYPE=FORMATTED** or **TYPE=BINARY**. When **TYPE=BINARY**, you may combine the output from this module and module **OUTPUT2** on the same file.
2. The file positioning options, **POS**, are given in the table below:

POS	File Positioning Action
0	Begin writing at the current file position.
-1	Rewind the file before writing.
-2	Begin writing at the current file position, and when complete, write an EOF and rewind the file.
-3	Rewind the file, write, and when complete, write an EOF and rewind the file again.

3. The matrix storage modes are **SPARSE** and **FULL**. The **SPARSE** format represents each matrix column as a set of one or more ***strings***. A string contains a ***header*** which is followed by the actual matrix terms contained within the string. In this manner, only the nonzero terms of the matrix are actually stored on the disk file. The **FULL** representation stores all terms in a column from the first nonzero to the last nonzero, i.e. the band of the matrix. Although this mode may take more disk space because some intermediate zero values are stored, it is easier to use in your own Fortran program.
4. **MAXREC** specifies the maximum record length of the physical Fortran records which will be written. **UAI/*NASTRAN*** automatically splits any logical records across the required number of physical records. The default **MAXREC** is the available memory on your **UAI/*NASTRAN*** host computer.
5. The default **PREC**ision depends on your **UAI/*NASTRAN*** host computer.
6. The following file structure compatibility modes are available:

   UAI selects the file structure for **UAI/*NASTRAN*** Versions 11.6 and later. This format works for all size matrices, including matrices with more than 65535 rows and/or columns.

   UAIOLD selects the file structure which is compatible with **UAI/*NASTRAN*** Version 11.5 and earlier.

   MSCSMALL or **MSC** selects the file structure which is compatible with MSC/NASTRAN for matrices smaller than 65535 rows and/or columns, and **MSCBIG** selects the structure for all size matrices, including matrices with more than 65535 rows and/or columns.
7. The format for binary files in both the **FULL** and **SPARSE** storage modes, along with examples of their usage, are given in the following tables.
8. The **ALIASi** parameters can be used to change the entity names that are specified by **DBi**. The **ALIASi** names are written to the file **LFNAME**. This feature is useful when interfacing to post-processors that expect entities to have names which do not correspond to those in the DMAP program.
9. To define the FORTRAN format for writing real data you specify the number of decimal places, **DPLACES**. The format is then defined by:

   1P,*r*E*w*.*d*

   where:

   ***d* = DPLACES**
   ***w* = DPLACES + 7**
   r* = 80/*w

   For example, the default value of **9** results in the format:

   1P,5E16.9

BINARY Output Format: FULL Storage Mode

Record	Contents						
1	NCOL	NROW	FORM	PREC	NAME(1)	NAME(2)	

The first record contains the basic data describing the matrix: the number of columns, **NCOL**, the number of rows, **NROW**, the matrix **FORM**, the matrix numerical **PREC**ision and the matrix **NAME** which is stored in two computer words. A typical FORTRAN read statement for this data is:

```
READ(IUNIT) NCOL,NROW,IFORM,ITYPE,NAME(1),NAME(2)
```

2	COLID	IROW	NW	D(1)	D(2)	...	D(NW)
3	COLID	IROW	NW	D(1)	D(2)	D(NW)	
	...	...	...	...	...	...	...
N-1	NCOL+1						

Each nonnull column is found in a separate record as shown. The first word in the record is the column number, **COLID**. This is followed by the number of the first nonzero row in the column, **IROW**. The next entry specifies the number of words that follow and contain the actual terms in the column. Remember that the actual number of terms depends on the **PREC** of the matrix. Because there are an unspecified number of output columns, depending on the number of null columns, a special record is written to tell you that the data for the matrix is finished. It contains a **COLID** which is one greater than the number of columns in the matrix.

Remember that you must read the entire column data record in a single FORTRAN read statement. The actual FORTRAN code requires that you use the **PREC** variable so that you can convert the number of words in the column, **NW**, to the actual number of matrix terms in the column. In addition, the reading of the file depends on the type of array into which you read the data. Suppose that your matirx has a **PREC** of 1 indicating real, single precision data. Then one way to read a column would be:

```
DIMENSION X(100)
...
READ(IUNIT) COLID,IROW,NW,(X(J),J=1,NW)
```

On the other hand, if your matrix was real, double precision, then you could use either

```
DIMENSION X(100)
DOUBLE PRECISION DX(50)
EQUIVALENCE (X(1),DX(1))
...
READ(IUNIT) COLID,IROW,NW,(X(J),J=1,NW)
```

or

```
DOUBLE PRECISION DX(50)
...
READ(IUNIT) COLID,IROW,NW,(DX(J),J=1,NW/2)
```

Note that the number of words must be divided by two to determine the number of terms.

8 DMAP

BINARY Output Format: SPARSE Storage Mode							
Record	**Contents**						
1	**NCOL**	**NROW**	**FORM**	**PREC**	**NAME(1)**	**NAME(2)**	
	This record contains the same information as in the **FULL** sotrage mode.						
2	**COLID**	**0**	**NW**	**STRING1**	**STRING2**	...	**STRINGm**
3	**COLID**	**0**	**NW**	**STRING1**	**STRING2**	**STRING3**	
	...	...	...	...	...	...	...
N-1	**NCOL+1**						
	Again, each nonnull column is found in a separate record as shown. The first word in the record is the column number, **COLID**. This is followed by the integer value zero. The next word, **NW**, is length of the remainder of the record including string headers and data. Each string has the following format:						
	LS	**IROW**	**TERM(1)**	**TERM(2)**	...	...	**TERM(LS)**
	The first non-zero row in the string is **IROW**. The number of words, or length, of the string is **LS**, and this includes the 2 words **LS** and **IROW**.						
	The following program will read a real, single precision matrix into array **a**:						

```
      INTEGER IZ(100,NAME(2),FORM,PREC,COLID
      REAL Z(100,A(100,100)

      OPEN(1,FILE='filename',STATUS='OLD',FORM='UNFORMATTED')

      READ(1) NCOL,NROW,FORM,PREC,NAME

10    READ(1) COLID,NULL,NW,(Z(J),J=1,NW)
      IF( COLID .GT. NCOL ) STOP

      J = 1
20    LS = IZ(J)
      IROW = IZ(J+1)
      J = J + 2
      DO 30 I = IROW, (IROW+LS-3)
            A(COLID,I) = Z(J)
            J = J + 1
30    CONTINUE
      IF( J .LE. NW ) GO TO 20

      GO TO 10

      END
```

8 DMAP

While the **BINARY** option is the preferred method when the file is to be used on the same computer, the **FORMATTED** option allows transfer of the file to another computer. The format for **FORMATTED** files in both the **FULL** and **SPARSE** storage modes, along with examples of their usage, are given in the following tables:

FORMATTED Output Format: FULL Storage Mode

Record	Contents						
1	NCOL (I8)	NROW (I8)	FORM (I8)	PREC (I8)	NAME(1) (A4)	NAME(2) (A4)	

The first record contains the same data as in the **BINARY** mode. This time, however, it is read with a FORTRAN formatted read statement such as:

```
      READ(IUNIT,100)   NCOL,NROW,IFORM,ITYPE,NAME(1),NAME(2)
100   FORMAT(4I8,2A4)
```

2	COLID (I8)	IROW (I8)	NT (I8)				
3	V(1) (E16.9)	V(2) (E16.9)	V(3) (E16.9)	V(4) (E16.9)	V(5) (E16.9)		
4	V(6) (E16.9)	V(7) (E16.9)	V(8) (E16.9)	V(9) (E16.9)	V(10) (E16.9)		
n-1	...	...	...	...	...		
n	NCOL+1						

Each nonnull column now occupies several records as shown. First there is a record which contains the column number, **COLID**, the first nonzero row in the column, **IROW**, and the number of values, **NT**, in the column. This is followed by 80 character records each of which contains as many as five of the matrix values in the indicated format. The number of records used by a column, **NREC**, is then simply found by:

```
      NREC = NT / 5 + 1
```

Suppose that your matirx has a **PREC** of 1 indicating real, single precision data. Then one way to read a column would be:

```
      DIMENSION X(100)
      ...
      READ(IUNIT,100) COLID,IROW,NT
      READ(IUNIT,200) (X(I),I=1,NT)
100   FORMAT(3I8)
200   FORMAT(5E16.9)
```

Note that the accessing of this data does not depend on values being single or double precision. However, when processing complex matrices, the number of terms in the column is **NT/2** because each term requires 2 values.

FORMATTED Output Format: SPARSE Storage Mode

Record	Contents						
1	NCOL (I8)	NROW (I8)	FORM (I8)	PREC (I8)	NAME(1) (A4)	NAME(2) (A4)	
	The first record is the same as in the **FULL** format described above.						
2	COLID (I8)	0 (I8)	NW (I8)				
3	LS (I8)	IROW (I8)					
4	V(1) (E16.9)	V(2) (E16.9)	V(3) (E16.9)	V(4) (E16.9)	V(5) (E16.9)		
5	V(6) (E16.9)	V(7) (E16.9)	V(8) (E16.9)	V(9) (E16.9)	V(10) (E16.9)		
n-1	...	...	...	...	...		
n	NCOL+1						

Each non-null column is defined by several records as shown. First, there is a record which contains the column number, **COLID** and the total number of words of string data in the column, **NW**. Each string of non-zero values is defined by at least 2 records. The first record contains the first non-zero row number of the string, **IROW**, and the length of the string, **LS**. The string length includes the count of the 2 words used by **LS** and **IROW**. Following this record are as many records as required to output the current string values, at five values per record.

The following program will read a real, single precision matrix from this type of file:

```
      INTEGER IZ(100,NAME(2),FORM,PREC,COLID
      REAL Z(100,A(100,100)

      OPEN(1,FILE='filename',STATUS='OLD',FORM='FORMATTED')
      READ(1,'(4I8,2A4)') NCOL,NROW,FORM,PREC,NAME

10    READ(1,'(3I8)') COLID,NULL,NW
      IF( COLID .GT. NCOL ) STOP

20    READ(1,'(2I8)') LS,IROW
      NW =  NW - 2
      READ(1,'(5E16.9)') (Z(J),J=1,LS-2)
      J = 1
      DO 30 I = IROW, (IROW+LS-3)
            A(COLID,I) = Z(J)
            J = J + 1
            NW = NW - 1
30    CONTINUE
      IF( J .LE. NW ) GO TO 20

      GO TO 10

      END
```

8 DMAP

Module: PARAM

Performs specified arithmetic, logical and relational operations on integer parameters.

DMAP Instruction Syntax:

```
PARAM       // 'OP' / RESULT / PARM1 / PARM2 $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
OP	INPUT	PARAM	CHAR	Operation to be performed. [1]	Required
RESULT	OUTPUT	PARAM	INT	Result of the operation.	**1**
PARM1 PARM2	INPUT	PARAM	INT	Operands.	**1**

Remarks:

1. The operations that may be performed include arithmetic and logical parameter manipulation:

OP	Result	OP	Result
ADD	RESULT = PARM1+PARM2	**EQ**	If PARM1=PARM2, then RESULT=-1 else RESULT=1
AND	RESULT = PARM1.AND.PARM2	**GE**	If PARM1≥PARM2, then RESULT=-1 else RESULT=1
DIV	RESULT = PARM1/PARM2	**GT**	If PARM1>PARM2, then RESULT=-1 else RESULT=1
MAX	RESULT = MAX(PARM1,PARM2)	**LT**	If PARM1<PARM2, then RESULT=-1 else RESULT=1
MIN	RESULT = MIN(PARM1,PARM2)	**LE**	If PARM1≤PARM2, then RESULT=-1 else RESULT=1
MPY	RESULT = PARM1*PARM2	**NE**	If PARM1≠PARM2, then RESULT=-1 else RESULT=1
NOP	RESULT = RESULT	**IMPL**	If PARM1≥0 or PARM2<0, then RESULT=-1 else RESULT=1
NOT	RESULT = - PARM1		
OR	RESULT = PARM1.OR.PARM2		
SUB	RESULT = PARM1-PARM2		

and certain systems functions:

OP	RESULT
DIAGON	Enables **DIAG**s **PARM1** through **PARM2**
DIAGOFF	Disable **DIAG**s **PARM1** through **PARM2**
KLOC	**RESULT** = Number of seconds since midnight
PREC	**RESULT** = Machine precsion; 1 for 60 and 64 bit computers, 2 for 32 bit computers
TMTO	**RESULT** = Number of CPU seconds remaining for execution as specified by the Executive Control command **TIME**.

2. **PARAM** automatically **SAVE**s the resulting parameter.

Module: PARAML

Selects parameters from a user input matrix or table.

DMAP Instruction Syntax:

```
PARAML      INPUT // 'OP' / RECNO / WORDN / RPART / IPART
            INTVAL / 'CHAR' / SET1 / NSET1 / SET2 / NSET2 $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
INPUT	INPUT	ANY ENTITY	ANY	Entity name.	Required
OP	INPUT	PARAM	CHAR	Operation to be performed: **DMI** Extract from matrix entity. **DTI** Extract from table entity. **NULL** Test for a null matrix. **TRAILER** Extract a word from the entity trailer. **PRES** Test for existance of entity. **USET** Determine number of set members.	Required
RECNO	INPUT	PARAM	INT	Position within entity: If **DMI**, column number. If **DTI**, record number. If **TRAILER**, trailer word number (1-7).	**1**
WORDN	INPUT/ OUTPUT	PARAM	INT	Position within column or record: If **DMI**, row number. If **DTI**, word number. If **TRAILER**, the output value of the selected word.	**1**
RPART	OUTPUT	PARAM	RDP	Real parameter value extracted.	**0.0**
IPART	OUTPUT	PARAM	RDP	Imaginary part of a complex value extracted.	**0.0**
INTVAL	OUTPUT	PARAM	INT	Integer value extracted. If input is purged, set to -1. If **NULL**, -1 if input matrix is null. If **PRES**, +1 if input is not purged.	**0**
CHAR	OUTPUT	PARAM	CHAR	Character string value extracted.	Blank
SET1	INPUT	PARAM	CHAR	The name of a **USET** set to be checked for existance and number of members.	Blank
NSET1	OUTPUT	PARAM	INT	Number of degrees of freedom in **SET1**, if it exists, or **-1** if it does not exist.	**-1**
SET2	INPUT	PARAM	CHAR	The name of a **USET** set to be checked for existance and number of members.	Blank
NSET2	OUTPUT	PARAM	INT	Number of degrees of freedom in **SET2**, if it exists, or **-1** if it does not exist.	**-1**

Remarks:

1. **RPART**, **IPART**, **INTVAL**, and **CHAR** will be set whenever they are **V** type parameters.
2. **PARAML** does its own **SAVE**; therefore, a **SAVE** is not needed following the module.

Example:

1. Extract the value in column 1, row 1 of a matrix **KGG** and save it as parameter **TERM**:

```
PARAML   KGG // 'DMI' / 1 / 1 / TERM $
```

2. Extract the tenth word of the fourth record of table **BGPDT** and store the resultant real value as **BGOUT**:

```
PARAML   BGPDT // 'DTI' / 4 / 10 / BGOUT $
```

3. Determine the number of degrees of freedom in the *s-set*.

```
PARAML   USET//'USET'///////'S'/NUMSPC $
```

Module: PARAMR

Performs specified arithmetic and logical operations on real or complex parameters.

DMAP Instruction Syntax:

```
PARAMR      // 'OP' / RRES / R1 / R2 / CRES / C1 / C2 / F $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
OP	INPUT	PARAM	CHAR	Operation to be performed, see Remark 1.	Required
RRES	INPUT/ OUTPUT	PARAM	RSP	Result of any Real operation.	**0.0**
R1,R2	INPUT/ OUTPUT	PARAM	RSP	Real operands.	**0.0**
CRES	INPUT/ OUTPUT	PARAM	CSP	Result of any Complex operation.	**(0.0,0.0)**
C1,C2	INPUT	PARAM	CSP	Complex operands.	**(0.0,0.0)**
F	OUTPUT	PARAM	INT	True/False result for logical operations.	**0**

Remarks:

1. The available operators and their definitions are described in the table below:

OP	Result	OP	Result
ADD	**RRES = R1 + R2**	**NORM**	**RRES = |CRES|**
ADDC	**CRES = C1 + C2**	**COMPLEX**	**CRES = (R1,R2)**
SUB	**RRES = R1 - R2**	**CONJ**	**CRES = C1**
SUBC	**CRES = C1 - C2**	**REAL**	**R1 = Re(CRES)** and **R2 = Im(CRES)**
MPY	**RRES = R1 * R2**		
MPYC	**CRES = C1 * C2**	**EQ**	If **R1=R2,** then **F=-1** else **F=1**
DIV	**RRES = R1 / R2**	**GT**	If **R1>R2,** then **F=-1** else **F=1**
DIVC	**CRES = C1 / C2**	**LT**	If **R1<R2,** then **F=-1** else **F=1**
SIN	**RRES = sin(R1)**	**LE**	If **R1≤R2,** then **F=-1** else **F=1**
COS	**RRES = cos(R1)**	**GE**	If **R1≥R2,** then **F=-1** else **F=1**
ABS	**RRES = |R1|**	**NE**	If **R1≠R2,** then **F=-1** else **F=1**
EXP	**RRES = e^{R1}**	**LOG**	**RRES = log(R1)**
TAN	**RRES = tan(R1)**	**LN**	**RRES = ln(R1)**
POWER	**RRES = $R1^{R2}$**	**FIX**	**F = FIX(RRES)**
FLOAT	**RRES = FLOAT(F)**	**NOP**	**RRES = RRES**
SQRT	**RRES = SQRT(R1)**	**CSQRT**	**CRES = CSQRT(C1)**

2. If division by zero is requested, the result is set to zero.
3. For the trigonometric operators **SIN**, **COS**, and **TAN**, the input must be expressed in radians.

Module: PARTN

Partitions a matrix based on independent row and column partitioning vectors:

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$$

DMAP Instruction Syntax:

```
PARTN        A,CP,RP / A11,A21,A12,A22 / SYM $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	MATRIX	ANY	Name of the matrix to be partitioned. [2]	Required
CP	INPUT	MATRIX	RSP	Column partitioning vector. [3]	[4]
RP	INPUT	MATRIX	RSP	Row partitioning vector. [3]	[4]
Aij	OUTPUT	MATRIX	Same as A	Resulting matrix partitions.	At least one required [2]
SYM	INPUT	PARAM	INT	Partitoning option: <**0** A symmetric partitioning is done using **CP** for both vectors ≥**0** An unsymmetric partitioning is performed using both **CP** and **RP**	**-1**

Remarks:

1. The sizes of the matrix partitions are:

Matrix Partition	Number of Rows	Number of Columns
A11	Number of **0**'s in **RP**	Number of **0**'s in **CP**
A12	Number of **0**'s in **RP**	Number of **1**'s in **CP**
A21	Number of **1**'s in **RP**	Number of **0**'s in **CP**
A22	Number of **1**'s in **RP**	Number of **1**'s in **CP**
A	Number of rows in **RP**	Number of rows in **CP**

2. If a given partition is not wanted, its position in the input list should be left blank.
3. A partitioning vector is simply a column of **0**s, **1**s and **2**s. A **0** indicates that the given row, or column, is a member of the partition, and a **1** indicates it is a member of the second partition. A value of **2** indicates that this position in the partitioning vector is to be ignored. This vector must be the same size as matrix **A.**
4. If **RP** is given and **CP** is not, and **SYM**≥**0**, then a row partitioning will occur as shown in the examples. If **CP** is given and **RP** is not, and **SYM**≥**0**, then a column partitioning will occur as shown in the examples. If **SYM<0**, a symmetric partitioning is done, that is **CP** is used for both the column and row partitioning vectors. One of the vectors **CP** or **RP** must be given.

Examples:

The following examples use the following three matrices **A**, **CP** and **RP**:

$$\boldsymbol{A} = \begin{bmatrix} 1.0 & 2.0 & 3.0 & 4.0 \\ 5.0 & 6.0 & 7.0 & 8.0 \\ 9.0 & 10.0 & 11.0 & 12.0 \end{bmatrix}$$

$$\boldsymbol{CP} = \begin{Bmatrix} 1.0 \\ 0.0 \\ 1.0 \\ 1.0 \end{Bmatrix} \qquad \boldsymbol{RP} = \begin{Bmatrix} 0.0 \\ 0.0 \\ 1.0 \end{Bmatrix}$$

1. Perform a four-way partitioning of **A** using both partitioning vectors **CP** and **RP**:

   ```
   PARTN    A,CP,RP/A11,A21,A12,A22/1 $
   ```

   which results in:

   $$\boldsymbol{A_{11}} = \begin{Bmatrix} 2.0 \\ 6.0 \end{Bmatrix} \qquad \boldsymbol{A_{12}} = \begin{bmatrix} 1.0 & 3.0 & 4.0 \\ 5.0 & 7.0 & 8.0 \end{bmatrix}$$

   $$\boldsymbol{A_{21}} = \{ 10.0 \} \qquad \boldsymbol{A_{22}} = [\ 9.0 \ \ 11.0 \ \ 12.0\ ]$$

2. If, in example 1, the column partitioning vector is omitted,

   ```
   PARTN    A,,RP/A11,A21,,/1 $
   ```

   then the results are:

   $$\boldsymbol{A_{11}} = \begin{bmatrix} 1.0 & 2.0 & 3.0 & 4.0 \\ 5.0 & 6.0 & 7.0 & 8.0 \end{bmatrix}$$

   $$\boldsymbol{A_{21}} = [\ 9.0 \ \ 10.0 \ \ 11.0 \ \ 12.0\ ]$$

   and the partitons $\boldsymbol{A}_{12}$ and $\boldsymbol{A}_{22}$ are purged.

3. Similarly, if the row partitioning vector is omitted:

   ```
   PARTN    A,CP,/A11,,A12,/1 $
   ```

   then the results are:

   $$\boldsymbol{A_{11}} = \begin{Bmatrix} 2.0 \\ 6.0 \\ 10.0 \end{Bmatrix} \qquad \boldsymbol{A_{12}} = \begin{bmatrix} 1.0 & 3.0 & 4.0 \\ 5.0 & 7.0 & 8.0 \\ 9.0 & 11.0 & 12.0 \end{bmatrix}$$

   and the partitions $\boldsymbol{A}_{21}$ and $\boldsymbol{A}_{22}$ are purged.

Module: POST

Converts **UAI/*NASTRAN*** model information and analysis results to a form compatible with commercially available post-processors.

DMAP Instruction Syntax:

```
POST  IN1,IN2,IN3,IN4,IN5 / OUT1,OUT2,OUT3,OUT4,OUT5,OUT6 // $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
INi	INPUT	TABLE	N/A	Table entity names. [1]	At least one required
OUTi	OUTPUT	TABLE	N/AI	New table entity names corresponding to **INi**. [2]	At least one required

Remarks:

1. The following table lists all entities which are processed by module **POST**.

CSTM	**GEOM4**	**OES1**	**OES1L**	**OMESH1**
ECT	**MPT**	**OES1A**	**OES1NL**	**ONRGY1**
EPT	**OEF1**	**OES1AB**	**OES2**	**ONRGY2**
EQEXIN (HEAT)	**OEF1A**	**OES1AL**	**OES2A**	**USET**
GEOM2	**OEF1NL**	**OES1ANL**	**OMES1B**	
GEOM3	**OEF2**	**OES1B**	**OMES1AB**	

   The following table lists all table entities known to be compatible with post-processor programs and therefore require no processing by module **POST**. **All matrix entities are compatible and need not be processed.**

GEOM1	**LAMA**	**OESC1A**	**OPG2**	**OQG2**	**OUGV1**
GPDT	**OEFC1**	**OGPFB1**	**OPHIG**	**OQMG1**	**OUGV2**
GPL	**OESC1**	**OPG1**	**OQG1**	**OQMG2**	

   Any input entities other than those shown above do not require processing, and are copied to the output entities unchanged.

2. The output entity names must begin with the same first four characters of the input entity name for the proper recognition of the data by some commercial post-processor programs.

3. To use this module, you must **ALTER** the Rigid Format after the input entities have been created by **UAI/*NASTRAN*** modules.

4. The **OUTPUT2** module is normally used to write the output entities to a FORTRAN file.

5. If either of the entities of each pair **INi** and **OUTi** is purged, processing will continue on to the next pair.

Example:

1. Convert the output entities **CSTM**, **EPT** and **GEOM2** to the compatible format and write them to file **USER1** using the **OUTPUT2** module:

```
ALTER X
TRAILER  CSTM // 'RETURN' / 1 / S, N, PURGED $
COND     NOCSTM, PURGED $
POST     CSTM, , , ,/ CSTMOUT, , , ,$
OUTPUT2  CSTMOUT, , , ,/ / 0/ V, Y, OUTUNIT=USER1 / / / / 'CSTM' $
LABEL    NOCSTM $
POST     EPT, GEOM2, , , / EPTOUT, GEOM2OUT, , ,$
OUTPUT2  GPL, GPDT, EPTOUT, GEOM2OUT, / / 0/ V, Y, OUTUNIT ////
         'EPT' / 'GEOM2' / $
```

Note that the location of the **ALTER X** depends upon the specific Rigid Format. The use of parameters 6 through 10 of **OUTPUT2** is recommended since some post-processors check for exact entity name matches, thus you may not use the new name such as **CSTMOUT**.

Executive Instruction: PRECHK

Specifies a single, or limited number, of Checkpoint declarations without the need for a large number of individual **CHKPNT** instructions.

DMAP Instruction Syntax:

```
PRECHK      ALL $
PRECHK      END $
PRECHK      <DB_list> $
PRECHK      ALL EXCEPT <DB_list> $
            where  <DB_list> ⇒ DB1,DB2,...
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	INPUT	ANY ENTITY	ANY	Names of entities that will be automatically Checkpointed. [1]	Required

Remarks:

1. A maximum of 50 entity names may appear in a **PRECHK** instruction.
2. Any number of **PRECHK** declarations may appear in a DMAP program. Each time a new statement is encountered the previous one is invalidated. The **PRECHK END $** option will negate the current **PRECHK** status.
3. **CHKPNT** instructions may be used in conjunction with **PRECHK** declarations. The **CHKPNT** instruction will override any **PRECHK** condition. The example, if the **PRECHK ALL EXCEPT** option is in effect, an entity named in the excepted list may still be explicitly **CHKPNT**ed.
4. **PRECHK** automatically **CHKPNT**s all output entities from each functional module or **PURGE** instruction, and all secondary entities of an **EQUIV** instruction.

Module: PRTPARM

Prints a parameter value.

DMAP Instruction Syntax:

```
PRTPARM     // 0 / 'PNAME' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
PNAME	INPUT	PARAM	CHAR	Name of parameter to be printed	**XXXXXXXX**

Remarks:

1. The first parameter must always have the value zero.
2. If the second parameter is left blank, the default value will cause printing of all parameters as shown in Example 2 below.

Examples:

1. Print the current value of parameter **LUSET**:

```
PRTPARM  // 0 / 'LUSET' $
```

2. Print all of the DMAP parameters:

```
PRTPARM  // 0 $
```
or
```
PRTPARM  // 0 / 'XXXXXXXX' $
```

8 DMAP

Executive Instruction: PURGE

Indicates that an entity will not exist.

DMAP Instruction Syntax:

```
PURGE <list1> / PARM1 / <list2> / PARM2 ...$
      where <listi> ⇒ DB1,DB2,...
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	INPUT	ANY ENTITY	N/A	Name of entity to be purged.	Required
PARMi	INPUT	PARAM	INT	Parameter specifying the purge operation: <**0** The entities in <**listi**> are purged. ≥**0** The entities in <**listi**> are unpurged so that they are able to be reallocated.	**-1**

Remarks:

1. Any number of <**listi**> / **PARMi** groups may be specified in the instruction.

Examples:

1. Issue the instruction which purges entities **DB1**, **DB2** and **DB3** in all cases:

```
PURGE     DB1,DB2,DB3 $
```

2. Purge entities **A1**, **A2** and **A3** depending on the parameter **P1** and entities **B1** and **B2** depending on parameter **P2**:

```
PURGE     A1,A2,A3/P1/B1,B2/P2 $
```

8 DMAP

Executive Instruction: REPT

Loops on a group of DMAP instructions a specified number of times.

DMAP Instruction Syntax:

```
REPT  LAB,NLOOP $
REPT  LAB,PNAME $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
LAB	N/A	LABEL	N/A	The name appearing in a **LABEL** instruction which specifies the location of the beginning of a group of DMAP instructions to be repeated.	Required
NLOOP	N/A	CONST	INT	Constant which specifies the number of times to repeat the instructions.	Required
PNAME	N/A	PARAM	INT	Variable parameter set by a previously executed module specifying the number of times to repeat the instructions.	Required

Remarks:

1. **REPT** is placed at the end of the group of instructions to be repeated.
2. When a variable number of loops is to be performed, as in the second example below, the value of the variable the first time the **REPT** instruction is encountered will determine the number of loops. This number will not be changed after the initial assignment.
3. A **COND** instruction may be used to exit from the loop if desired.

Examples:

1. Write a DMAP sequence which executes modules **MODULE1** through **MODULEN** four times:

```
BEGIN $
...
LABEL    L1 $
MODULE1  A/B/P1 $
...
MODULEN  B/ /P2 $
REPT     L1,3 $
...
END $
```

8 DMAP

2. Next write a similar sequence as in Example 1 above except that the number of loops to be performed is set as an output of a module **MODX**:

```
BEGIN $
...
MODX     //NLOOP $
LABEL    L1 $
MODULE1  A/B/P1 $
...
MODULEN  B/ /P2 $
REPT     L1,NLOOP $
...
END $
```

Executive Instruction: SAVE

Specifies which variable parameter values are to be saved from the preceding functional module DMAP instruction for use by subsequent modules.

DMAP Instruction Syntax:

```
SAVE  V1,V2,...,VN $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
Vi	INPUT	PARAM	ANY	The names of variable parameters which appear in the immediately preceding DMAP statement.	At least one is Required

Remarks:

1. A **SAVE** instruction must immediately follow the functional module instruction generating the parameter to be saved.

Module: SCALAR

Extracts a specified element from a matrix so it may be used as a parameter.

DMAP Instruction Syntax:

```
SCALAR      A // NROW / NCOL / VALUE $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	MATRIX	ANY	Matrix entity name.	Required
ROWID	INPUT	PARAM	INT	Row number of the element to be extracted.	**1**
COLID	INPUT	PARAM	INT	Column number of the element to be extracted.	**1**
VALUE	OUTPUT	PARAM	CSP	Matrix element at **ROWID,COLID**	**(0.0,0.0)**

Remarks:

1. If **A** is purged, the value will be returned as **(0.0, 0.0)**.

Module: SETVAL

Sets, or assigns, a value to a DMAP parameter which is either another DMAP parameter variables or a constant.

DMAP Instruction Syntax:

```
SETVAL        //X1/A1/X2/A2/X3/A3/X4/A4/X5/A5 $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
Xi	OUTPUT	PARAM	ANY	The names of variable parameters.	At least one required
Ai	INPUT	PARAM	ANY	The names of variable parameters or constant values.	Required

Remarks:

1 A **SAVE** instruction must immediately follow the **SETVAL** instruction if the output parameter values are to be used later in the DMAP program.

2. See the **PARAM** instruction for an alternate method of defining parameter values.

Examples:

1. Set the value of parameter **MYPARAM** to **NOKGG**:

```
SETVAL    //MYPARAM/NOKGG $
```

2. Set the values of parameters **A**, **B**, **C**, and **D** to **0.0**:

```
SETVAL    //A/0.0/B/0.0/C/0.0/D/0.0 $
```

8 DMAP

Module: SMPYAD

Multiplies between two and five matrices and, optionally, adds another matrix to the product. Options allow the computation of:

$$X = \pm (A\,B\,C\,D\,E \pm F)$$

Any or all of the matrices ***A***, ***B***, ***C***, or ***D*** may be transposed.

DMAP Instruction Syntax:

```
SMPYAD      A,B,C,D,E,F / X / NUMBER / SIGNX / SIGNF / PREC /
            TRANA / TRANB / TRANC / TRAND $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	MATRIX	ANY	Name of the first product matrix.	Required
B	INPUT	MATRIX	ANY	Name of the second product matrix.	Required
C	INPUT	MATRIX	ANY	Name of the third product matrix.	Blank
D	INPUT	MATRIX	ANY	Name of the fourth product matrix.	Blank
E	INPUT	MATRIX	ANY	Name of the fifth product matrix.	Blank
F	INPUT	MATRIX	ANY	Name of the optional addend matrix.	Blank
X	OUTPUT	MATRIX	ANY	Name of the resultant matrix.	Required
NUMBER	INPUT	PARAM	INT	The number of product matrices.	Required
SIGNX	INPUT	PARAM	INT	The specification of the sign of the resultant matrix: **+1**: Positive product. **-1**: Negative product.	**1**
SIGNF	INPUT	PARAM	INT	The specification of the sign of the addend matrix: **+1**: Positive addend. **-1**: Negative addend.	**1**
PREC	INPUT	PARAM	INT	Numeric precision of the operation: **1**: Single precision. **2**: Double precision.	[2]
TRANA **TRANB** **TRANC** **TRAND**	INPUT	PARAM	INT	The transpose option for each of the first four product matrices: **0**: Do not use the transpose of the product matrix. **1**: Use the transpose of the product matrix.	**0**

Remarks:

1. Matrix **X** may not be purged.
2. The default **PREC**ision of operation is the maximum of the precisions of the input matrices and the precision of your **UAI/*NASTRAN*** host computer.

Examples:

1. Perform the operation $\boldsymbol{X} = \boldsymbol{AB} + \boldsymbol{F}$:

```
SMPYAD    A,B,,,,F/X/2 $
```

2. Perform the operation $\boldsymbol{X} = \boldsymbol{A}^T\boldsymbol{B} - \boldsymbol{F}$ and create a double precision resultant matrix:

```
SMPYAD    A,B,,,,F/X/2/+1/-1/2/1 $
```

Module: SOFI

Copies from one to five items from the SOF into **UAI/*NASTRAN*** entities.

DMAP Instruction Syntax:

```
SOFI  / OUT1, OUT2, OUT3, OUT4, OUT5 / DRY / 'NAME' / 'ITEM1' /
      'ITEM2' / 'ITEM3' / 'ITEM4' / 'ITEM5' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
OUTi	OUTPUT	ANY ENTITY	N/A	Entities corresponding to **ITEMi**. The referenced entities will be created for use in **UAI/*NASTRAN*** from the substructure items stored on the SOF.	[1]
DRY	INPUT	PARAM	INT	Substructure error value. [2]	Required
NAME	INPUT	PARAM	CHAR	Substructure name.	Required
ITEMi	INPUT	PARAM	CHAR	SOF data item names. [3]	Required

Remarks:

1. From one to five entities may be retrieved from the SOF with a single **SOFI** instruction.
2. The **DRY** parameter must have the value of **0** for module **SOFI** to execute.
3. Any existing SOF item may be retrieved. One item name must exist for each requested output entity.

Module: SOFO

Copies from one to five **UAI/*NASTRAN*** entities to the SOF.

DMAP Instruction Syntax:

```
SOFO  IN1, IN2, IN3, IN4, IN5 / DRY / 'NAME' / 'ITEM1' / 'ITEM2' /
      'ITEM3' / 'ITEM4' / 'ITEM5' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
INi	INPUT	ANY ENTITY	N/A	Entities corresponding to **ITEMi**.The referenced entities will be written to the SOF.	[1]
DRY	INPUT	PARAM	INT	Substructure error value [2].	Required
NAME	INPUT	PARAM	CHAR	Substructure name.	Required
ITEMi	INPUT	PARAM	CHAR	SOF data item names. [3]	Required

Remarks:

1. From one to five entities may be copied to the SOF with a single **SOFO** instruction.
2. The **DRY** parameter must have the value of **0** for module **SOFO** to execute.
3. Any entity may be copied to any SOF item. It is your responsibility to assure that entities have the correct format for the substructure item under which they are stored.

Module: SOFUT

Performs utility operations on the SOF.

DMAP Instruction Syntax:

```
SOFUT // DRY / 'NAME' / 'OPER' / OPT / 'NAME2' / 'PREF' / 'ITEM1' /
      'ITEM2' / 'ITEM3' / 'ITEM4' / 'ITEM5' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DRY	INPUT	PARAM	INT	Substructure error value.	Required [1]
NAME	INPUT	PARAM	CHAR	Substructure name.	Required
OPER	INPUT	PARAM	CHAR	Name of the operation to be performed.	Required [2]
OPT	INPUT	PARAM	INT	Option code.	**0**
NAME2	INPUT	PARAM	CHAR	Secondary substructure name.	[2]
PREF	INPUT	PARAM	CHAR	Prefix character.	[2]
ITEMi	INPUT	PARAM	CHAR	SOF data item names.	[2]

Remarks:

1. The **DRY** parameter must have the value of **0** for module **SOFUT** to execute.
2. The following table lists the available **OPER**ations along with the appropriate values of **OPT** in each case. The unused parameters in each case may be skipped.

OPER	OPT	Action Taken
DELE	Not Used	**ITEMi** of substructure **NAME** are deleted from the SOF.
DEST	Not Used	All items of substructure **NAME**, and any other structure of which **NAME** is a component, are deleted from the SOF.
EDIT	See Substructure Control command **EDIT**	Selected items of substructure **NAME** are deleted from the SOF.
EQUI	Not Used	Substructure **NAME2**, prefixed by **PREF**, is created and equivalenced to substructure **NAME**.
RENA	Not Used	Substructure **NAME** is renamed as **NAME2**.

OPER	OPT	Action Taken
SOFP	0	The SOF Table of Contents is printed.
	1	**ITEMi** of substructure **NAME** are printed. (Unformatted print for tables.)
	2	**ITEMi** of substructure **NAME** are printed. (Formatted print for tables.)
	-1	Action as taken with **OPT**ions 0 and 1.
	-2	Action as taken with **OPT**ions 0 and 2.

Examples:

1. Perform an **SOFPRINT(TOC)** operation with the following DMAP statement:

```
SOFUT    // 0 / / 'SOFP' / 0 $
```

2. Remove all solution data from substructure **ZZZ**:

```
SOFUT    // 0 / 'ZZZ' / 'EDIT' / 8 $
```

Module: SOLVE

Solves the set of simultaneous linear equations:

$$A X = \pm B$$

DMAP Instruction Syntax:

```
SOLVE A,B,USET,EQEXIN/ X / SYM / SIGN / PARM1 / PARM2 / 'SET' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	SQUARE MATRIX	ANY	Coefficient matrix entity name.	Required
B	INPUT	MATRIX	ANY	Right-hand-side matrix entity name.	[1]
USET	INPUT	TABLE	N/A	Name of the displacement set definition table.	[2]
EQEXIN	INPUT	TABLE	N/A	Name of the external to internal GRID point equivalence table.	[2]
X	OUTPUT	MATRIX	ANY	Solution matrix entity name. [3]	Required
SYM	INPUT	PARAM	INT	Symmetry characteristic of matrix **A**: **-1**: **A** is unsymmetric. **0**: Module determines symmetry. **+1**: **A** is symmetric.	**0**
SIGN	INPUT	PARAM	INT	Sign of **B** matrix: **+1**: Positive **-1**: Negative	**1**
PARM1	INPUT	PARAM	N/A	Reserved for future use.	[4]
PARM2	INPUT	PARAM	N/A	Reserved for future use.	[4]
SET	INPUT	PARAM	INT	The name of the displacement set corresponding to matrix **A**.	[2]

Remarks:

1. If **B** is not given, it is assumed to be an identity matrix and therefore, **X** will be the inverse of **A**.
2. **USET** and **EQEXIN** are optional and are used when singularities are found in the matrix being decomposed. The tables allow the module to translate the singular column numbers into more easily understood values referencing GRID points and degrees of freedom. The **SET** value is required if **USET** and **EQEXIN** are input.
3. Matix **X** may not be purged.
4. These parameters must be left blank.

Module: SWITCH

Interchanges two entity names.

DMAP Instruction Syntax:

```
SWITCH      DB1,DB2 // ENABLE $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DB1 **DB2**	INPUT	ANY ENTITIES	ANY	Names of entities to be switched.	Required
ENABLE	INPUT	PARAM	INT	Parameter enabling the operation. If negative, the **SWITCH** will be performed.	**-1**

Remarks:

1. This operation is of use in iterative DMAP programs.
2. All attributes of the entities remain the same, only the names are changed.

Module: TABPCH

Punches **UAI/*NASTRAN*** table entities onto **DTI** Bulk Data entries so that they may be used to transfer data from one run to another or for postprocessing.

DMAP Instruction Syntax:

```
TABPCH      TAB1,TAB2,TAB3,TAB4,TAB5 // 'A1' / 'A2' / 'A3' /
            'A4' / 'A5' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
TAB	INPUT	ANY TABLE ENTITIES	N/A	Table entities to be punched.	At least one required
Ai	INPUT	PARAM	CHAR	Continuation field prefix for the i^{th} table.	{**AA**, **AB**, **AC**, **AD**, **AE**}

Remarks:

1. Integers and characters are written as fixed-field entries. Real numbers are written as high-precision entries.
2. Up to 99,999 data entries may be written per table.
3. You must use the **ASSIGN** Executive Control command to define a logical file with the parameter **USE=BULK**.

Module: TABPRT

Prints selected table entities with formats for ease of reading.

DMAP Instruction Syntax:

```
TABPRT      TDB // 'TYPE' / OPT $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
TDB	INPUT	Table Entity	N/A	Table entity names to be printed. [1]	Required
TYPE	INPUT	PARAM	CHAR	Selects the table format. [2]	Blank
OPT	INPUT	PARAM	INT	Line spacing parameter: =**0** No blank lines between entries. ≠**0** One blank line will separate entries.	**0**

Remarks:

1. The table entities that may be printed are:

BGPDT	**EQEXIN**	**GPL**
CSTM	**GPCT**	**GPLG**
EQDYN	**GPDT**	**GPTT**

2. Parameter **TYPE** may be any of the entity names listed in the table above to override the otherwise automatic selection of the print format. Normally, this parameter should be left blank.
3. The alternate **TABPT** module can be used to print the contents of any entity.

Module: TABPT

Prints table and matrix entities.

DMAP Instruction Syntax:

```
TABPT        DB1,DB2,DB3,DB4,DB5 // $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
DBi	INPUT	ANY ENTITIES	ANY	Names of entities to be printed.	Required

Remarks:

1. The type, table or matrix, of each entity is determined automatically and its contents are formatted and printed accordingly.
2. The contents of the entity trailer are also printed.
3. Purged entities are not printed.

Module: TRAILER

Examines or modifies the trailer of an entity.

DMAP Instruction Syntax:

```
TRAILER     A // 'OPT' / WORD / VALUE $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	ANY ENTITIES	ANY	Name of entity for which the trailer is desired.	Required
OPT	INPUT	PARAM	CHAR	Option selector: **RETURN** The trailer value is returned. **STORE** The trailer value is changed. [1]	Required
WORD	INPUT	PARAM	INT	The trailer element on which to perform the requested **OPT**. **WORD** must be an integer between **1** and **6**.	Required [2]
VALUE	INPUT/ OUTPUT	PARAM	INT	The trailer value returned when **OPT=RETURN**, or the value to store if **OPT=STORE**.	Required [3]

Remarks:

1. The trailer of matrix entities **may not be modified**.
2. For table entities, the contents of the trailer vary. For matrix entities, the trailer positions contain:

Word	Contents
1	Number of columns.
2	Number of rows.
3	Matrix form. 1 square 5 upper triangular 2 rectangular 6 symmetric 3 diagonal 7 row vector 4 lower triangular 8 identity
4	Type of matrix elements. 1 real, single precision 2 real, double precision 3 complex, single precision 4 complex, double precision
5	Maximum number of non-zero words in any one column. This number is always the number of single precision words. For most matrices on 32 bit computers, the number of matrix terms is one half this number.
6	Matrix density. For example, a density of 80% is presented as **8000**.

3. If the entity is purged, the parameter value will be returned negative.

Examples:

1. Retrieve the first word of the trailer for entity **M1** and store it in parameter **COL**:

```
TRAILER  M1//'RETURN'/1/S,N,COL $
```

2. Store a value of 1 in the third word of the trailer for entity **T2**:

```
TRAILER  T2//'STORE'/3/1 $
```

Module: TRNSP

Transposes a matrix:

$$X = A^T$$

DMAP Instruction Syntax:

```
TRNSP A / X $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
A	INPUT	MATRIX	ANY	Name of matrix entity to be transposed.	Required
X	OUTPUT	MATRIX	ANY	Resultant transpose.	Required

Remarks:

1. **X** cannot be purged.

Module: UMERGE

Merges two matrices into a third based on the displacement sets:

$$\left\{ \begin{matrix} A_j \\ A_k \end{matrix} \right\} \Rightarrow A_i$$

DMAP Instruction Syntax:

```
UMERGE      USET,AJ,AK / AI / 'I' / 'J' / 'K' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
USET	INPUT	TABLE	ANY	Displacement set definition entity.	Required
AJ,AK	INPUT	MATRIX	ANY	Matrices to be merged.	Required
AI	OUTPUT	MATRIX	[1]	Matrix to be created.	Required
I	INPUT	PARAM	CHAR	The name of the displacement set to be partitioned or **ALL**. [2,3]	Required
J	INPUT	PARAM	CHAR	The upper partition subset of **I**, or **COMP**. [2,3]	Required
K	INPUT	PARAM	CHAR	The lower partition subset of **I**, or **COMP**. [2,3]	Required

Remarks:

1. The type of **AI** is the maximum of the types of **AJ** and **AK**. When used, the sizes of the partitions must conform with the specified partitioning sets.
2. The partitioning sets must obey the following rules:

   $\mathbf{J} \cap \mathbf{K} = \varnothing$; $\mathbf{J} \cup \mathbf{K} \subseteq \mathbf{I}$

   That is, **J** and **K** must must be mutually exclusive subsets of **I** whose union is all of, or a subset of, **I**.

   Finally, you may use the character string **COMP** for **J** or **K** if you wish a given set and its complement. You may use the character string **ALL** for **I** to signify all degrees of freedom in **USET**. Consult the ***User's Guide*** for a complete definition of the sets.

   The resulting matrix is of dimension **I** X **I**.

3. The most commonly used set combinations are presented below:

I Set Code	J and K Set Codes	I Set Code	J and K Set Codes	I Set Code	J and K Set Codes
A	L and R	FE	F and E	NE	N and E
D	E and A	G	M and N	P	E and G
F	O and A	N	S and F	S	SB and SG

User-defined sets **U1-U8** are also available. You may assign individual degrees of freedom to these sets with the **USET** and **USET1** Bulk Data entries. Also note that you may redefine the user set names with the **DEFUSET** Bulk Data entry.

Example:

1. Merge partitions of mode shapes, φ_a and φ_o, into the matrix φ_f:

```
UMERGE   USET,PHIA,PHIO/PHIF/''F''A''O' $
```

Module: UMERGE1

Merges four matrices into a fifth based on the displacement sets:

$$\begin{bmatrix} A_{jj} & A_{jk} \\ A_{kj} & A_{kk} \end{bmatrix} \Rightarrow A_{ii}$$

DMAP Instruction Syntax:

```
UMERGE1     USET,AJJ,AKJ,AJK,AKK / AII / 'I' / 'J' / 'K' / IOPT $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
USET	INPUT	TABLE	ANY	Displacement set definition entity.	Required
Ajk	OUTPUT	MATRIX	[1]	Matrices to be merged.	None [1]
AII	INPUT	MATRIX	ANY	Matrix to be created.	Required
I	INPUT	PARAM	CHAR	The name of the displacement set to be partitioned or **ALL**. [2,3]	Required
J	INPUT	PARAM	CHAR	The upper partition subset of **I**, or **COMP**. [2,3]	Required
K	INPUT	PARAM	CHAR	The lower partition subset of **I**, or **COMP**. [2,3]	Required
IOPT	INPUT	PARAM	INT	Selects a symmetric or vector partition.	**0** [4]

Remarks:

1. The type of **AII** is the maximum of the types of **Ajk**. Any or all of the **Ajk** partitions may be left blank if you do not need the resulting partition. When used, the sizes of the partitions must conform with the specified partitioning sets.
2. The partitioning sets must obey the following rules:

   $\mathbf{J} \cap \mathbf{K} = \varnothing$; $\mathbf{J} \cup \mathbf{K} \subseteq \mathbf{I}$

   That is, **J** and **K** must must be mutually exclusive subsets of **I** whose union is all of, or a subset of, **I**.

   Finally, you may use the character string **COMP** for **J** or **K** if you wish a given set and its complement. You may use the character string **ALL** for **I** to signify all degrees of freedom in **USET**. Consult the ***User's Guide*** for a complete definition of the sets.

   The resulting matrix is of dimension **I** X **I**.

3. The most commonly used set combinations are presented below:

I Set Code	J and K Set Codes	I Set Code	J and K Set Codes	I Set Code	J and K Set Codes
A	L and R	FE	F and E	NE	N and E
D	E and A	G	M and N	P	E and G
F	O and A	N	S and F	S	SB and SG

User-defined sets **U1-U8** are also available. You may assign individual degrees of freedom to these sets with the **USET** and **USET1** Bulk Data entries. Also note that you may redefine the user set names with the **DEFUSET** Bulk Data entry.

4. Parameter **IOPT** determines the type of partition to be performed as shown below:

IOPT=0	IOPT=1	IOPT=2
$A = \begin{bmatrix} A_{jj} & A_{jk} \\ A_{kj} & A_{kk} \end{bmatrix}$	$A = \begin{bmatrix} A_{jj} \\ A_{kj} \end{bmatrix}$	$A = \begin{bmatrix} A_{jj} & A_{jk} \end{bmatrix}$

8 DMAP

Module: UPARTN

Performs symmetric partitioning of a matrix:

$$A_{ii} => \begin{bmatrix} A_{jj} & A_{jk} \\ A_{kj} & A_{kk} \end{bmatrix}$$

DMAP Instruction Syntax:

```
UPARTN      USET,AII / AJJ,AKJ,AJK,AKK / 'I' / 'J' / 'K' / IOPT $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
USET	INPUT	TABLE	ANY	Displacement set definition entity.	Required
AII	INPUT	MATRIX	ANY	Matrix to be partitioned	Required
Ajk	OUTPUT	MATRIX	[1]	Partitions of **AII**.	None [1]
I	INPUT	PARAM	CHAR	The name of the displacement set to be partitioned or **ALL**. [2,3]	Required
J	INPUT	PARAM	CHAR	The upper partition subset of **I**, or **COMP**. [2,3]	Required
K	INPUT	PARAM	CHAR	The lower partition subset of **I**, or **COMP**. [2,3]	Required
IOPT	INPUT	PARAM	INT	Selects a symmetric or vector partition.	**0** [4]

Remarks:

1. The type of **Ajk** is same as the type of **AII**. Any or all of the **Ajk** partitions may be left blank if you do not need the resulting partition. The size of **AII** must be consistent with the size of set **I**.
2. The partitioning sets must obey the following rules:

   $\mathbf{J} \cap \mathbf{K} = \varnothing$; $\mathbf{J} \cup \mathbf{K} \subseteq \mathbf{I}$

   That is, **J** and **K** must must be mutually exclusive subsets of **I** whose union is all of, or a subset of, **I**.

   Finally, you may use the character string **COMP** for **J** or **K** if you wish a given set and its complement. You may use the character string **ALL** for **I** to signify all degrees of freedom in **USET**. Consult the ***User's Guide*** for a complete definition of the sets.

   The resulting matrix partitions are the sizes of the sets **J** and **K** as indicated.

3. The most commonly used set combinations are presented below:

I Set Code	J and K Set Codes	I Set Code	J and K Set Codes	I Set Code	J and K Set Codes
A	L and R	FE	F and E	NE	N and E
D	E and A	G	M and N	P	E and G
F	O and A	N	S and F	S	SB and SG

User-defined sets **U1-U8** are also available. You may assign individual degrees of freedom to these sets with the **USET** and **USET1** Bulk Data entries. Also note that you may redefine the user set names with the **DEFUSET** Bulk Data entry.

4. Parameter **IOPT** determines the type of partition to be performed as shown below:

IOPT=0	IOPT=1	IOPT=2
$A = \begin{bmatrix} A_{jj} & A_{jk} \\ A_{kj} & A_{kk} \end{bmatrix}$	$A = \begin{bmatrix} A_{jj} \\ A_{kj} \end{bmatrix}$	$A = \begin{bmatrix} A_{jj} & A_{jk} \end{bmatrix}$

Examples:

1. Partition the *n-size* stiffness matrix **KNN** symmetrically by the degrees of freedom in the *s-set* and *f-set*:

```
UPARTN    USET,KNN/KFF,KSF,KFS,KSS/'N'/'F'/'S' $
```

2. Partition the load vector PG into PN and PM.

```
UPARTN    USET,PG/PN,PM,,/'G'/'N'/'M'/1 $
```

Module: VEC

Creates a partitioning vector using the **UAI/*NASTRAN*** displacement sets:

$$v_i \Rightarrow \left\{ \begin{array}{c} v_j \\ v_k \end{array} \right\}$$

DMAP Instruction Syntax:

```
VEC   USET / VECTOR / 'I' / 'J' / 'K' $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
USET	INPUT	TABLE	ANY	Displacement set definition table entity.	Required [1]
VECTOR	OUTPUT	MATRIX	ANY	The resulting patitioning vector.	Required
I	INPUT	PARAM	CHAR	The name of the displacement set to be partitioned or **ALL**. [2,3]	Required
J	INPUT	PARAM	CHAR	The upper partition subset of **I**, or **COMP**. [2,3]	Required
K	INPUT	PARAM	CHAR	The lower partition subset of **I**, or **COMP**. [2,3]	Required

Remarks:

1. The entity **USET** depends on the Rigid Format and position of the **VEC** use. For Rigid Formats 1 through 5, and for Rigid Formats 7-12, if used prior to module **GKAD**, **USET** is the correct name. If used after module **GKAD** in Rigid Formats 7-12, it is **USETD**.

2. The partitioning sets must obey the following rules:

   $\mathbf{J} \cap \mathbf{K} = \varnothing ; \qquad \mathbf{J} \cup \mathbf{K} \subseteq \mathbf{I}$

   That is, **J** and **K** must must be mutually exclusive subsets of **I** whose union is all of, or a subset of, **I**. The resulting vector has one term for each term in **I**. The values are **0** for degrees of freedom in **J**, and **1** for degrees of freedom in **K**. When the union of the two sets is less that **I**, then all terms in the output vector for these extra degrees of freedom have a value of **2**.

   Finally, you may use the character string **COMP** for **J** or **K** if you wish a given set and its complement. You may use the character string **ALL** for **I** to signify all degrees of freedom in **USET**. Consult the ***User's Guide*** for a complete definition of the sets.

3. The most commonly used set combinations are presented below:

I Set Code	J and K Set Codes	I Set Code	J and K Set Codes	I Set Code	J and K Set Codes
A	L and R	FE	F and E	NE	N and E
D	E and A	G	M and N	P	E and G
F	O and A	N	S and F	S	SB and SG

User-defined sets **U1-U8** are also available. You may assign individual degrees of freedom to these sets with the **USET** and **USET1** Bulk Data entries. Also note that you may redefine the user set names with the **DEFUSET** Bulk Data entry.

Example:

1. Partition **KFF** into its a-set and o-set components:

```
VEC       USET / V / 'F' / 'O' / 'A' $
PARTN     KFF,V / KOO,KAO,KOA,KAA $
```

Alternately, the **VEC** module in this example could be called as:

```
VEC       USET / V / 'F' / 'O' / 'COMP' $ or
VEC       USET / V / 'F' / 'COMP' / 'A' $
```

Module: VECPLOT

Performs various matrix generation options.

DMAP Instruction Syntax:

```
VECPLOT     MATIN,BGPDT,EQEXIN,CSTM,, / MATOUT / GID / CID / OPTION $
```

Arguments:

Name	*Mode*	*Argument Type*	*Data Type*	*Description*	*Default*
MATIN	INPUT	MATRIX	ANY	Input matrix.	Required
BGPDT	INPUT	TABLE	N/A	Basic GRID Point Definition Table	Required
EQEXIN	INPUT	TABLE	N/A	Internal to external GRID point identification table.	Required
CSTM	INPUT	TABLE	N/A	Coordinate system transformation table.	Required
MATOUT	OUTPUT	MATRIX	[1]	Output matrix depending on the **OPTION** selected.	Required
GID	INPUT	PARAM	INT	GRID point identification number. [2]	**0**
CID	INPUT	PARAM	INT	Coordinate system identification number. [3]	**0**
OPTION	INPUT	PARAM	INT	Selects option to be performed. [4]	Required

Remarks:

1. The type of **MATOUT** is real, with the precision depending on your **UAI*/NASTRAN*** host computer.
2. If **GID** is not specified, then the origin of the Basic Coordinate System is used.
3. If **CID** is not specified, then the Basic Coordinate System is used.

4. Three unrelated functions are provided. These are selected by parameter **OPTION** and perform the operations described in the table below:

OPTION	Description
1	Creates the matrix **MATOUT** from the input vector **MATIN** by transforming it to the coordinate system defined by **CID.**

3	Creates the matrix **MATOUT** from the input matrix **MATIN**, where **MATIN** has one row for each *g-set* degree of freedom, and is usually a matrix such as **UVEC**. This option updates **MATIN** to include the geometric location of the GRID points as follows:
	`T1 = T1 + BGPDT(T1)`
	`T2 = T2 + BGPDT(T2)`
	`T3 = T3 + BGPDT(T3)`
	`Ri = 0.0`
	and for each scalar point degree of freedom:
	`T1 = 0.0`

4	Creates matrix **MATOUT** as a 6 row by *g-set* column matrix representing structural motion with unit displacements at **GID**. **MATIN** and **CID** are not used with this option.
6	Creates matrix **MATOUT** as a *g-set* row by 6 column matrix representing structural motion with unit displacements of the *r-set*. You define the *r-set*, which may contain no more than six component degrees of freedom, with a **SUPORT** Bulk Data entry.