NASA Technical Memorandum 101500

INTEGRATED MULTIDISCIPLINARY ANALYSIS TOOL

IMAT
USER'S GUIDE
FOR THE VAX/VMS COMPUTER

Edited by Frances T. Meissner

December 1988

(NASA-TM-101500) IMAT (INTEGRATED MULTIDISCIPLINARY ANALYSIS TOOL) USER'S GUIDE FOR THE VAX/VMS COMPUTER (NASA) 231 p CSCL 20K N89-14475

Unclas G3/39 0187242

NASA
National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665-5225

PREFACE

The Integrated Multidisciplinary Analysis Tool (IMAT) is a computer software system for the VAX/VMS computer that has been developed at the Langley Research Center. IMAT provides researchers and analysts with an efficient capability to analyze satellite control systems influenced by structural dynamics. Using a menu-driven executive system, IMAT leads the user through the program options. IMAT links a relational database manager to commercial and in-house structural and controls analysis codes. This paper describes the IMAT software system and how to use it.

CONTRIBUTORS:

PROGRAMMING TEAM

HEAD	Frances Meissner
UNISYS	Walter Halsey
	Sonya Smith
	Cynthia Holston
	Troy Landers
PRC	Mareta Chambers
	Daniel Giesy
	George Tan
CSC	Valerie Corker

ENGINEERING TEAM

HEAD	Paul Cooper
NASA	Thomas Sutter
	John Young
PRC	Alan Stockwell
	Tae Lim
	Zeen Kim

TABLE OF CONTENTS

Section 1
INTRODUCTION
Software
Software Overview - - - - - - - - - - - - - - - - - - 1-1
List of Commercial Software Packages - - - - - - - 1-4
Solution Procedure
Overview - 1-7
Step-by-Step Outline - - - - - - - - - - - - - - - - 1-12
Detailed Description - - - - - - - - - - - - - - - - 1-14

Section 2
EXECUTIVE
Introduction - 2-1
IMAT Commands - - - - - - - - - - - - - - - - - - 2-1
IMAT EXECUTIVE - - - - - - - - - - - - - - - - - 2-2

Section 3
DATABASE MANAGEMENT
Database Overview - - - - - - - - - - - - - - - - - 3-1
Database and File Management Processors
CREATE DATABASE - - - - - - - - - - - - - - - - - 3-2
QUERY DATABASE - - - - - - - - - - - - - - - - - 3-4
UNLOAD DATABASE - - - - - - - - - - - - - - - - 3-11
EXECUTE RIM - - - - - - - - - - - - - - - - - - - 3-12

Section 4
GRAPHICS
Metafile Definition - - - - - - - - - - - - - - - - - 4-1
Graphics Processors
GENERATE METAFILE - - - - - - - - - - - - - - - - 4-2
DISPLAY METAFILE - - - - - - - - - - - - - - - - - 4-11
GENERATE PICSURE COMMAND FILE - - - - - - - 4-15
EXECUTE PICSURE - - - - - - - - - - - - - - - - - 4-28
HARDCOPY METAFILE (for LaRC use only) - - - - - 4-29

Section 5

FINITE ELEMENT

IMAT Interfaces to Finite Element Application Codes 5-1

MSC/NASTRAN Processors

LOAD BULK DATA - - - - - - - - - - - - - - - - - - 5-10

FORMAT BULK DATA - - - - - - - - - - - - - - - - 5-12

EXECUTE MSC/NASTRAN - - - - - - - - - - - - - - 5-15

LOAD ANALYSIS RESULTS - - - - - - - - - - - - - 5-17

RECOVER PHYSICAL OUTPUT - - - - - - - - - - - - 5-19

SUPERTAB Processors

EXECUTE SUPERTAB/GEOMOD - - - - - - - - - - - - 5-32

LOAD UNIVERSAL FILE - - - - - - - - - - - - - - 5-34

FORMAT UNIVERSAL FILE- - - - - - - - - - - - - - 5-37

Section 6

CONTROLS

Development of Dynamic Equations of Motion for Flexible Structures with Active Controls - - - - - - 6-1

Generation of Linear Systems Matrices - - - - - - - 6-7

Controls Processors

DEFINE CONTROL SYSTEM - - - - - - - - - - - - - 6-14

GENERATE LINEAR SYSTEMS MATRICES - - - - - 6-19

EXECUTE MATRIXX - - - - - - - - - - - - - - - - - - 6-21

RECORD CONTROL SIMULATION - - - - - - - - - - - 6-22

QUERY EIGENVECTOR FILE - - - - - - - - - - - - - 6-29

APPENDIX A: IMAT Database Definition - - - - - - - - - - A-1

APPENDIX B: IMAT Example Problem - - - - - - - - - - - B-1

APPENDIX C: Raster Metafile Translator - - - - - - - - - - C-1

SECTION 1

INTRODUCTION

Software

Overview - 1- 1
List of Commercial Software Packages - - - - - - - - 1- 4

Solution Procedure

Overview - 1- 7
Step-by-Step Outline - - - - - - - - - - - - - - - - - - 1-12
Detailed Description - - - - - - - - - - - - - - - - - - 1-14

Software Overview 1

The Integrated Multidisciplinary Analysis Tool (IMAT) is a computer software system consisting of modular processors designed to transfer information to and from various structures and controls software packages via a central database management system. In general, each processor either takes information from the database and formats it for use in the user software package or converts the data obtained from the software package into a form to be accepted by the database. In this manner, information may be shared by researchers using different software packages. New application software can be attached to IMAT by writing a processor to format information from the database into the new application and by writing another processor to put the results into the database. Figure 1 is a graphic representation of the IMAT system of processors.

The IMAT EXECUTIVE is an interactive menu-driven system which leads the user through the program options. The IMAT EXECUTIVE resides on a Digital Equipment Corporation VAX with the VMS operating system. IMAT features include:

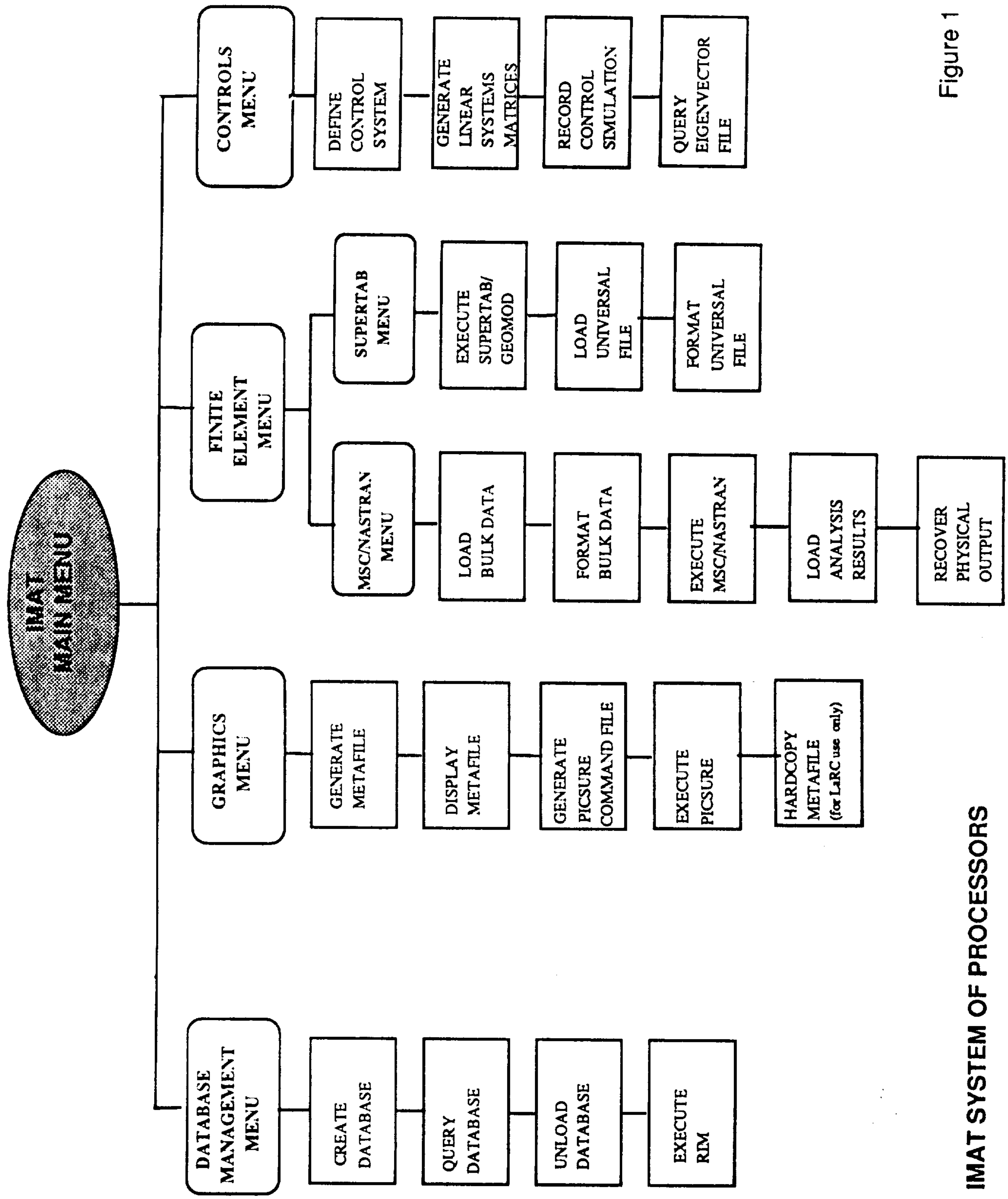
1. A database management system. Relational Information Management System (RIM) by Boeing Computer Services is the database manager. An important feature of RIM databases is portability. A RIM database may be transferred from the VAX to a variety of computer systems. The researcher need not be familiar with the RIM commands to use IMAT.

2. Maintained IMAT database schema. Each database has a consistent schema defined by the database dictionary. This ensures that all databases created by the IMAT software may be used by all of the IMAT processors and shared by IMAT users.

3. Utility processors for database creation, query, and unload. The user does not need to know any database commands to create or interrogate the database.

4. Access to software applications packages. Commercial as well as in-house code is available through the IMAT EXECUTIVE. IMAT has computer-aided design code, finite element generators, structural analysis code, controls analysis code, and plotting packages. The commercial code includes MSC/NASTRAN, I-DEAS (GEOMOD and SUPERTAB), and MATRIXx/SYSTEM_BUILD. The plotting software includes DI-3000 and PicSure. Controls software, developed in-house, is available which generates linear systems matrices used to model rigid body and flexible motion of a structure with feedback control.

5. Plotting capabilities. DI-3000 metafiles can be generated from I-DEAS, MATRIXx or MSC/NASTRAN plot files. PicSure can be executed either interactively or from a command file generated to plot analysis data from MATRIXx or MSC/NASTRAN.



IMAT SYSTEM OF PROCESSORS

Figure 1

DI-3000 and PicSure, developed by Precision Visuals, Incorporated, are versatile graphics software tools. DI-3000 provides a library of over 200 FORTRAN-callable two- and three-dimensional graphics subroutines. DI-3000 is device and machine independent. Tailored interfaces for over 70 graphics devices allow application programs that are written for a "virtual graphics device" to automatically utilize the full features of any supported display terminal. DI-3000's full color features include three-dimensional viewing, window clipping and depth clipping, and graphics quality text.

PicSure is a graphics system for producing professional quality charts and graphs. PicSure is also machine and device independent.

REFERENCES:

DI-3000 USER'S GUIDE, March 1986, Precision Visuals, Inc.

PicSure USER'S GUIDE, October 1986, Precision Visuals, Inc.

I-DEAS GEOMOD, Structural Dynamics Research Corporation's interactive graphics solid geometry modeling program, can be used to create complex solid objects. Two or more objects can be combined using simple Boolean operations. Physical properties of completed objects can be easily obtained, and finished geometry can be transferred to finite element preprocessors for mesh generation. A rigid body mechanism analysis is also available.

REFERENCE:

I-DEAS GEOMOD SOLID MODELING AND DESIGN REFERENCE MANUAL, LEVEL 3, 1986, Structural Dynamics Research Corporation

MATRIXx, a product of Integrated Systems Incorporated, is a programmable set of matrix subroutines that performs design and analysis functions for either classical or modern control problems. SYSTEM_BUILD, an optional capability of MATRIXx, provides an interactive graphics tool for building, modeling and editing computer simulation models. Complex linear or non-linear systems can be assembled from basic building blocks. Simulation can be performed using a variety of integration algorithms and results can be displayed graphically.

REFERENCE:

MATRIXx USER'S GUIDE, VERSION 6, 1986, Integrated Systems Inc.

MSC/NASTRAN is a version of the general purpose structural analysis program that has been developed and is maintained by the MacNeal-Schwendler Corporation. The MSC/NASTRAN code has an open architecture that allows researchers to easily modify structured solution sequences to solve non-standard static and dynamic structural analysis problems using the finite element method.

REFERENCE:

MSC/NASTRAN USER'S MANUAL, Version 65, 1985,
The MacNeal-Schwendler Corporation

Relational Information Management System (RIM) is a database management system developed by Boeing Computer Services. RIM is based on the relational algebra model for data management. Data grouped in tables called relations can be manipulated using relational algebra commands. RIM features a built-in interface to FORTRAN, COBOL, or PASCAL application programs, an English-like command language for data qualification, and facilities for the transfer of data between different machines.

REFERENCE:

RIM - RELATIONAL INFORMATION MANAGEMENT SYSTEM USER'S MANUAL, 1985, The Boeing Company

List of Commercial Software Packages 3

I-DEAS SUPERTAB, a product of Structural Dynamics Research Corporation, is an interactive graphics pre and postprocessor that can be used on a variety of graphics terminals to create finite element models. SUPERTAB contains translators that can create data files for input to commercial finite element codes such as MSC/NASTRAN. Analysis results can be displayed in various forms, including deformed structure plots, color stress contours, and X, Y plots.

REFERENCE:

I-DEAS SUPERTAB USER'S MANUAL, LEVEL 3, 1986, Structural Dynamics Research Corporation

The approach to the analysis of the dynamic behavior of structures with closed-loop control forces is represented schematically in figure 2. Processors are provided by IMAT and accessed through an interactive menu-driven executive to transfer data between applications through a relational database. The IMAT database schema was developed specifically for storage and retrieval of finite element models, modes, frequencies, material properties, rigid-body inertia information, and other engineering information of interest in the definition of large space-based structures. For this reason, IMAT does not support a complete set of the elements and solution sequences in MSC/NASTRAN. For example, neither solid elements nor aeroelastic analysis is supported. The database for a given structure is managed by a commercial relational database manager code, Relational Information Management System (RIM), from Boeing Computer Services.

Phase I. Structural Analysis

The analyst develops a finite element model (e.g., using the SUPERTAB modeling code from Structural Dynamics Research Corporation) and stores the model in an IMAT-defined database. The analyst can then retrieve finite element information from the database, create an MSC/NASTRAN data deck, and perform an undamped modal analysis using Solution 3 of MSC/NASTRAN. The analyst will usually compute a number of modes and frequencies sufficient to describe the dynamic behavior range of interest by a modal analysis. The analyst saves the New Problem Tape for future use in the recovery of physical data and stores the modes and frequencies in the IMAT database.

Phase II. Control Design and Analysis

With the normal modes available, the linear system matrices which define the structure (plant) are generated in a condensed form using IMAT's interactive GENERATE LINEAR SYSTEMS MATRICES processor. This processor asks the analyst to select the modes of interest which will be used to represent the dynamics of the plant, to define modal damping values for each mode, to define sensor locations and type, and to define controller locations and type.

The system dynamic equations of motion are reformulated in full matrix form using the MATRIXx and SYSTEM_BUILD codes developed and marketed by Integrated Systems Incorporated. In general, MATRIXx and SYSTEM_BUILD are used for linear systems and nonlinear systems, respectively. Once a system is formulated, a user can modify the system by adding more dynamics and control gains or deleting some parts of the system in the MATRIXx and SYSTEM_BUILD codes. MATRIXx can also be used to generate control gains and investigate the stability of a linearized system by using Bode plots and eigenvalue solvers.

Phase III. Simulation

If proportional damping is assumed, the dynamic equations for the structure can be represented by a set of uncoupled modal equations; however, the equations become coupled with the introduction of feedback control forces (forces which depend on response measurements) and the addition of dynamic equations for the actuators and sensors. The coupled system dynamic equations, not readily solvable in MSC/NASTRAN, are recast by the GENERATE LINEAR SYSTEMS MATRICES processor into first order differential equations based on state-space variables and solved directly in either the time domain or the frequency domain using MATRIXx. The output solutions could be responses of physical measurements or modal coordinates, and the input matrix representing the external forces could be generated using MATRIXx commands. For nonlinear systems, SYSTEM_BUILD can be used to solve the equations by choosing one of several available numerical integration algorithms based on the complexity of the dynamics and applied forces.

Phase IV. Data Recovery

After the controls analyst has completed the simulation using MATRIXx or SYSTEM_BUILD, the structural analyst uses MSC/NASTRAN to transform the modal solutions to physical output such as grid point displacements and element stresses. The physical data recovery procedure, illustrated in figure 3, requires three files - the Old Problem Tape from the Solution 3 normal modes analysis, the modal solution matrix calculated by MATRIXx, and an MSC/NASTRAN Solution 31 transient analysis restart data deck (including the checkpoint dictionary). The Old Problem Tape contains a complete physical and modal description of the structure, including physical and material properties, and the matrix of eigenvectors used to transform modal coordinates to physical coordinates. It does not require any modifications prior to being used in the restart process. An IMAT processor converts the modal solution matrix that was calculated by MATRIXx to MSC/NASTRAN INPUTT4 format and constructs the complete MSC/NASTRAN data deck to be used in the physical recovery operations. The last section of this overview provides details of these IMAT processors and their input/output requirements.

Integration of the controls and structures disciplines requires methods of organizing and transmitting data between analysts that reduce the probability of errors caused by poor communications. IMAT uses relations in an IMAT database to store the information that the structural analyst requires to calculate physical quantities based on modal simulation results. The controls analyst enters the information interactively, using an IMAT processor, by describing the system, including modal damping ratios, time steps and length of time used in the simulation, and the modes that were used. Controller and sensor descriptions loaded during the control system definition phase, as well as applied load descriptions, are also stored in the database. The updated database contains the names of the MATRIXx/SYSTEM_BUILD files that contain the modal solution matrices and the applied load and controller force/torque time histories.

The structural analyst uses an IMAT interactive processor to create the MSC/NASTRAN Solution 31 modal transient analysis restart data deck. In addition to user-supplied input, the processor retrieves data from the database and from files that are referenced in the database. The processor inserts a simple DMAP alter in the Executive Control Deck that forces MSC/NASTRAN to "jump" over the transient response calculations and read in the MATRIXx simulation results instead. The data recovery phase of Solution 31 then proceeds in the normal manner. The processor also customizes a DMAP alter that edits the eigenvalue table and eigenvector matrix to ensure that the modes used in the data recovery phase are the same solution modes that were used in the simulation. This feature allows the controls analyst to select any combination of modes for the simulation.

The Case Control Deck is assembled automatically except for the titles and the X, Y plot requests. The plot request package is created by a menu-driven procedure that interrogates the user to determine the information to be plotted and the format of the plots.

The Bulk Data Deck for the restart contains the cards needed to define the loads, the modal damping ratios, the simulation times and time steps, and the vectors that are used to partition the eigenvalues and eigenvectors. All of the Bulk Data cards are created automatically by the processor.

Utilities

The IMAT software includes a set of database management and graphics utilities that facilitate inspection of database contents and analysis results. Database management processors allow a user to easily create or interactively inspect the contents of a database without using RIM commands. A separate menu selection allows a user experienced in the use of RIM to execute RIM directly. In addition, a means to unload a database (i.e., write the contents to an ASCII file) is provided. This feature enables a user to transfer all or part of a database's contents to any other computer. If the other computer has RIM software installed, a new database can be easily created by reading the unload file into RIM.

IMAT provides a common interface for storing, retrieving, and displaying graphical information. The IMAT graphics system is based on Precision Visuals' device-independent DI-3000 software. IMAT's graphics processors generate DI-3000 metafiles (device-independent plot files) from the plot files created by SDCR's I-DEAS software (e.g., SUPERTAB), MSC/NASTRAN, and MATRIXx. The metafiles may be displayed on a monochrome or color graphics terminal by using the DISPLAY METAFILE processor. Publication-quality X, Y plots such as time histories or Bode plots can be produced by using a processor that takes MSC/NASTRAN or MATRIXx data and generates a command file for use with Precision Visuals' PicSure plotting program.

Outline of Procedure

The following sections consist of a brief outline, followed by a detailed step-by-step description of the IMAT solution procedure. Appendix B uses a numerical example to illustrate this procedure.

Step-by-step Outline of the Solution Procedure

Phase I. Structural Analysis

Step 1. Build structural model with SUPERTAB.

Step 2 Run CREATE DATABASE processor to create database.

Step 3. Run LOAD UNIVERSAL FILE to load finite element model data into database.

Step 4. Run FORMAT BULK DATA to create MSC/NASTRAN data deck.

Step 5. Run MSC/NASTRAN to calculate natural frequencies and mode shapes. Save Old Problem Tape, Punch File, and utility file containing rigid-body mass properties (default name FOR011.DAT).

Step 6. Run LOAD ANALYSIS RESULTS to load frequencies, mode shapes, and rigid-body mass properties into database.

Phase II. Control Design and Analysis

Step 7. Use DEFINE CONTROL SYSTEM to store control system definition in database.

Step 8. Run GENERATE LINEAR SYSTEMS MATRICES processor to create system matrices.

Step 9. Run MATRIXx to calculate control gain matrix, assemble the plant matrix and store these matrices in a MATRIXx file.

Step 10. Assemble closed-loop plant (A) matrix in MATRIXx.

Step 11. Run MATRIXx to calculate closed-loop eigenvalues.

Phase III. Simulation

Step 12. Generate a system matrix including disturbance matrix [E] in MATRIXx. (Nonlinear systems should be formulated and solved using SYSTEM_BUILD.)

Step 13. Run MATRIXx to calculate transient response to the desired forcing function, e.g., an impulse load. Save modal solution ($q, \dot{q},$ and $\ddot{q}$) and actuator and applied-load force/time histories in formatted MATRIXx files.

Step 14. Use RECORD CONTROL SIMULATION to enter simulation information.

Phase IV. Data Recovery

Step 15. Run RECOVER PHYSICAL OUTPUT processor to create UHV (modal solution) file in MSC/NASTRAN INPUTT4 format and to create MSC/NASTRAN data deck (including plot requests).

Step 16. Run MSC/NASTRAN using Old Problem Tape from normal modes solution and UHV (modal solution) matrix calculated by MATRIXx to obtain physical results.

Detailed Description of the Solution Procedure

In order to solve the dynamic equations, three design techniques are applied - assume the structure may be represented as a linear elastic system, discretize the model by finite element techniques, and perform a modal analysis. The menu-driven IMAT process lends itself to a step-by-step breakdown from model development to controls and structures analysis. The individual steps are grouped in four phases - Structural Analysis, Control Design and Analysis, Simulation, and Data Recovery. Bold face type in the text refers to the interactive IMAT menu prompts. The numerical example presented in Appendix B illustrates an application of the procedure to a simple three-mass model with an active control system.

Phase I. Structural Analysis

Step 1. SUPERTAB model building

Begin the session by constructing the model in response to menu queries (the test case is a 3-mass structure as shown in the Numerical Example section, connected by damped springs). From the **SUPERTAB MENU**, choose **EXECUTE SUPERTAB/GEOMOD**, choose the terminal type (e.g., T4014), establish a model file, and pick the desired units of weight and measure (**NOTE**: IMAT allows any consistent set of units that has mass, length, and time as the primary quantities). Now select the SUPERTAB (**EA**) option, get into PRE/POSTPROCESSING (**P**), and create the physical and material tables to describe:

a. modulus of elasticity
b. Poisson's ratio
c. shear modulus
d. rod cross-sectional area (for spring constants, k)
e. mass data

At this point, set up the nodes and elements of the model and assign them the appropriate physical and material properties. (**NOTE**: In the three-mass example problem in Appendix B, rod elements are used in place of springs which are not in the IMAT library. Modal damping is used instead of physical viscous dampers.) Write a Universal file before exiting from SUPERTAB.

Step 2. CREATE DATABASE

This process creates an empty database formatted with the IMAT schema. Beginning with the **DATABASE MANAGEMENT MENU** of IMAT, select **CREATE DATABASE**. The procedure follows logically from the menu prompts. At the end of the process, there will be three files in the current directory - dbname1.DAT, dbname2.DAT, and dbname3.DAT, where "dbname" is the name of the IMAT database. These files are created with a DELETE protection that must be changed by the owner before they can be deleted.

Step 3. LOAD UNIVERSAL FILE

Execute the **LOAD UNIVERSAL FILE** processor and input the name of the database created in step 2. This procedure loads finite element model data contained in the universal file into the IMAT database. At this point the user should use the IMAT processor called **QUERY DATABASE** to inspect the database contents or to obtain a printout of the database.

Now the model is analyzed using MSC/NASTRAN (developed by the MacNeal-Schwendler Corporation) to compute its modes and frequencies. The IMAT Normal Modes Analysis Data Flow is illustrated in figure 4.

Step 4. FORMAT BULK DATA

First select the **MSC/NASTRAN MENU** from the **FINITE ELEMENT ANALYSIS MENU**. Using the **FORMAT BULK DATA** processor, the user provides a maximum frequency of interest and specifies the degrees of freedom (DOF) that are constrained on the GRDSET card (if there are less than six rigid-body degrees of freedom). An MSC/NASTRAN data file is then created containing a complete finite element description of the model. The data deck includes CASE and EXECUTIVE CONTROL decks for a Solution 3 Normal Modes analysis.

Step 5. EXECUTE MSC/NASTRAN

From the **MSC/NASTRAN MENU**, select **EXECUTE MSC/NASTRAN**. Enter the name of the MSC/NASTRAN data deck, and select from the **FILE OPTIONS MENU** the **SAVE RESULTING UTILITY FILES** option. This feature allows the user to assign a name to the utility file associated with FORTRAN unit 11 (default name is FOR011.DAT). The utility file will contain binary output from MSC/NASTRAN's Grid Point Weight Generator. This is the rigid-body mass-property data that is loaded into the RIGPROP relation in the IMAT database (see step 6). Eigenvalues, eigenvectors and the Checkpoint Dictionary (used later for restart) are saved in the PUNCH file called "name.PCH", where "name" is the name of the MSC/NASTRAN data deck.

Step 6. LOAD ANALYSIS RESULTS

This procedure takes MSC/NASTRAN analysis data from the PUNCH file and utility file and loads it into the database created in step 2. Choose **FINITE ELEMENT ANALYSIS** from the IMAT menu followed by **MSC/NASTRAN MENU** and **LOAD ANALYSIS RESULTS**. The output data is added to the database - that is, vibrational eigenvector and frequency data (contained in the punch file) and rigid-body mass properties (contained in the utility file). The analyst should now use **QUERY DATABASE** (under the **DATABASE MANAGEMENT MENU**) to carefully inspect the database contents, especially the EIGNVALS, EIGNVECT and RIGPROP relations. An additional file (default name "dbname.EGN") is created. This is a rapid-access binary file that contains the eigenvector data. It is used by the **GENERATE LINEAR SYSTEMS MATRICES** (see step 8) and the **QUERY EIGENVECTOR FILE** processors.

Phase II. Control Design and Analysis

Step 7. DEFINE CONTROL SYSTEM

From the **CONTROLS MENU**, select **DEFINE CONTROL SYSTEM** and enter the actuator, sensor and applied force (disturbance) information for the model. Additionally, the user must specify which modes are to be used, the damping coefficients for each mode, and the system description for the control model -- this is essential in the next step.

Step 8. GENERATE LINEAR SYSTEMS MATRICES

From the **CONTROLS MENU**, select **GENERATE LINEAR SYSTEMS MATRICES** and enter the name of the controls database, identifying the particular system that will be used in the analysis. From this data the A, B, C, and E matrices are generated. The plant matrix [A] and the control influence matrix [B] specify the linear system completely so that the dynamics and stability of the model can be determined by investigating these matrices. The observation matrix [C] contains information relating physical output to the state vector. The disturbance matrix [E] can also be created for use in the simulation phase. The matrices are written in compressed form to a formatted file that can be read by MATRIXx. A DATA PRINT FILE can also be generated so that the user can visually inspect the matrices.

Step 9. Run MATRIXx to calculate control gain matrix

Run **EXECUTE MATRIXX** to load the linear system matrix file and set up the plant matrix. MATRIXx can also be used to compute control gains via an appropriate control law. Modern control theory (time domain) is used in the example problem in Appendix B to establish a control law that minimizes a specified performance index. Store the newly created control gain matrix and plant matrix in a MATRIXx file.

Step 10. Create the closed-loop plant (A) matrix in MATRIXx

Assemble the control gain and plant matrices and set up the system for closed-loop analysis without accounting for the disturbance forces.

Step 11 Compute closed-loop eigenvalues in MATRIXx

Now that the closed-loop [A] matrix is created, run MATRIXx and compute the closed-loop system eigenvalues to determine the stability of the system. System stability can also be investigated by using MATRIXx to generate information such as Bode plots. Store this information in a MATRIXx file. If the system is stable, move on to the simulation phase.

Phase III. Simulation

Step 12. Generate system matrix, including MATRIXx

Now that the matrices have been assembled in MATRIXx, the user can perform a simulation for a given input function, such as a unit impulse load. The closed-loop system now must include the E matrix. (**NOTE:** SYSTEM_BUILD, found within MATRIXx, is a valuable tool for building block diagrams of feedback control systems. The procedure will not be shown here, but it is available in the IMAT process.)

Step 13. Calculate transient response to impulse load

Compute the response of the system to a disturbance by performing a simulation in MATRIXx. Use MATRIXx to write three separate formatted files (i.e., use FSAVE) for modal displacement, velocity and acceleration. All files should be written such that each **column** of the vector or matrix represents a time step and each **row** represents a structural mode. A separate file must also be generated for each controller force and disturbance force.

Step 14. RECORD CONTROL SIMULATION

Select the **CONTROLS MENU** and execute the **RECORD CONTROL SIMULATION** processor to store the information pertaining to the simulation. This information includes the names of the files generated in the previous step. The **RECOVER PHYSICAL OUTPUT** processor uses this data to create a complete MSC/NASTRAN modified-restart data deck.

Phase IV. Data Recovery

Step 15. RECOVER PHYSICAL OUTPUT

Enter the **CONTROLS MENU** of IMAT and select the **RECOVER PHYSICAL OUTPUT** option. This processor reads the modal (state-space) solution calculated by MATRIXx and converts it to a form that can be read by MSC/NASTRAN. The **RECOVER PHYSICAL OUTPUT** processor also creates a complete MSC/NASTRAN data deck that is used to calculate desired physical output quantities such as nodal displacements or element stresses. The processor allows the user to interactively select the information to be plotted and generates appropriate XYPLOT commands.

Step 16. Run modified MCS/NASTRAN Solution 31 to get physical output

Run MSC/NASTRAN using the Old Problem Tape from the normal modes solution, the UHV (modal solution) matrix file created in step 15, and the MSC/NASTRAN data deck created in step 15. The UHV matrix file must be assigned to FORTRAN unit 15 prior to MSC/NASTRAN execution.

Description of Data Recovery Processors

Two processors are employed in using MSC/NASTRAN to recover physical output from a state-space time simulation calculated using MATRIXx or SYSTEM_BUILD as shown in figure 3. A description of each processor follows.

RECORD CONTROL SIMULATION: This interactive program allows the user to enter a record of one closed-loop simulation into the IMAT-defined database which contains the structural and control data for the model. This information is necessary for the RECOVER PHYSICAL OUTPUT processor when generating an MSC/NASTRAN bulk data deck to obtain displacements or stresses. The data items requested by the program in order to create a record of a control simulation include the name of the analyst, date of the analysis, the simulation times, and related modal output and time history file names. The program will also allow the user to examine and make changes to previous control simulation records as needed.

RECOVER PHYSICAL OUTPUT: This program reads the MATRIXx or SYSTEM_BUILD modal simulation results and creates an MSC/NASTRAN UHV (modal solution) matrix. A binary file is used for the UHV matrix to conserve disk space due to the anticipated size of these files and also to take advantage of the significant performance advantage of using a binary structured file. The RECOVER PHYSICAL OUTPUT program also allows the structural dynamicist during an interactive menu-driven terminal session to create the MSC/NASTRAN Executive, Case Control, and Bulk Data Decks required to recover physical results from the closed-loop simulation. The program uses the information stored in the database by the controls analyst to locate the load time history files. It also extracts the modes, damping ratio, and time step interval used in the simulation from the database.

Creation of the Executive Control Deck: As a precaution, the RECOVER PHYSICAL OUTPUT processor compares the number of structural modes used in the MATRIXx/SYSTEM_BUILD simulation with the number of modes stored in the IMAT database, and alerts the analyst to any discrepancies. It then creates the Executive Control Deck which includes alters which will tailor the execution of MSC/NASTRAN to read in the state-space solution from MATRIXx/SYSTEM_BUILD files (See figure 5):

1. ALTER 455 is used to remove the structural modes not used by the controls analyst from the solution.

2. ALTERS 479 and 480 are used to skip MSC/NASTRAN's transient analysis calculations and substitute a UHV (modal solution) matrix that has been calculated by the controls code, MATRIXx or SYSTEM_BUILD.

The checkpoint dictionary is extracted from the punch file created by the original MSC/NASTRAN normal modes analysis and inserted into the Executive Control Deck.

Creation of the Case Control Deck: The RECOVER PHYSICAL OUTPUT processor initializes the Case Control Deck including title information entered by the structural dynamicist and allows the analyst to set plot parameters such as paper size, axis specifications, and scaling. The program then provides for automated generation of plot commands for dynamic load plots, modal and physical displacement, velocity, and acceleration plots, and element force and stress plots through a menu-driven interface.

Creation of the Bulk Data Deck: The RECOVER PHYSICAL OUTPUT processor completes the MSC/NASTRAN data deck by generating the TLOAD, DAREA, and TABLED1 cards required for each dynamic load. DMI cards are written to generate the partitioning vectors that are used to remove modes from the analysis.

The recovery of physical output for the closed-loop simulation is completed with the execution of MSC/NASTRAN using the data deck and the UHV matrix file created by the RECOVER PHYSICAL OUTPUT processor, as well as the Old Problem Tape (OPTP) from the normal modes analysis.

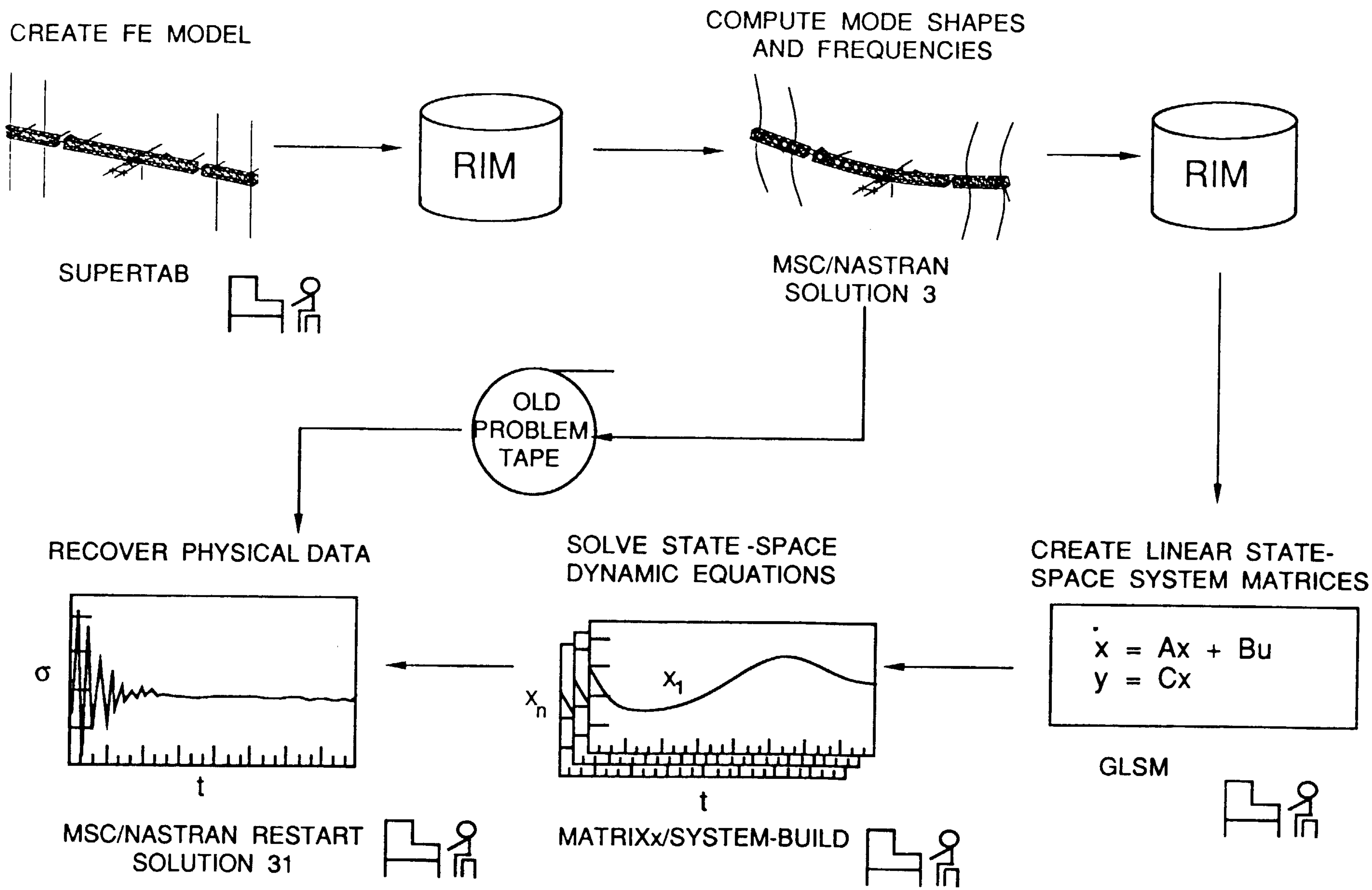


Figure 2 IMAT process for the simulation of actively controlled flexible structures

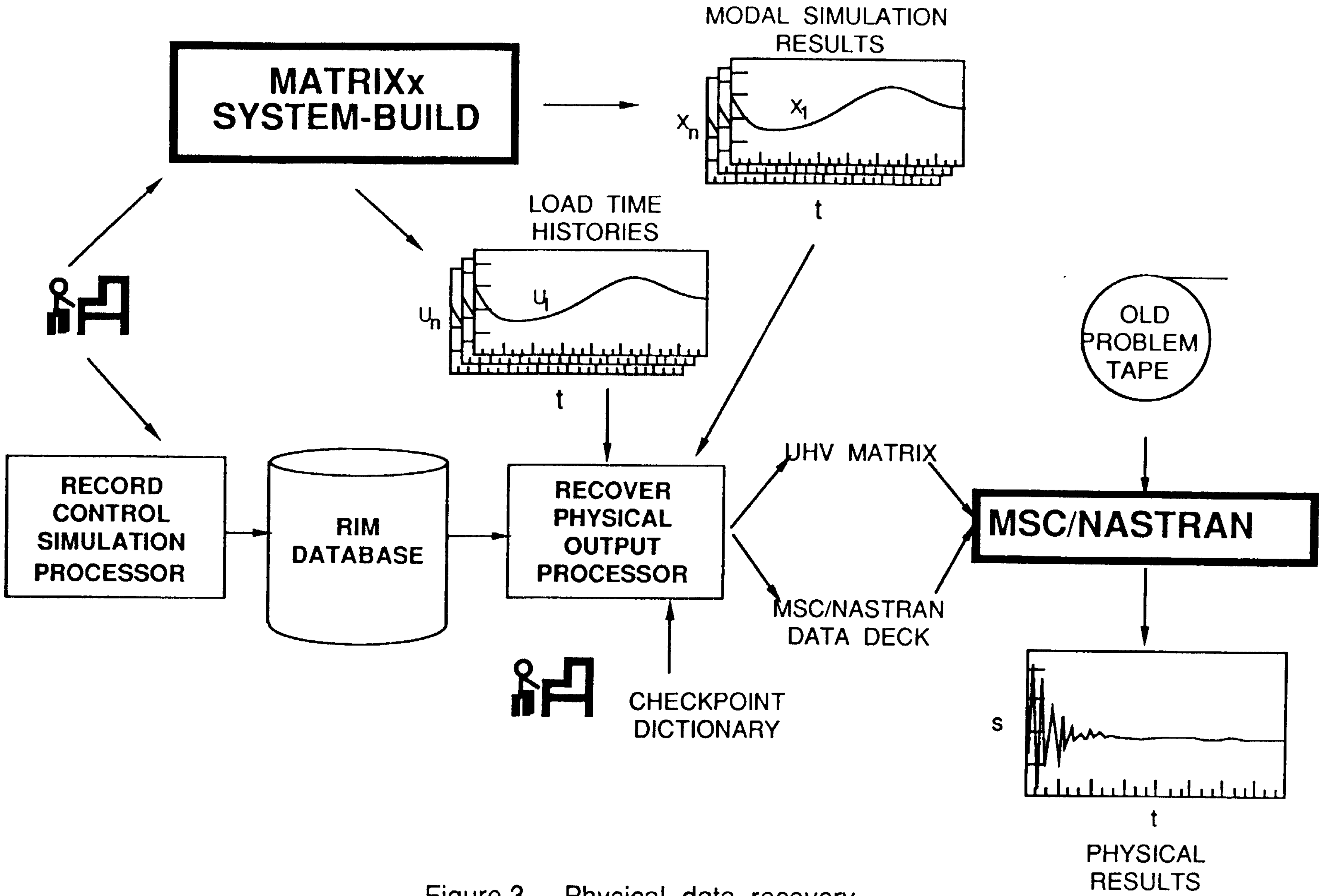
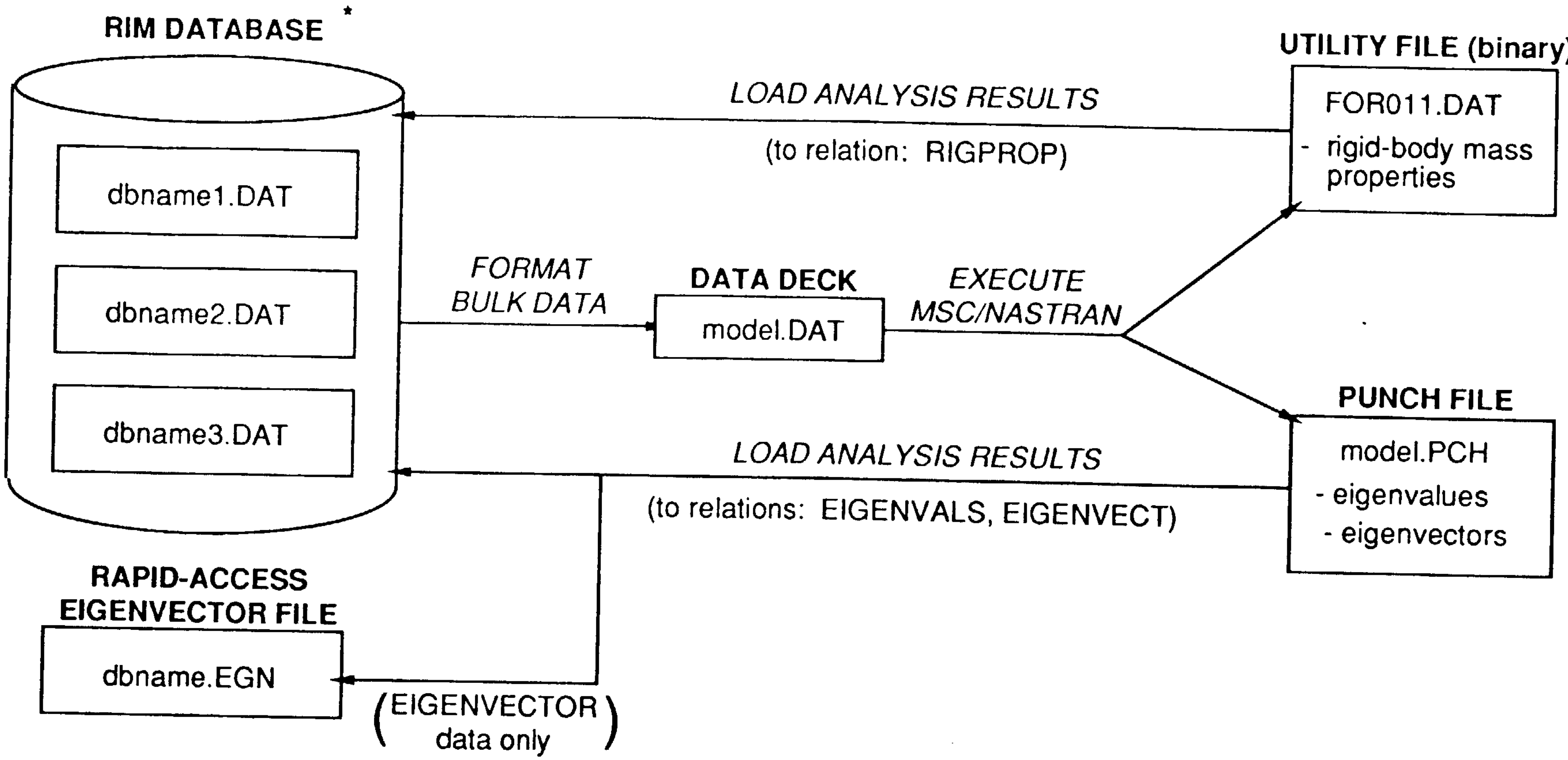


Figure 3 Physical data recovery procedure



*NOTE: THE THREE RIM DATABASE FILES AND THE RAPID-ACCESS EIGENVECTOR FILE ARE ALL BINARY FILES. DATA IN THESE FILES CAN ONLY BE ACCESSED BY IMAT/RIM SOFTWARE.

Figure 4 IMAT normal modes analysis data flow

```
ID TRANSIENT ANALYSIS, IMAT ALTERS
SOL 31,0
COMPILER = LIST
$
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$                                                                               $
$   THE FOLLOWING ALTER IS USED TO SKIP MSC/NASTRAN'S TRANSIENT ANALYSIS        $
$ CALCULATIONS AND SUBSTITUTE A "UHV" (MODAL SOLUTION) MATRIX THAT              $
$ HAS BEEN CALCULATED BY A CONTROLS CODE (E.G.,MATRIXX).  THE                   $
$ PROCEDURE ALLOWS A CLOSED LOOP TRANSIENT ANALYSIS MODAL SOLUTION              $
$ TO BE READ INTO MSC/NASTRAN USING THE "INPUTT4" UTILITY MODULE.  THE          $
$ MSC/NASTRAN SOLUTION SEQUENCE IS THEN USED TO OBTAIN PHYSICAL OUTPUT          $
$ SUCH AS DISPLACEMENTS OR STRESSES FROM THE CLOSED LOOP MODAL                  $
$ SOLUTION                                                                      $
$    TWO  ADDITIONAL DATA FILES MUST BE USED IN THE EXECUTION OF THIS           $
$ MODIFIED SOLUTION SEQUENCE.                                                   $
$                                                                               $
$     1) THE "NEW PROBLEM TAPE" CONTAINING THE MODE SHAPES AND                  $
$        FREQUENCIES CALCULATED BY MSC/NASTRAN SOLUTION 3 MUST                  $
$        BE ASSIGNED AS "OLD PROBLEM TAPE" (OPTP) FOR THE RESTART.              $
$                                                                               $
$     2) THE CONTROLS-CODE-GENERATED MODAL SOLUTION MATRIX MUST                 $
$        BE ASSIGNED AS FORTRAN UNIT "FOR015" (FOR015 IS ASSUMED TO BE          $
$        A FORTRAN READABLE BINARY FILE.)                                       $
$                                                                               $
$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
$                                                                               $
$$$$$$$$ ALTER TO REMOVE USER SELECTED MODES FROM SOLUTION SET          $$$$$$$$
$                                                                               $
$        SUMMARY OF MODES USED IN CLOSED LOOP SIMULATION                        $
$                 (0 = UNUSED   1 = USED)                                       $
$               ORIGINAL MODES CALCULATED:   26                                 $
$                                                                               $
$                 1         2                                                   $
$        12345678901234567890123456                                             $
$        --------------------------                                             $
$        00000011111111110000000000                                             $
$                                                                               $
ALTER 455                                                                       $
LAMX EMAT,LAMA/LAMB/ 10 $ELIMINATE UNWANTED FREQUENCIES&REORDER TABLE$
EQUIV LAMB,LAMA/TRUE $                                                          $
PURGE MI/ALWAYS $                                                               $
$                                                                               $
PARTN PHIA,CP,/A11,A21,A12,A22/1 $ PARTITION EIGENVECTOR MATRIX (A-SET)         $
EQUIV A11,PHIA/TRUE $                                                           $
$                                                                               $
```

Figure 5 Example of modified MSC/NASTRAN Solution 31 Executive Control Deck

```
$$$$$$$$ ALTER TO SUBSTITUTE CONTROLS-CODE-GENERATED MODAL SOLUTION    $$$
$                                                                        $
ALTER 479           $ SKIP MODULE TRD1 (TRANSIENT ANALYSIS)             $
JUMP SKIPTR                                                              $
ALTER 480                                                                $
LABEL SKIPTR                                                             $
$                                                                        $
INPUTT4/UHV,,,,//15/-1/1 $ READ UHV MATRIX FROM FOR015                   $
$                                                                        $
ENDALTER                                                                 $
$                                                                        $
$$$$$$$$$$$$$$$$ END OF TRANSIENT ANALYSIS ALTER  $$$$$$$$$$$$$$$$$$$$$$$$$$$ $
$                                                                        $
TIME 500
DIAG 8,16
RESTART  BAILEY  ,ZBORB1  ,10/ 8/87,   44192,
     1,  XVPS    ,  FLAGS = 0,  REEL =  1,  FILE =     5

               ..................CHECKPOINT  DICTIONARY....................

    257,  XVPS    ,  FLAGS = 0,  REEL =  1,  FILE =   126
$
$ END OF CHECKPOINT DICTIONARY
CEND
```

Figure 5 concluded

SECTION 2

EXECUTIVE

Introduction - 2- 1
IMAT Commands - 2- 1
IMAT EXECUTIVE - 2- 2

INTRODUCTION:

The IMAT EXECUTIVE is a program that allows the user to access and execute IMAT processors by means of a menu format. At the bottom of each menu in the IMAT EXECUTIVE is an option line which contains the additional commands defined below. These same commands can be found in the menus of the individual IMAT processors and will perform the function indicated in any menu which offers them.

COMMANDS USED THROUGHOUT IMAT:

H	Help	Display helpful information on the current problem.
M	Menu	In the IMAT EXECUTIVE - return to the MAIN MENU. In most IMAT processors - redisplay the current menu.
V	VMS	Enter an operating system (DCL) command.
Q	Quit	Leave the IMAT EXECUTIVE or terminate the current processor.
<CR>	Return	Select default option (see below). Return to the previous menu of the current processor if using the IMAT EXECUTIVE.
[ ]	Default	Indicates the option selected as default chosen by entering <CR> alone.
!!! <<< >>>	Errors	Error messages to the user are marked by either of these symbols.

PROCESSOR: IMAT EXECUTIVE

PURPOSE: The IMAT EXECUTIVE permits users to select and execute the IMAT utilities and processors from a tree-like menu structure.

DISPLAYING HELP INFORMATION TO A NEW USER:

When you first enter the IMAT EXECUTIVE you will be shown the INTRODUCTORY MENU:

IMAT EXECUTIVE: USER TYPE MENU

1. EXPERIENCED USER.
2. NEW USER. DISPLAY INFORMATIVE MESSAGES.

ENTER SELECTION: 1-2, Q OR FOR HELP ENTER ?

Enter option 1 if you are familiar with the system, or option 2 to see a summary of the commands used in IMAT. The following command options are presented in the prompt following each menu:

ENTER SELECTION: (1-4, H, M, Q, V)

These commands are explained in the IMAT COMMANDS section of the USER'S GUIDE. Local help may be accessed at any menu by entering H.

UPPER/LOWERCASE INPUT:

The EXECUTIVE will put your terminal into uppercase mode as you enter IMAT and return you to your previous case setting upon exit.

PROCESSOR SUMMARIES:

A brief summary of each processor and utility follows. Consult the documentation for the individual processor for a more detailed explanation.

I. DATABASE MANAGEMENT

1. CREATE DATABASE: will generate a new IMAT-defined RIM database containing supported IMAT relations.

2. QUERY DATABASE: will assist you in inspecting the contents of a database. No experience with RIM commands is required.

3. UNLOAD DATABASE: will unload a RIM database to a text file for transfer to other computer systems.

4. EXECUTE RIM: will require you to communicate directly with the RIM system. You should be conversant with RIM commands before selecting this option.

II. GRAPHICS

1. GENERATE METAFILE: will convert picture files from several sources (SUPERTAB, GEOMOD, MATRIXx) to a vector or raster metafile.

2. DISPLAY METAFILE: will allow you to view metafiles using the Metafile Translator.

3. GENERATE PICSURE COMMAND FILE: will generate a command file for PicSure to create plots from MATRIXx and MSC/NASTRAN.

4. EXECUTE PICSURE: will execute the PicSure graphics system. This processor allows you to interactively create charts or graphs and save them as metafiles.

5. HARDCOPY METAFILES (for LaRC use only): will generate the output of DI-3000 metafiles and Raster Metafile format files on ACD production plotters.

III. FINITE ELEMENT ANALYSIS

A. MSC/NASTRAN

1 LOAD BULK DATA: will load a file of MSC/NASTRAN bulk data into an IMAT database.

2. FORMAT BULK DATA: will format a MSC/NASTRAN bulk data deck from finite element model data stored in an IMAT database.

3. EXECUTE MSC/NASTRAN: will run MSC/NASTRAN using a bulk data deck generated by option 2 (or any MSC/NASTRAN bulk data deck).

4. LOAD ANALYSIS RESULTS: will load a file of MSC/NASTRAN analysis data into an IMAT database.

5. RECOVER PHYSICAL OUTPUT: will recover physical output using MSC/NASTRAN from a state-space time simulation calculated using MATRIXx.

B. SUPERTAB

1. EXECUTE SUPERTAB/GEOMOD: interactively executes SDRC I-DEAS SUPERTAB or GEOMOD.

2. LOAD UNIVERSAL FILE: will load finite element model data from an I-DEAS universal file into an IMAT database.

3. FORMAT UNIVERSAL FILE: will generate an I-DEAS universal file from finite element model data stored in an IMAT database.

IV. CONTROLS

1. DEFINE CONTROL SYSTEM: allows you to interactively edit your database to enter or modify control system information.

2. GENERATE LINEAR SYSTEMS MATRICES: generates a MATRIXx file containing the linear systems matrices from a control system defined in an IMAT database.

3. EXECUTE MATRIXX: interactively executes MATRIXx /SYSTEM_BUILD with a choice of memory size.

4. RECORD CONTROL SIMULATION: allows you to interactively edit your database to enter the specifics of a controls simulation. This information will be required before recovering physical output (MSC/NASTRAN MENU).

5. QUERY EIGENVECTOR FILE: allows you to examine the eigenvector file created by the LOAD ANALYSIS RESULTS processor (MSC/NASTRAN MENU).

SECTION 3

DATABASE MANAGEMENT

Database Overview - 3- 1

Database Management Processors

CREATE DATABASE - - - - - - - - - - - - - - - - 3- 2
QUERY DATABASE - - - - - - - - - - - - - - - - 3- 4
UNLOAD DATABASE - - - - - - - - - - - - - - - 3-11
EXECUTE RIM - - - - - - - - - - - - - - - - - - - 3-12

Database Overview 1

The Relational Information Management System (RIM) developed by Boeing Computer Services is the information manager used by IMAT. IMAT has a maintained RIM database definition or schema which contains 27 relations and 158 attributes at the present time. The relations and attributes are described in APPENDIX A. A maintained schema ensures that information may be shared by the IMAT users.

A relational database stores data as a collection of two-dimensional tables called relations. A relation consists of rows, sometimes called tuples, and columns called attributes. A row represents an occurrence of a group of attributes. Attributes may be of different types - text, integer, real, vector, matrix, date, and time. Attributes names may appear in more than one relation.

When RIM creates a database, it generates three files. The database name may contain up to six alphanumeric characters. To this RIM will add a suffix of 1, 2, or 3. If a database is named DBNAME, then the database files are DBNAME1.DAT, DBNAME2.DAT, and DBNAME3.DAT. DBNAME1.DAT contains the schema which describes the database structure. DBNAME2.DAT is the file which has the actual data for each relation and DBNAME3.DAT has index pointers. The CREATE DATABASE processor will create an empty IMAT-defined database for you.

The RIM database may be written to an ASCII file for transfer to other computer systems. This is called unloading the database. An IMAT processor, UNLOAD DATABASE, will unload your database for you.

If you would like more information about RIM, consult RIM - RELATIONAL INFORMATION MANAGEMENT SYSTEM USER'S MANUAL, 1985, The Boeing Company.

PROCESSOR: CREATE DATABASE

PURPOSE: This processor will create an IMAT-defined RIM database. The name of the database is chosen by the user.

THE RIM DATABASE:

The password for the RIM database will be set to NONE, and the database will contain all of the standard relations required for IMAT. The database will be created with delete protection to avoid accidental deletion. If you want to delete the database, you must reset the protection on each of the three database files. For example, the command, SET PROTECTION=(O:RWED) DBNAME%.DAT will reset the protection for database DBNAME.

ENTERING THE DATABASE NAME:

The program will ask you to enter a name for your database. The name may be no longer than six characters, and must begin with an alphabetic character. The name may not have embedded blanks. If the database name you enter is invalid, the program will give you a chance to enter a different name. The program will not allow you to overwrite a database of the same name in your directory. It will inform you that a database of that name exists and give you the choice of either entering a different name or exiting the processor.

ENTERING THE DATABASE DESCRIPTION:

In addition to entering a name for the database, you must also supply the information for the database description relation DBDESC. You will be asked to enter the following information:

1. The name of the project.
2. The name of the person responsible for the database.
3. The phone number where the responsible person can be reached.
4. A brief description of the database.

5. The type of units used. The units may be any consistent system whose primary physical quantities are mass (M), length (L), and time (T).

The values that you enter will be added to the relation DBDESC when the database is created.

PROCESSOR: QUERY DATABASE

PURPOSE: The QUERY DATABASE processor allows the user to inspect the contents of a RIM database without having to know RIM commands.

SELECTING A DATABASE:

When you first enter the QUERY DATABASE processor, you will be prompted to enter the name of the database you wish to inspect. If your database is not in the current directory, the program can access your database from another directory as long as you have "read" permission to the database files. Enter the pathname to the directory where the database files are located:

EX: DUA0:[YOURDIR.YOURSUBDIR]

SELECTING AN OUTPUT MODE:

The QUERY DATABASE processor will now present the OUTPUT MODE MENU:

QUERY DATABASE: OUTPUT MODE MENU

ENTER OUTPUT MODE DESIRED:

1. TERMINAL OUTPUT ONLY
2. TERMINAL OUTPUT PLUS LOG FILE
3. LIST DATABASE CONTENTS TO TEXT FILE
4. RETURN TO DATABASE ENTRY PROMPT [DEFAULT]

CURRENT LOGFILE: NONE

CHANGE: D-DATABASE ... OR
ENTER SELECTION: (1 - 4, H, M, Q, V)

Choose option 1 if you would like all output go to the terminal screen only or option 2 to have a log file generated in addition to screen output. The log file will have the name of your DATABASE.LOG and will first contain a listing of all the relations in the database, and the schema and data from any relations you inspect. A log file will only contain data from one database. If you choose to look at a different database from any point in the program (at this menu - either by choosing option 4 or D from the options line at the bottom of the menu), the log file will be closed and you will have the option of opening a new log file from this menu.

If you currently have an open log file when you see this menu, an alternate set of options is presented:

QUERY DATABASE: OUTPUT MODE MENU

ENTER OUTPUT MODE DESIRED:

1. TERMINAL OUTPUT ONLY
2. TERMINAL OUTPUT - OPEN NEW LOG FILE
3. TERMINAL OUTPUT - CONTINUE CURRENT LOG FILE
4. LIST DATABASE CONTENTS TO TEXT FILE
5. RETURN TO DATABASE ENTRY PROMPT [DEFAULT]

CURRENT LOGFILE: YOURDB.LOG;1

CHANGE: D-DATABASE ... OR
ENTER SELECTION: (1 - 4, H, M, Q, V)

From this alternate OUTPUT MODE MENU you may either continue with the same log file for this database (option 3) or close the current log file and open a new one (option 2).

Option 3 (LIST DATABASE CONTENTS TO TEXT FILE - This is option 4 in the alternate menu) will generate a file which contains a full listing of the schema and contents of your database. If you select this option, you will be prompted to enter a name for your listing file, and to select 80 or 132 column width for your output, whether you want to include the schema for empty relations, and whether you want the listing generated interactively or submitted as a batch job. If the database contains a large amount of data (a relation with over 2000 rows), it would be better to submit this as a batch job.

> NOTE: To prevent any conflict with your currently opened database, this batch job will be submitted with a one-hour delay. You can change the execution time by issuing the DCL command "SET QUEUE /ENTRY=xxx /AFTER=xxx SYS$BATCH". Consult VMS on-line help for the specifics of this command.

SELECTING A RELATION:

Once you have selected an output mode, the QUERY DATABASE processor will present a menu of the relations (tables) in your database:

QUERY DATABASE: "YOURDB" RELATIONS MENU

	RELATION	ATTRIBUTES	ROWS
1.	ACTUATOR	4	4
2.	BEAMPROP	12	6
3.	BEAMREF	5	22
4.	BEAMS	17	174
5.	...	...	...

[MORE] ENTER <CR> TO CONTINUE ... OR
CHANGE: D-DATABASE O-OUTPUT ... OR
ENTER SELECTION: (1 - 14, H, M, Q, V)

Choose one database relation at a time to inspect. If you see the prompt: [MORE], your database contains more relations than appear on the screen. Enter a <CR> if you would like to see these additional relations. Additional commands available directly from this menu are:

M	-	to return to the top of the RELATION MENU
D	-	to select a new database
O	-	to return to the the OUTPUT MODE MENU to select a new output mode

SELECTING ATTRIBUTES:

Once you have selected a relation, the QUERY DATABASE processor will present a menu of the attributes (data items) in the relation. This menu will give the name of each attribute, its data type, units, and if available, a description of the attributes. The following data types may be present:

I -	integer	V5 -	5-element vector
R -	real	VV -	variable length vector
D -	date	T8 -	8-character text
M4,5 -	4x5 matrix	TV -	variable length text
		MV -	variable length matrix

Enter an A if you want to see all the attributes in a relation, or separate individual selections with a comma. You may use a dash to indicate a range (EX: 1, 3-5, 9). If more than one screen must be used to display all of the attribute choices for a relation, the program will ask you to choose attributes from all screens for the relation before actually retrieving any data.

Additional commands available directly from the ATTRIBUTES MENU are:

M	-	to return to the top of the ATTRIBUTE MENU
D	-	to select a new database
O	-	to return to the the OUTPUT MODE MENU to select a new output mode
R	-	to select a new relation
W	-	toggles on and off the program option to qualify data retrieval using a RIM WHERE condition (See Below).

QUALIFYING THE DATA ITEMS YOU SELECT:

You may not wish to see all rows in your relation but may want to select only certain rows that meet a specific qualification (e.g., a data item greater than 100). The QUERY DATABASE processor provides a WHERE option to allow you to do this. You may turn this option ON or OFF at any time by entering a W at the ATTRIBUTES MENU.

When you first enter the QUERY DATABASE processor, the WHERE option is turned ON. Unless you immediately turn this option OFF, the first time you complete a choice from the ATTRIBUTES MENU, the program will present a help screen with a summary of the WHERE commands. (In future executions, you may want to set the WHERE option to OFF to prevent the display of this help screen.)

When the WHERE option is ON, the program will prompt you to enter up to five RIM condition clauses to further qualify the data items you select. The following RIM data qualification clauses are supported by this processor:

COMMAND (ATT1 & ATT2 ARE COLUMNS)		SELECTS ROWS WHERE
WHERE...		
ATT1 EXISTS		ATT1 HAS A VALUE
ATT1 FAILS		ATT1 HAS NO VALUE
ATT1 EQ 3	(ALSO: NE,GT,GE,LT,LE)	ATT1 EQUALS 3
ATT1 EQS "JONES"	(ALSO:NES,CONTAINS)	ATT1 EQUALS SUBSTRING "JONES"
ATT1 EQA ATT2	(ALSO:NEA,GTA,LTA,LEA)	ATT1 EQUALS ATTRIBUTE ATT2
ROWS EQ 1,3,5	(ALSO: NE,GT,GE,LT,LE)	ROW NUMBER EQUALS 1,3,5

NOTE: EQ (equals), NE (not equals), GT (greater than), GE (greater than or equal), LT (less than), LE (less than or equal), EQS (equals substring), NES (not equal substring), CONTAINS (contains substring), EQA (equals attribute), NEA (not equal attribute), GTA (greater than attribute), GEA (greater than or equal to attribute), LTA (less than attribute), LEA (less than or equal attribute).

Be certain to enclose your comparison value in double quotes and use the operators EQS, NES, or CONTAINS, when selecting text attributes. Enter one clause at a time when the QUERY DATABASE processor prompts you:

CONDITION CLAUSE # 1: ... OR H-HELP A-ATTRIBUTES S-STOP [<CR>-NONE]

For example, you might enter:

NODENUM EQ 100

Also at this prompt you may enter A to get a brief listing again of the attributes in the current relation, H to see the help screen again, or S to return to the ATTRIBUTE MENU to enter a different list of attributes desired (or to return to the RELATION MENU, etc.). A <CR> at this point will end your entry of condition clauses. (If this is the first condition clause, this would be the equivalent of entering no WHERE condition at all.

After each condition clause you enter (for the first 4 clauses), the program will present the prompt:

ANOTHER CONDITION? ENTER: OR/AND/[NONE]

Enter the Boolean value OR or AND if you wish to make a multiple clause qualification (i.e. where NODENUM GT 10 AND NODENUM LT 100), or a <CR> to end the condition clause input.

DISPLAYING THE DATA:

The QUERY DATABASE PROCESSOR will now display the data you have selected one screenful at a time, followed by the prompt:

MORE ... CONTINUE ? [Y]/N

Enter a <CR> to continue or N to return to the ATTRIBUTES MENU if you want to stop the data display. In general, depending on the data types, approximately seven attributes will fit on a screen. If you have selected more than seven attributes, the program will show you data for the first seven attributes chosen before going on to the remaining attributes selected. The program will return you to the ATTRIBUTES MENU after displaying all the data you have chosen.

NOTE: The QUERY DATABASE processor may be very slow when inspecting extremely large relations - in particular, the IMAT relation, EIGNVECT (eigenvectors). Use the QUERY EIGENVECTOR FILE (CONTROLS MENU), to look at eigenvectors.

EXITING THE PROCESSOR:

Enter a Q at any menu offering it to exit the QUERY DATABASE processor. If you have requested a log file, it will be in your default directory when you exit the IMAT EXECUTIVE. If you have requested a batch-mode database listing, the QUERY DATABASE processor will inform you of any other files you should expect to see.

PROCESSOR: UNLOAD DATABASE

PURPOSE: The UNLOAD DATABASE processor permits users to unload a RIM database to an ASCII text file which may be transferred to other computer systems. The unload function may be executed interactively or submitted to a queue for execution in a batch environment.

ENTERING THE NAME OF THE DATABASE:

You will be asked to enter the name of the database that you want to unload. If the database is not in your current directory, you will be asked to supply the full pathname of the directory where the database is located. The processor will make a temporary copy of the database in your current directory and will delete it upon completion.

SELECTING RELATIONS:

After a database is selected, you will be presented with a menu of submission types available. You may select BATCH, INTERACTIVE, or TERMINATE. TERMINATE is the equivalent of entering a Q (quit). Selecting INTERACTIVE will cause the RIM unload to begin immediately and you must wait for its completion. This may require 30 minutes or more depending on computer activity and database size. Selecting BATCH will submit a job to the SYS$BATCH queue for execution in a batch environment.

TERMINATION:

When you have completed your session with UNLOAD DATABASE and elect to quit, all files you have generated will remain in your current directory. You will then be returned to the IMAT EXECUTIVE.

PROCESSOR: EXECUTE RIM

PURPOSE: This processor allows the user to access the standalone version of Boeing RIM (Relational Information Management) from the IMAT EXECUTIVE.

You must be familiar with RIM commands to use the standalone version. By executing RIM directly, you will be able to modify or examine the data in the database, protect the data by adding passwords to specific relations or to the entire database, and create new relations from existing relations. For a discussion of RIM commands consult the RIM USER'S MANUAL.

SECTION 4

GRAPHICS

Metafile Definition - 4- 1

Graphics Processors

GENERATE METAFILE - - - - - - - - - - - - - - - 4- 2
DISPLAY METAFILE - - - - - - - - - - - - - - - - 4-11
GENERATE PICSURE COMMAND FILE - - - - - - 4-15
EXECUTE PICSURE - - - - - - - - - - - - - - - - - 4-28
HARDCOPY METAFILE (for LaRC use only) - - - - 4-29

Vector Metafile

Device-independent display format, which is composed of positioning (move, draw) and non-positioning (color, etc.) commands, providing a universal method for the transfer of graphical images between vector display devices.

Raster Metafile

Device-independent display format consisting of an image header, which describes the image format, and opcodes, which are used to delineate entities (run-lengths, pixel values, color maps, intensity) within the image structure, providing a method for the transfer of graphical images between raster-oriented devices.

PROCESSOR: GENERATE METAFILE

PURPOSE: The GENERATE METAFILE processor provides the IMAT user with the ability to translate package specific graphics output into device-independent metafile format. Allowable package specific output includes: I-DEAS, MSC/NASTRAN, and MATRIXx. This processor converts the graphics output into vector or raster metafile format.

When the GENERATE METAFILE processor is invoked, the following menu will be displayed:

GENERATE METAFILE: TRANSLATOR MENU

1. I-DEAS
2. NASTRAN
3. MATRIXX
4. EXIT TO PREVIOUS MENU [DEFAULT]

This menu allows you to select the type of graphics output data you wish to have processed.

TRANSLATOR MENU OPTION 1: I-DEAS

This option will process I-DEAS picture file data. This data may be of vector or raster/shaded data type and you should be aware of the data format of your file. Upon selection of option 1 the following menu will be displayed:

I-DEAS METAFILE: DATA TYPE MENU

1. VECTOR DATA
2. RASTER DATA (512 X 512)
3. RASTER DATA (1024 X 1024)
4. RASTER DATA (2048 X 2048)
5. EXIT TO PREVIOUS MENU [DEFAULT]

GENERATE METAFILE 2

Option 1 (VECTOR DATA) of the I-DEAS METAFILE MENU will process I-DEAS vector data into a DI-3000 metafile format. You will be asked to enter the name of the file containing the I-DEAS vector data.

ENTER THE NAME OF THE FILE CONTAINING THE I-DEAS VECTOR PICTURE DATA:
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

You will then be prompted to enter the file type.

I-DEAS METAFILE: FILE TYPE MENU

1. BINARY FORMAT [DEFAULT]
2. FORMATTED

You must enter the format of the data file that is being processed. This format selection was made within I-DEAS when picture output was activated. The next question within the processor relates to the SDRC header information placed by default on all graphics output within I-DEAS.

DO YOU WISH TO HAVE THE SDRC HEADER
INFORMATION INCLUDED WITH YOUR PICTURE ?

ENTER SELECTION: H, Y OR [N]

Selecting N (the default) will remove the SDRC header information from the final output written to the metafile. The final question regards the name of the metafile.

ENTER VAX METAFILE NAME:

You will then be asked if you wish to process another vector data file.

PROCESS ANOTHER I-DEAS VECTOR FILE ?

ENTER SELECTION: H, Y OR [N]

Entering N (the default) will return you to the I-DEAS METAFILE: DATA TYPE MENU.

Option 2 (RASTER DATA (512 X 512)) of I-DEAS MENU will process I-DEAS raster/shaded data. This option will translate raster data into the device-independent raster metafile format with a resolution of 512 x 512. The resolution is determined by the I-DEAS device driver used during creation of the raster file. Raster files with a resolution larger than 512 x 512 will be clipped. This translation of raster data is done by the following sequence:

ENTER THE NAME OF THE FILE CONTAINING THE I-DEAS RASTER PICTURE DATA:
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

This question allows you to select the I-DEAS raster picture file to be processed. An entry of carriage return will return you to the I-DEAS METAFILE: DATA TYPE MENU. This selection is followed by the FILE TYPE MENU:

I-DEAS METAFILE: FILE TYPE MENU

1. BINARY FORMAT [DEFAULT]
2. FORMATTED

You must enter the option number of the format of your data file that is to be processed. This format was selected within I-DEAS when picture output was activated. An entry of carriage return will result in the default selection of binary format.

The next question regards the raster metafile name.

ENTER THE NAME OF THE FILE TO CONTAIN RASTER OUTPUT DATA:
(ENTER FULL PATHNAME IF FILE IS NOT TO BE LOCATED IN CURRENT DIRECTORY.)

This allows the user to select the name of the output file. The next question relates to the SDRC header information placed by default on all graphics output within I-DEAS.

DO YOU WISH TO HAVE THE SDRC HEADER INFORMATION INCLUDED IN YOUR PICTURE?

ENTER SELECTION: H, Y OR [N]

Selecting N (default) will remove the SDRC header from the final output. SDRC header selection is followed by the BACKGROUND COLOR MENU.

I-DEAS METAFILE: BACKGROUND COLOR MENU

1. BLACK
2. WHITE
3. USER DEFINED
4. BACKGROUND COLOR FROM I-DEAS FILE [DEFAULT]

This menu allows you to choose the color of the background regardless of what the background color was defined when the picture was created in I-DEAS. Selection of option 3 will require red, green, and blue intensity values from 0 to 255 to be entered.

ENTER RED COMPONENT (VALUE 0 - 255):

ENTER GREEN COMPONENT (VALUE 0 - 255):

ENTER BLUE COMPONENT (VALUE 0 - 255):

After processing, which may take 5 to 30 minutes, you will be asked if you wish to process another raster data file.

PROCESS ANOTHER I-DEAS RASTER FILE ?

ENTER SELECTION: H, Y OR [N]

Entering N (default) will return you to the I-DEAS METAFILE: DATA TYPE MENU.

Option 3 (RASTER DATA (1024 X 1024)) of the I-DEAS MENU will process I-DEAS raster/shaded data. This option will translate raster data into a device-independent raster metafile format with a resolution of 1024 x 1024. Raster files with resolutions larger than 1024 x 1024 will be clipped. Raster files with a resolution less than 1024 x 1024 will be centered within the 1024 x 1024 resolution. The process for this option follows the same sequence as option 2 (RASTER DATA (512 X 512)) and should be referenced in regards to this option.

Option 4 (RASTER DATA (2048 x 2048)) of the I-DEAS MENU will process I-DEAS raster/shaded data. This option will translate raster data into a device-independent raster metafile format with a resolution of 2048 x 2048. Raster files with resolutions larger than 2048 x 2048 will be clipped. Raster files with a resolution less than 2048 x 2048 will be centered within the 2048 x 2048 resolution. The process for this option follows the same sequence as option 2 (RASTER DATA (512 X 512)) and should be referenced in regards to this option.

Option 5 (EXIT TO PREVIOUS MENU) of I-DEAS MENU returns you to the GENERATE METAFILE: TRANSLATOR MENU.

GENERATE METAFILE MENU OPTION 2: MSC/NASTRAN

Option 2 (MSC/NASTRAN) of GENERATE METAFILE will process MSC/NASTRAN plot files. This option handles the translation of a plot file into a DI-3000 vector metafile. You will be asked to enter the name of the file containing the MSC/NASTRAN plot data.

> ENTER THE NAME OF THE FILE CONTAINING THE MSC/NASTRAN PLOT DATA:
> (ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

After the plot file name is entered, the data is immediately processed. The frame number and the number of data points within the current frame will be displayed. When processing is completed, the following question will be displayed.

> ENTER VAX METAFILE NAME:

GENERATE METAFILE 6

This allows you to enter the name of the metafile. The final question regards processing another plot file.

PROCESS ANOTHER MSC/NASTRAN PLOT FILE ?

ENTER SELECTION: H, Y OR [N]

Entering N (default) will return you to the GENERATE METAFILE MENU.

TRANSLATOR MENU OPTION 3: MATRIXX

Option 3 (MATRIXX) of GENERATE METAFILE MENU will process MATRIXx laser output files. The MATRIXx file must be output by a laser type device driver (ex. LNO3). Typically the default output name of the MATRIXx laser device driver is MATLASER.DAT. Once the MATRIXx data is of the laser output format then this option may be utilized to translate the output into a device-independent DI-3000 metafile. You will be prompted for the name of the file containing the MATRIXx laser picture data.

ENTER THE NAME OF THE FILE CONTAINING THE MATRIXX LASER PICTURE DATA:
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

After entering the MATRIXx file name, the main MATRIXX PROCESSOR MENU will be displayed.

MATRIXX METAFILE: FILE ACTION MENU

1. LIST FRAME NUMBERS
2. PROCESS FILE
3. ENTER A NEW FILE NAME
4. EXIT TO PREVIOUS MENU [DEFAULT]

Option 1 (LIST FRAME NUMBERS) will list the number of frames in the selected data file.

Option 2 (PROCESS FILE) processes the laser data file. To process the data, a series of questions will be asked about the output attributes to be used during processing. The first attribute is the orientation of the data.

MATRIXX METAFILE: PLOT ORIENTATION MENU

1. PORTRAIT [DEFAULT]
2. LANDSCAPE

Portrait orientation will cause the data to be processed in a window dimensioned 8.5 inches wide by 11 inches high. Landscape results in a window of 11 inches wide by 8.5 inches high.

The number of frames to be processed is entered next.

ENTER FRAME NUMBER TO BE PROCESSED:
ENTER START FRAME NUMBER OR <CR> FOR DEFAULT:

Entering carriage return (<CR>) results in the processing of all frames on the data file. Entering a number will result in the starting frame number and will be followed with the question:

ENTER END FRAME NUMBER OR <CR> FOR DEFAULT:

Entering <CR> causes all remaining frames to be processed. Entering a number will result in all frames through the ending frame number to be processed. The scale factor attribute menu is displayed next.

ENTER SCALE FACTORS:

ENTER X-SCALE FACTOR OR <CR> FOR DEFAULT:

ENTER Y-SCALE FACTOR OR <CR> FOR DEFAULT:

Entering a <CR> results in the default of no scaling of data. Entering a real number scale factor will result in data being scaled. X and Y scale factors should be kept proportional to original X,Y ratio. This is the responsibility of the user. Next you will be prompted for the number of the character font you would like.

ENTER FONT NUMBER OR <CR> FOR DEFAULT:

The entering of a font number will choose a specific DI-3000 character font. Font choice is only possible when the DI-3000 library is loaded with the processor, otherwise hardware characters are used for text. By default, font capability is not available. If font capability is needed, see your System Administrator. The final choice is a color option:

MATRIXX METAFILE : COLOR PROCESSING MENU

1. B/W [DEFAULT]
2. COLOR

Option 1 (B/W) will process data in black and white.

Option 2 (COLOR) provides a crude method of adding color to MATRIXx plots. This option will inform the user of all pen changes, font calls (indicating text output), symbol changes and new frames. At each point the user will be allowed to choose a color. The user must be warned that there is no distinction within the data file of axis, tick marks, data curves, etc., and color consistency will have to come through trial and error in adding color at the pen changes. After the data has been processed you will be asked:

ENTER VAX METAFILE NAME:

After entering a metafile name you will be asked:

PROCESS ANOTHER MATRIXX LASER FILE ?

ENTER SELECTION: H, Y OR [N]

Selecting N (default) will return you to the GENERATE METAFILE : TRANSLATOR MENU. Option Y will return you to the beginning of the MATRIXX Processor.

Option 3 (ENTER A NEW FILE NAME) will allow the user to enter a new laser data file name for processing.

Entering Q (QUIT) of GENERATE METAFILE MENU will return you to the IMAT EXECUTIVE: GRAPHICS MENU.

PROCESSOR: DISPLAY METAFILE

PURPOSE: The DISPLAY METAFILE processor provides the IMAT user with the ability to display DI-3000 metafiles and Raster Metafile format files on interactive display devices.

When the DISPLAY METAFILE processor is invoked, the following menu will be displayed:

DISPLAY METAFILE: METAFILE TYPE MENU

1. VECTOR
2. RASTER

METAFILE TYPE MENU OPTION 1: VECTOR

Option 1 (VECTOR) will let you interactively display DI-3000 metafiles. By using this path you will be presented with the following sequence of menus:

DISPLAY METAFILE: DEVICE SELECTION MENU

1. VT 240
2. TEK 4105
3. TEK 4107/4109
4. TEK 4114
5. TEK 4115
6. POSTSCRIPT
7. EXIT TO PREVIOUS MENU [DEFAULT]

Select the device type which you are using. The device list should reflect the device drivers available on your machine. You will then be presented with a list of DI-3000 commands. It might be helpful to have your printer turned on for this display or make a hard copy of the display as you will need to know these commands during the plotting session. The display is presented here for your convenience:

DISPLAY METAFILE 2

<<< YOU WILL NEED THE FOLLOWING COMMANDS TO >>>
<<< RESPOND TO THE DI-3000 PROMPTS TO FOLLOW >>>

THE FOLLOWING IS A SAMPLE OF TRANSLATOR COMMANDS

S MF # FILE NAME	<- SETS METAFILE TO FILE NAMED
D P # MF #	<- DRAW PICTURE
D P FROM FIRST TO LAST	<- DRAWS ALL PICTURES
S W # (L R B T)	<- SET WINDOW
S V # (L R B T)	<- SET VIEWPORT
QUIT	<- EXIT TRANSLATOR
S DEV # ON/OFF	<- SET DEVICE ON/OFF

<<< WAIT FOR M> PROMPT TO RESPOND >>>

NOTE: # should be replaced with a number.

At this point, your terminal will be placed in graphics mode and you may use the above commands to respond to prompts. See the METAFILE SYSTEM USER'S GUIDE for a complete description of metafile commands.

METAFILE TYPE MENU OPTION 2: RASTER

Option 2 (RASTER) will let you interactively display Raster Metafiles. By using this path you will be presented with the following menus:

DISPLAY METAFILE: DEVICE SELECTION MENU

1. CONVERT INTENSITY TO COLOR TABLE
2. TEK 4107/4109
3. TEK 4125
4. AED767
5. DICOMED
6. VERSATEC

DISPLAY METAFILE 3

7. POSTSCRIPT
8. DUMMY DRIVER
9. EXIT TO PREVIOUS MENU [DEFAULT]

Option 1 (CONVERT INTENSITY TO COLOR TABLE) will convert the intensity raster file, which is produced in GENERATE METAFILE I-DEAS processor, into a color table raster file. Once in color table format, the file may be interactively displayed on a raster terminal.

NOTE: Option 1 must be selected and executed prior to any interactive display.

Select the device type you are using (options 2 through 8). You will then be presented with a short list of Raster Metafile Translator commands. Appendix C contains a complete list of Raster Metafile Translator commands. Entering HELP while in the Raster Metafile Translator will activate an on-line help facility.

<<< YOU WILL NEED THE FOLLOWING COMMANDS TO RESPOND TO >>>
<<< THE RASTER METAFILE TRANSLATOR PROMPTS TO FOLLOW >>>

THE FOLLOWING IS A SAMPLE OF TRANSLATOR COMMANDS

SET MF # FILE NAME	<- SET METAFILE TO FILE NAMED
D P # MF #	<- DRAW PICTURE
QUIT	<- EXIT TRANSLATOR
SET WINDOW # (XMIN XMAX YMIN YMAX)	<- SET WINDOW
SET VIEWPORT # (XMIN XMAX YMIN YMAX)	<- SET VIEWPORT
CONVERT MF # MF #	<- CONVERTS METAFILE FORMAT

<<< WAIT FOR RMT> PROMPT TO RESPOND >>>

DISPLAY METAFILE 4

At this point, your terminal will be placed in graphics mode and you may use the above commands to respond to prompts. When QUIT is entered, you will return to the IMAT EXECUTIVE.

PROCESSOR: GENERATE PICSURE COMMAND FILE

PURPOSE: The GENERATE PICSURE COMMAND FILE processor provides the IMAT user with the ability to create PicSure command files, which will generate X,Y plots from data files. The data files can be from MATRIXx (vector, matrix, or Bode formats), MSC/NASTRAN (PUNCH file format), or free format data.

When the GENERATE PICSURE COMMAND FILE processor is invoked, the following menu will be displayed:

GENERATE PICSURE COMMAND FILE: PLOT TYPE MENU

1. MATRIXX VECTOR/MATRIX PLOTS
2. MATRIXX BODE PLOTS
3. NON-MATRIXX DATA

PLOT TYPE MENU OPTION 1: MATRIXX VECTOR/MATRIX PLOTS

Option 1 (MATRIXX VECTOR/MATRIX PLOTS) will allow you to generate PicSure command files which will create X, Y plots from MATRIXX formatted data. The following sequence of menus will be displayed for this option.

ENTER THE NAME OF THE FILE CONTAINING THE MATRIXX DATA:
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

Enter the name of the file which contains the MATRIXx formatted data. This will be followed by:

ENTER A NAME FOR YOUR PICSURE COMMAND FILE [PIC.CMD]:

Enter the name of a file which will contain the PicSure commands. Entering carriage return will result in the default file name of PIC.CMD. This question will be followed by a list of the data components found in the named data file. You will then be requested to select one independent dataset from the list and up to seven dependent datasets. The format of the dataset list is as follows:

```
GENERATE PICSURE COMMAND FILE: DATA SET MENU
DATA VARIABLES FOUND IN MATRIXX FILE: [FILE NAME]

            VARIABLE          ROWS      COLUMNS
            -----------       ------    ----------

1           VARIABLE 1         #          #
2           VARIABLE 2         #          #
              .
              .
              .
N           VARIABLE N         #          #

NOTE:       DATASETS CHOSEN MUST BE VECTORS WITH
            DIMENSIONS THAT MATCH.


SELECT ONE INDEPENDENT VARIABLE:

SELECT DEPENDENT VARIABLE # 1:

SELECT DEPENDENT VARIABLE # 2 [NONE]:
```

NOTE: Datasets chosen must have the same dimensions.

Enter the number of the dataset you wish to select for each question. Enter a carriage return when you have completed your selections. At least one independent and one dependent dataset must be chosen to output a plot. If the variable chosen is of matrix format, (i.e., COLUMNS>1) then you will be asked to select which columns are to be used as dependent variables:

GENERATE PICSURE COMMAND FILE 3

NOTE: VARIABLE CHOSEN IS OF MATRIX FORMAT
PLEASE SELECT BY COLUMN.

MATRIX HAS COLUMNS 1 - N :
SELECT COLUMN #1 [NONE]:
SELECT COLUMN #2 [NONE]:

Select the column(s) desired and enter carriage return when done. You will then be prompted to enter text for your graph through the following sequence of questions:

ENTER A TITLE FOR THIS GRAPH:

ENTER A SUBTITLE FOR THIS GRAPH [NONE]:

ENTER A LABEL FOR THE X-AXIS:

ENTER A LABEL FOR THE Y-AXIS:

Enter a text string for each question. A carriage return for the subtitle will result in no subtitle for the graph. All questions must be answered with a text string. When this sequence has been completed the following message is displayed listing files that have been or will be created. Please read the note carefully.

THIS PROCESSOR CREATES THE FOLLOWING FILES:
COMMAND FILE ⇒ [FILE NAME]
DATA FILES ⇒ DATASET NAME FROM DATA FILE
WITH ONE OF THE FOLLOWING
ENDINGS: _I.DAT, _D.DAT, _D#.DAT

AFTER RUNNING THE COMMAND FILE WITHIN PICSURE
THE FOLLOWING FILES WILL EXIST:
METAFILE => [FILE NAME].MFL
CHART FILE => [FILE NAME].CHT

GENERATE PICSURE COMMAND FILE 4

BE SURE TO DELETE THESE FILES WHEN NO LONGER NEEDED.

YOUR PICSURE COMMAND FILE FOR THIS PLOT IS: [FILE NAME]
THIS COMMAND FILE SHOULD BE EXECUTED WITHIN PICSURE
WITH THE COMMAND:

READ COMMAND [FILE NAME]

BE SURE TO DELETE THIS COMMAND FILE AND ALL PREVIOUSLY
MENTIONED FILES WHEN YOU HAVE COMPLETED PROCESSING.

This command file will display the plot interactively and will generate a chart file and a vector metafile.

PLOT TYPE MENU OPTION 2: MATRIXX BODE PLOTS

Option 2 (MATRIXX BODE PLOTS) will allow you to create a Bode plot from phase-magnitude data. The command file for this format is generated through the following sequence of commands:

ENTER THE NAME OF THE FILE CONTAINING THE MATRIXX DATA:
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

Enter the name of the file containing the Bode plot data. Then you will be prompted for the name of a file to contain the PicSure commands.

ENTER A NAME FOR YOUR PICSURE COMMAND FILE [PIC.CMD]:

Entering a carriage return will result in the command file being named PIC.CMD. Following this question will be a list of the data elements found on the named file. The format of the data list is as follows:

GENERATE PICSURE COMMAND FILE: DATA SET MENU
DATA VARIABLES FOUND IN MATRIXX FILE: [FILE NAME]

	VARIABLE	ROWS	COLUMNS
	-------------	------	----------
1	VARIABLE 1	#	#
2	VARIABLE 2	#	#
	.		
	.		
	.		
N	VARIABLE N	#	#

NOTE: DATASETS CHOSEN MUST BE VECTORS WITH DIMENSIONS THAT MATCH.

SELECT ONE INDEPENDENT VARIABLE:

SELECT DEPENDENT VARIABLE # 1:

SELECT DEPENDENT VARIABLE # 2 [NONE]:

Enter the number of the dataset you wish to select for each question. Enter a carriage return when you have completed your selections. At least one independent and one dependent dataset must be chosen to output a plot. A maximum of two dependent datasets may be chosen for the Bode plot format. If the variable chosen is of matrix format, (i.e., COLUMNS>1) then you will be asked to select which column(s) are to be used as dependent variables by the following question:

NOTE: VARIABLE CHOSEN IS OF MATRIX FORMAT, PLEASE SELECT BY COLUMN.

MATRIX HAS COLUMNS 1 - N :
SELECT COLUMN #1:
SELECT COLUMN #2 [NONE]:

GENERATE PICSURE COMMAND FILE 6

Select column(s) and enter carriage return when done. You will then be prompted to enter text for your graph through the following sequence of questions:

ENTER A TITLE FOR THIS GRAPH:

ENTER A SUBTITLE FOR THIS GRAPH [NONE]:

ENTER A LABEL FOR THE X-AXIS:

ENTER A LABEL FOR THE Y-AXIS:

Enter a text string for each question. A carriage return for the subtitle will result in no subtitle for the graph. All questions must be answered with a text string. When two dependent datasets have been chosen you will be questioned for two y-axis labels:

ENTER A LABEL FOR THE FIRST DEPENDENT
DATASET CHOSEN FOR THE Y-AXIS:

ENTER A LABEL FOR THE SECOND DEPENDENT
DATASET CHOSEN FOR THE Y-AXIS:

When this sequence has been completed one of the following messages will be displayed. Please read the note carefully. For a single Bode plot, phase or magnitude only, the following message will be displayed:

THIS PROCESSOR CREATES THE FOLLOWING FILES:
COMMAND FILE ⇒ [FILE NAME]
DATA FILES ⇒ DATASET NAME FROM DATA FILE
WITH ONE OF THE FOLLOWING
ENDINGS: _I.DAT, _D.DAT

GENERATE PICSURE COMMAND FILE 7

```
AFTER RUNNING THE COMMAND FILE WITHIN PICSURE
THE FOLLOWING FILES WILL EXIST:
METAFILE           =>  [FILE NAME].MFL
TITLE CHART FILE   =>  [FILE NAME]_TITLE.CHT
CHART FILE         =>  [FILE NAME]_TOP.CHT

BE SURE TO DELETE THESE FILES WHEN NO LONGER NEEDED.

YOUR PICSURE COMMAND FILE FOR THIS PLOT IS: [FILE NAME]
THIS COMMAND FILE SHOULD BE EXECUTED WITHIN PICSURE
WITH THE COMMAND:

   READ COMMAND [FILE NAME]

BE SURE TO DELETE THIS COMMAND FILE AND ALL PREVIOUSLY
MENTIONED FILES WHEN YOU HAVE COMPLETED PROCESSING.
```

For a composite Bode plot, one in which both phase and magnitude are selected, the following message will be displayed:

```
THIS PROCESSOR CREATES THE FOLLOWING FILES:
COMMAND FILE  =>  [FILE NAME]
DATA FILES    =>  DATASET NAME FROM DATA FILE
                  WITH ONE OF THE FOLLOWING
                  ENDINGS:  _I.DAT, _D.DAT

AFTER RUNNING THE COMMAND FILE WITHIN PICSURE
THE FOLLOWING FILES WILL EXIST:
METAFILE           =>  [FILE NAME].MFL
TITLE CHART FILE   =>  [FILE NAME]_TITLE.CHT
TOP CHART FILE     =>  [FILE NAME]_TOP.CHT
BOTTOM CHART FILE  =>  [FILE NAME]_BOT.CHT
BE SURE TO DELETE THESE FILES WHEN NO LONGER NEEDED.
```

GENERATE PICSURE COMMAND FILE 8

YOUR PICSURE COMMAND FILE FOR THIS PLOT IS:
[FILE NAME]
THIS COMMAND FILE SHOULD BE EXECUTED WITHIN PICSURE
WITH THE COMMAND:

READ COMMAND [FILE NAME]

BE SURE TO DELETE THIS COMMAND FILE AND ALL PREVIOUSLY
MENTIONED FILES WHEN YOU HAVE COMPLETED PROCESSING.

The command file created will display the plot interactively and will generate a chart file and a vector metafile.

PLOT TYPE MENU OPTION 3: NON-MATRIXX DATA

Option 3 (NON-MATRIXX DATA) will allow you to process non-MATRIXx type data into an X,Y plot. The non-MATRIXx type data includes MSC/NASTRAN PUNCH file and free format data. First you must pick the type of data from the following menu:

NON-MATRIXX DATA: DATA FORMAT MENU

1. MSC/NASTRAN PUNCH DATA FORMAT
2. FREE FORMATTED DATA

Option 1 (NASTRAN PUNCH DATA FORMAT) will allow you to process MSC/NASTRAN data which was output to a PUNCH file using the XYPUNCH command. The command file for this format is generated through the following sequence of commands:

ENTER THE NAME OF THE FILE CONTAINING THE MSC/NASTRAN
DATA
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

Enter the name of the PUNCH file. You will then be prompted for the name of a file to contain PicSure commands.

ENTER A NAME FOR YOUR PICSURE COMMAND FILE [PIC.CMD]:

Entering a carriage return will result in the command file being named PIC.CMD. At this time the file is read to extract the data variables found in the file. There is a maximum of twenty allowable variables. If this limit is exceeded you will receive the following prompt:

EXCEEDED VARIABLE LIMIT, ONLY THE FIRST 20 VARIABLES ARE ACCESSIBLE.

ENTER <CR> TO CONTINUE

Enter carriage return to continue. The data variables retrieved will be listed in the following format:

GENERATE PICSURE COMMAND FILE: DATA SET MENU
DATA VARIABLES FOUND IN PUNCH FILE: [FILE NAME]

	DATA VARIABLES	NODE	DATA
1	$LOAD	65	3
2	$LOAD	102	3
3	$ACCE	11008	3
4	$ACCE	11008	4
5	$ACCE	11008	5
.			
.			
.			
N	$ACCE	######	#

You will then be prompted to enter the number of the variable you wish to have plotted:

SELECT VARIABLE:

You will then be asked to enter an increment value to be used in case of a large dataset which may exceed PicSure's data point limit of 10,000 data points. The increment will be used to select every Nth data value from the file.

ENTER INCREMENT TO BE USED WITH DATA VALUES:
ENTER VALUE 1 - 10 OR <CR>

Entering carriage return will result in all values being plotted. You will then be prompted to enter text for your graph through the following sequence of questions.

ENTER A TITLE FOR THIS GRAPH:

ENTER A SUBTITLE FOR THIS GRAPH [NONE]:

ENTER A LABEL FOR THE X-AXIS:

ENTER A LABEL FOR THE Y-AXIS:

Enter a text string for each question. A carriage return for the subtitle will result in no subtitle for the graph. All questions must be answered with a text string. When this sequence has been completed the following message is displayed listing the files which have been created or will be created. Please read the note carefully.

GENERATE PICSURE COMMAND FILE 11

```
THIS PROCESSOR CREATES THE FOLLOWING FILES:
COMMAND FILE => [FILE NAME]
DATA FILES   => DATASET NAME FROM DATA FILE
                WITH ONE OF THE FOLLOWING
                ENDINGS: _X.DAT, _Y.DAT

AFTER RUNNING THE COMMAND FILE WITHIN PICSURE
THE FOLLOWING FILES WILL EXIST:
METAFILE        => [FILE NAME].MFL
CHART FILE      => [FILE NAME].CHT
BE SURE TO DELETE THESE FILES WHEN NO LONGER NEEDED.

YOUR PICSURE COMMAND FILE FOR THIS PLOT IS: [FILE NAME]
THIS COMMAND FILE SHOULD BE EXECUTED WITHIN PICSURE
WITH THE COMMAND:

        READ COMMAND [FILE NAME]

BE SURE TO DELETE THIS COMMAND FILE AND ALL PREVIOUSLY
MENTIONED FILES WHEN YOU HAVE COMPLETED PROCESSING.

ENTER <CR> TO CONTINUE
```

This command file will display the plot interactively and will generate a chart file and a vector metafile.

Option 2 (FREE FORMATTED DATA) will allow you to process what PicSure considers free format data. This is any file which contains data points separated by commas, blank(s), or text. This is limited to a single dataset per file. The command file for this format is generated through the following sequence of commands:

```
ENTER A TITLE FOR THIS GRAPH:

ENTER A SUBTITLE FOR THIS GRAPH [NONE]:
```

ENTER A LABEL FOR THE X-AXIS:

ENTER A LABEL FOR THE Y-AXIS:

Enter a text string for each question. A carriage return for subtitle results in no subtitle for the graph. All questions must be answered with a text string (except for subtitle). No defaults are available. When this sequence has been completed, you will be prompted to enter a file name to contain the PicSure commands:

ENTER A NAME FOR YOUR PICSURE COMMAND FILE [PIC.CMD]:

Entering a carriage return will result in the file being named PIC.CMD. You will then be prompted to enter the name of the file containing the independent dataset. This will be followed by the prompt to enter the name to be given to the dataset within PicSure:

ENTER INDEPENDENT FILE NAME:

ENTER INDEPENDENT DATASET NAME:

At the next prompt, enter the name of the file containing the dependent dataset. This will be followed by the name to be given to the dataset within PicSure.

ENTER DEPENDENT FILE NAME:

ENTER DEPENDENT DATASET NAME:

These two questions will be repeated until a carriage return is entered for the dependent file name or seven dependent file names have been given. At this point the following message is displayed, listing the files that have been created or will be created. Please read the note carefully.

THIS PROCESSOR CREATES THE FOLLOWING FILES:
COMMAND FILE ⇒ [FILE NAME]

AFTER RUNNING THE COMMAND FILE WITHIN PICSURE
THE FOLLOWING FILES WILL EXIST:
METAFILE ⇒ [FILE NAME].MFL
CHART FILE ⇒ [FILE NAME].CHT
BE SURE TO DELETE THESE FILES WHEN NO LONGER NEEDED.

YOUR PICSURE COMMAND FILE FOR THIS PLOT IS: [FILE NAME]
THIS COMMAND FILE SHOULD BE EXECUTED WITHIN PICSURE
WITH THE COMMAND:

READ COMMAND [FILE NAME]

BE SURE TO DELETE THIS COMMAND FILE AND ALL PREVIOUSLY
MENTIONED FILES WHEN YOU HAVE COMPLETED PROCESSING
OF THE COMMAND FILE.

ENTER <CR> TO CONTINUE

This command file will display the plot interactively and will generate a chart file and a vector metafile.

Entering Q (QUIT) will return you to the IMAT EXECUTIVE:

EXECUTE PICSURE 1

PROCESSOR: EXECUTE PICSURE

PURPOSE EXECUTE PICSURE allows the user to use Precision Visuals' PicSure to generate charts and graphs interactively.

ENTERING YOUR GRAPHICS DEVICE:

At the beginning of execution, PicSure will display a menu of the available graphics devices and then prompt you to enter your graphics device type.

ENTER THE CODE FOR YOUR GRAPHICS TERMINAL:

Enter the code for the the graphics terminal or terminal emulation you are using. The following devices are examples of devices which may be supported and their PicSure code.

CODE	TERMINAL
405	TEK 4105
407	TEK 4107/4109
415	TEK 4115/4125
240	VT240
T14	TEK 4014
PST	POSTSCRIPT
750	HP 7550A

ON-LINE HELP:

Text charts, and line, bar, and pie charts may be created in PicSure using English-like commands. If you have never used PicSure, enter the command HELP TUTORIALS for an on-line tutorial session. For information on a specific command, type HELP or HELP XXXXXXXX where XXXXXXXX is the command for which you want an explanation. For more information on PicSure, see the PicSure USER'S GUIDE.

RETURNING TO THE IMAT EXECUTIVE:

Enter QUIT to exit from PicSure and return to the IMAT EXECUTIVE.

PROCESSOR: HARDCOPY METAFILE (for use at LaRC only)

PURPOSE: The HARDCOPY METAFILE processor provides the IMAT user with the ability to generate hardcopy output of DI-3000 metafiles and Raster Metafile format files on ACD Production devices.

When the HARDCOPY METAFILE processor is invoked, the following menu will be displayed:

HARDCOPY METAFILE: METAFILE TYPE MENU

1. VECTOR
2. RASTER
3. EXIT TO PREVIOUS MENU [DEFAULT]

METAFILE TYPE MENU OPTION 1: VECTOR

Option 1 (VECTOR) will let you output DI-3000 metafiles. By using this path you will be presented with the following sequence of menus. First, you will be asked to enter the metafile file name, followed by the entering of a filename to be given to the VAX command file.

ENTER THE NAME OF THE FILE CONTAINING THE METAFILE:
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

ENTER A NAME FOR THE COMMAND FILE [PLOTJOB.COM]:

If carriage return (<CR>) is entered for the command file, then the default file name PLOTJOB.COM will be used. Next will be a series of questions which relate to CDC/NOS job information. Answer each question as it is displayed.

ENTER NOS USER NUMBER

ENTER NOS PASSWORD

ENTER NOS CHARGE NUMBER

ENTER NOS MACHINE CODE (A/D/Y/Z)

ENTER VAX USER NAME

ENTER VAX PASSWORD

ENTER NOS DELIVERY DESTINATION

ENTER YOUR NAME

You will then be asked to enter a NOS filename, of seven or less characters, to be given to the metafile on the NOS machine.

ENTER NOS METAFILE FILE NAME

The next menu displays a list of available vector hardcopy devices.

VECTOR METAFILE: HARDCOPY DEVICE MENU

1. CALCOMP, 11 INCH
2. CALCOMP, 34 INCH
3. VARIAN, ROLL
4. VARIAN, FANFOLD
5. COLOR VERSATEC, (VER,39)
6. CELCO 16MM SLIDES, (CEL,16)
7. CELCO 8X10 VIEWGRAPHS, (CEL,8)
8. NO PLOTTING DESIRED [DEFAULT]

Upon selecting a hardcopy device, if appropriate, you will be asked to enter plotcard parameters and plot operator instructions. See ACD PRODUCTION PLOTTER DEVICE DRIVER GUIDE for further information.

ENTER ANY PLOT CARD PARAMETERS AT
THIS TIME, INCLUDE PARENTHESIS.

EX: (HEIGHT=11.,WIDTH=7.,YO=.5,XO=.5,RVAX,PAGE=8)

ENTER ANY SPECIAL PLOTTING INSTRUCTIONS TO BE
PLACED IN CONTINUATION LINES.
MAX OF 20 LINES WITH 40 CHARS PER LINE.
ENTER <CR> WHEN FINISHED.

You will then be asked to enter a NOS filename to contain dayfile information:

ENTER NOS DAYFILE NAME (UP TO 7 CHARS):

This will be followed by selection of default metafile processing commands, which will process all frames, or enter your own metafile commands:

DO YOU WISH DEFAULT METAFILE COMMANDS
FOR PROCESSING? [Y]/N

Upon completion of this final question your job and metafile are submitted automatically for processing and the following message will be displayed:

DATA IS ON FILE [FILE NAME] WHICH IS AUTOMATICALLY
FLINKED TO THE CYBERS FOR PROCESSING. YOUR DAYFILE AND
METAFILE WILL RESIDE ON THE MACHINE YOU CHOSE UNDER
YOUR CHOSEN NAME. THE DAYFILE WILL BE FLINKED BACK TO

THE VAX AS [DAYFILE NAME].LOG UPON COMPLETION AND CAN BE CHECKED BY YOU. BE SURE TO DELETE THIS .LOG FILE WHEN YOU ARE DONE. LIKEWISE THE DAYFILE ON NOS SHOULD BE DELETED AT YOUR CONVENIENCE. SINCE THE METAFILE RESIDES ON NOS, YOUR VAX VERSION CAN BE DELETED. ANY FUTURE NEED OF THE METAFILE, SIMPLY USE FLINK TO RETRIEVE THE FILE FROM NOS. VAX FILES [FILE NAME].COM AND [FILE NAME].DAT SHOULD BE DELETED WHEN PROCESSING IS COMPLETED.

METAFILE TYPE MENU OPTION 2: RASTER

Option 2 (RASTER) will let you generate hardcopy output of Raster Metafile format files. By using this path you will be presented with the following sequence of menus. First, you will be asked to enter the metafile file name, followed by a file name to be given to the command file.

ENTER THE NAME OF THE FILE CONTAINING THE METAFILE:
(ENTER FULL PATHNAME IF FILE IS NOT IN CURRENT DIRECTORY.)

ENTER A NAME FOR THE COMMAND FILE [PLOTJOB.COM]:

If carriage return (<CR>) is entered for the command file, then the default file name PLOTJOB.COM will be used. Next will be a series of questions which relate to CDC/NOS job information. Answer each question as it is displayed.

ENTER NOS USER NUMBER

ENTER NOS PASSWORD

ENTER NOS CHARGE NUMBER

ENTER NOS MACHINE CODE (A/D/Y/Z)

HARDCOPY METAFILE 5

ENTER VAX USER NAME

ENTER VAX PASSWORD

ENTER NOS DELIVERY DESTINATION

ENTER YOUR NAME

The next menu lists available raster hardcopy devices.

RASTER METAFILE: DEVICE DRIVER LIST

1. CELCO
2. VERSATEC
3. EXIT TO PREVIOUS MENU [DEFAULT]

Select a device. Selecting option 1 (CELCO) will result in the following additional menu. Select the type of output desired.

CELCO OUTPUT: OUTPUT TYPE

1. 8 X 10 (VIEWGRAPH)
2. 16MM (MOVIE)
3. 35MM (SLIDES)
4. EXIT TO PREVIOUS MENU [DEFAULT]

Option 2 (VERSATEC) can be selected only after the Raster Metafile format file has been converted to Versatec format. This converstion is accomplished by running the Raster Metafile Translator Versatec device driver found in the DISPLAY METAFILES processor.

After selecting the output device, you will then be asked to enter a NOS filename, of seven of less characters, to be given to the metafile on the NOS machine.

ENTER NOS METAFILE FILE NAME

The next question is a request for you to enter a NOS filename for dayfile information.

ENTER NOS DAYFILE NAME (UP TO 7 CHARACTERS):

Upon completion of this final question, your job and metafile are submitted for processing.

METAFILE TYPE MENU OPTION 3: EXIT

Option 3 (EXIT) will return you to the IMAT EXECUTIVE: GRAPHICS MENU.

SECTION 5

FINITE ELEMENT

IMAT Interfaces to Finite Element Application Codes - - - 5- 1

MSC/NASTRAN Processors

LOAD BULK DATA - - - - - - - - - - - - - - - - - - 5-10
FORMAT BULK DATA - - - - - - - - - - - - - - - - - 5-12
EXECUTE MSC/NASTRAN - - - - - - - - - - - - - - - 5-15
LOAD ANALYSIS RESULTS - - - - - - - - - - - - - - 5-17
RECOVER PHYSICAL OUTPUT - - - - - - - - - - - - 5-19

SUPERTAB Processors

EXECUTE SUPERTAB/GEOMOD - - - - - - - - - - - - 5-33
LOAD UNIVERSAL FILE - - - - - - - - - - - - - - - 5-34
FORMAT UNIVERSAL FILE - - - - - - - - - - - - - 5-37

Finite element models are usually created by an IMAT user with the aid of SDRC's preprocessor code SUPERTAB. The finished model data is then loaded into tables called relations in an IMAT-defined RIM database and an MSC/NASTRAN data deck is generated for performing a Solution 3 Normal Modes analysis. Both SUPERTAB and MSC/NASTRAN are general-purpose codes that support a variety of finite element types, analysis types, and analysis data types. The IMAT database and the IMAT processor codes, which were specifically developed to solve dynamics and control problems of large space-based structures, do not support all of the features of the commercial application codes. The following sections describe the finite element features that are supported and the correspondence of data stored in the database to data used in the application codes.

Coordinate Systems

Multiple coordinate systems are **not** supported in IMAT. All data containing coordinate locations refer to a single global XYZ coordinate system. In MSC/NASTRAN, this system is referred to as the **basic** coordinate system.

Finite Elements

Table 1 lists the finite elements currently supported by IMAT and the corresponding SUPERTAB and MSC/NASTRAN elements. MSC/NASTRAN physical property cards are shown in parentheses.

Two-dimensional triangular elements are stored in relation TRIANGLS and quadrilateral elements are stored in relation QUADS. Both relations contain connectivity descriptions, thickness, and non-structural mass per unit area. Therefore there is no separate physical property relation for these elements.

Lumped mass elements are stored in relation CONMASS. Attributes describing the lumped masses include the element number, node number, mass, and mass moments of inertia. Note that the mass moments of inertia (Mass Inertia Matrix) must be carefully specified in the correct order in SUPERTAB. The center of mass of a lumped mass element may be offset from the grid point (node) location by specifying x, y and z offsets. Remember, IMAT does not support multiple coordinate systems, so the xyz offsets must refer to the global (basic) coordinate system (coordinate system number 0 in SUPERTAB). Table 2 shows the attributes stored in CONMASS and the corresponding parameters in SUPERTAB and MSC/NASTRAN.

Data handling for beam elements is much more complicated than for any other type of supported element. IMAT supports two types of two-node elastic elements - rod elements, which have only axial and torsional stiffness, and linear prismatic beam elements, which have axial, torsional, shear and bending stiffness. (Rod element data is stored in the same relations as beam data. However, rod elements are not discussed in detail here.)

The IMAT database schema contains three relations for storing beam data. The BEAMS relation contains connectivity data such as node numbers, end releases, and shear center offsets, that are arranged in columns called attributes. Each row of information in the BEAMS relation contains the values of all attributes for one beam element. The BEAMPROP relation contains physical property attributes such as cross-sectional areas and moments of inertia. Each row of the BEAMPROP relation contains the attributes that describe one physical property entry. The BEAMREF relation contains beam orientation data in the form of either reference nodes or incremental vectors. Any number of beams may refer to a single entry (row) in the BEAMPROP or BEAMREF relations. Appendix A contains a complete listing of all attributes and relations. Note that the BEAMPROP relation contains some attributes that are not currently used in IMAT. Table 3 shows the connectivity, orientation and physical property attributes stored in the database, and their corresponding parameters in SUPERTAB and MSC/NASTRAN.

Material Properties

All material property data is stored in relation MAT-PROP, which contains attributes describing linear, elastic, homogeneous, isotropic materials. MAT-PROP contains some attributes that are not currently used by IMAT. The features that are currently supported are those that are compatible with MSC/NASTRAN's MAT1 material property card. Note that Poisson's ratio is not stored in MAT-PROP; however, any two of the three parameters E (modulus of elasticity), G (modulus of rigidity) and NU (Poisson's ratio) may be specified in SUPERTAB. The LOAD UNIVERSAL FILE processor will compute E or G, if it is not specified, by using the relation G = E/(2(1+NU)).

Loads and Constraints

The IMAT database schema contains tables (relations) for storing uniform pressures on plate elements (relation ELEMLOAD) and concentrated loads at nodes (relation NODELOAD). Dynamic load descriptions are not stored. However the RECOVER PHYSICAL OUTPUT processor will automatically generate the required MSC/NASTRAN dynamic load cards for use in the recovery of physical data from a state-space simulation. (See the description of the IMAT solution procedure).

Single point constraints (nodal restraints) are stored in the CONSTRN relation. A Y (for restrained) or an N (for unrestrained) is entered for each degree of freedom at the restrained nodes. Only one restraint case and one load case may be specified for a database.

Multipoint constraints are supported in the form of "rigid bar elements" stored in relation RIGIDBAR. These elements are equivalent to MSC/NASTRAN's RBAR "element". A rigid bar element consists of two nodes. Each node may have independent and/or dependent degrees of freedom associated with it. Note that SUPERTAB's "fore end" and "aft end" correspond to "end A" and "end B", respectively, in the IMAT-defined database schema, and "GA" and "GB" in MSC/NASTRAN. MSC/NASTRAN uses the RBAR descriptions to internally generate multipoint constraint equations.

Analysis Data

IMAT currently stores only normal modes analysis data in the database. Natural frequencies are stored in relation EIGNVALS and eigenvectors (mode shapes/slopes) are stored in relation EIGNVECT. In addition, a rapid-access binary file containing only eigenvector data is used to facilitate processing of linear system matrices and to provide an efficient method of reviewing eigenvector data (see QUERY EIGENVECTOR FILE processor). Rigid-body mass properties are loaded into relation RIGPROP from a binary MSC/NASTRAN OUTPUT2 file that contains the output from the Grid Point Weight Generator. Table 4 shows the correspondence between the Grid Point Weight Generator output and the data contained in the RIGPROP relation. Note that the signs of the off-diagonal inertia matrix terms are changed by the LOAD ANALYSIS RESULTS processor to ensure that the mass moment of inertia matrix (attribute MASMOMIN) is in inertia tensor form. An alter that writes the rigid-body mass information to the OUTPUT2 file is automatically generated by the FORMAT BULK DATA processor.

It is the user's responsibility to save certain files after performing a Solution 3 Normal Modes analysis with MSC/NASTRAN. A PUNCH file containing the checkpoint dictionary and the eigenvectors must be saved. The eigenvector (and eigenvalue) data is loaded from this file into the database. The Checkpoint Dictionary (contained in the PUNCH file) and the New Problem Tape file are required for performing restarts for open-loop modal analyses and to recover physical data from a closed-loop state-space simulation. The Grid Point Weight Generator Output, written to FORTRAN unit 11, must also be saved.

SUPERTAB[1]	IMAT	MSC/NASTRAN[2]
ROD	ROD	CROD (PROD)
LINEAR BEAM	BEAM	CBAR (PBAR)
PLN. STRS. LINEAR TRIA.	TM	TRIA3 (PSHELL)
FLAT PLT. LINEAR TRIA.	TB	"
THIN SHL. LINEAR TRIA.	TBM	"
PLN. STRS. LINEAR QUAD.	QM	QUAD4 (PSHELL)
FLAT PLT. LINEAR QUAD.	QB	"
THIN SHL. LINEAR QUAD.	QBM	"
LUMPED MASS	CONMASS	CONM2
RIGID BAR	RIGIDBAR	RBAR

Table 1 Finite elements supported by IMAT

SUPERTAB[1]	IMAT	MSC/NASTRAN[2]
ELEMENT NUMBER	ELEMENT	EID
NODE NUMBER	NODENUM	G
MA_MASS	MASS	M
MIM_MASS INER. MAT.(1)	I11	I11
" (2)	I22	I22
" (3)	I33	I33
" (4)	I21	I21
" (5)	I31	I31
" (6)	I32	I32
MO_MASS OFFSETS(1)	Z1M	X1
" (2)	Z2M	X2
" (3)	Z3M	X3

Table 2 Lumped mass attributes

SUPERTAB[1]	IMAT	MSC/NASTRAN[2]
GROUP NUMBER	CMPONENT	---
ELEMENT NUMBER	ELEMENT	EID
NODE 1 (fore end)	NODE1	GA
NODE 2 (aft end)	NODE2	GB
----	EL-TYPE	---
PHYS. PROP. NUM.	NOM-SIZE	PID
MAT. PROP NUM.	MATERIAL	MID
----	BMREFER	---
NSML	NONSTWHB	NSM
OFVF (1)	Z1A	W1A
" (2)	Z2A	W2A
" (3)	Z3A	W3A
OFVA (1)	Z1B	W1B
" (2)	Z2B	W2B
" (3)	Z3B	W3B
RFE	PA	PA
RAE	PB	PB
BOS[3] (1)	X	X1
" (2)	Y	X2
" (3)	Z	X3
" (1)	NODEREF	GO
IZZ	I1	I1
IYY	I2	I2
1/SRY[4]	ALPHA1	K1
1/SRZ[4]	ALPHA2	K2
AR	AREA	A
TC	FF	J
IXY	I12A	I12

Table 3 Beam attributes

MSC/NASTRAN Grid Point Weight Generator	IMAT (RIGPROP)
MO (1,1)	MASSMAT (1,1) (row 1)
" (1,2)	" (1,2) " 2
" (1,3)	" (1,3) " 3
" (2,1)	" (2,1) " 4
" (2,2)	" (2,2) " 5
" (2,3)	" (2,3) " 6
" (3,1)	" (3,1) " 7
" (3,2)	" (3,2) " 8
" (3,3)	" (3,3) " 9
I(S) (1,1)	MASMOMIN (1,1) (row 1)
- I(S) (1,2)	" (1,2) " 2
- I(S) (1,3)	" (1,3) " 3
- I(S) (2,1)	" (2,1) " 4
I(S) (2,2)	" (2,2) " 5
- I(S) (2,3)	" (2,3) " 6
- I(S) (3,1)	" (3,1) " 7
- I(S) (3,2)	" (3,2) " 8
I(S) (3,3)	" (3,3) " 9
X-CG	CG (1)
Y-CG	CG (2)
Z-CG	CG (3)

Table 4 Rigid-body mass attributes

NOTES:

1. SDRC I-DEAS SUPERTAB USER'S MANUAL, Level 4

2. MSC/NASTRAN USER'S MANUAL, Version 65c

3. SUPERTAB allows either a node or a set of xyz offsets to be used as an orientation specification. IMAT supports the following orientation codes and specifications:

CODE	SPECIFICATION
1	x, y, z offset
3	reference node
4	x, y, z offset

4. Note that the Shear Stiffness Factors (SRY and SRZ) specified in SUPERTAB are the inverse of the factors (K1 and K2) that MSC/NASTRAN uses. The IMAT LOAD UNIVERSAL FILE processor will automatically invert SRY and SRZ before storing them as ALPHA1 and ALPHA2.

PROCESSOR: LOAD BULK DATA

PURPOSE: The LOAD BULK DATA processor transfers bulk data from a fixed format MSC/NASTRAN bulk data deck to a user specified IMAT-defined RIM database.

ENTERING THE NAMES OF THE DATABASE AND BULK DATA FILE:

You will be asked to enter the names of the database and the bulk data file to be used with this processor. These files **must** reside in your current directory. The processor makes no attempt to copy files from other directories. Entering END at this point will cause you to exit from the processor and return to the IMAT EXECUTIVE.

LOADING THE DATA:

Data from the bulk data file will be loaded into the following IMAT-defined relations:

1. Grid points will be loaded into relation NODES.
2. MSC/NASTRAN elements will be loaded into relations BEAMS, QUADS, TRIANGLS, and RIGIDBAR.
3. Section properties for beams and rods will be loaded into relation BEAMPROP.
4. Beam orientations will be loaded into relation BEAMREF.
5. Material properties will be loaded into relation MAT-PROP.
6. Single point constraints will be loaded into relation CONSTRN.
7. Concentrated masses will be loaded into relation CONMASS.
8. Element pressure loads will be loaded into relation ELEMLOAD.
9. Concentrated forces and moments will be loaded into relation NODELOAD.

NASTRAN ELEMENTS vs. RIM ELEMENTS:

The processor will match MSC/NASTRAN element types with generic element types used in the IMAT relations and insert these element types in the appropriate connectivity relations.

LOAD BULK DATA 2

SUPPORTED BULK DATA CARDS:

NAME	FOUR CHARACTER ABBREVATION
CBAR	CBAR
CBEA	CBEA
CONM2	CONM
CONROD	CONR
CQUAD	CQUA
CROD	CROD
CSHEAR	CSHE
CTRIA	CTRI
FORCE	FORC
GRID	GRID
MAT1	MAT1
MOMENT	MOME
PBAR	PBAR
PLOAD	PLOA
PROD	PROD
PSHEAR	PSHE
RBAR	RBAR
SPC1	SPC1

ERROR HANDLING:

If a particular data set requested is not present in the bulk data file, a message is issued to that effect. The program will then continue by loading the next selection. If an error occurs during the processing (i.e., missing relation, error retrieving data, unmatched element type, etc.), the program will inform you of the error and give you the options of continuing to process the remaining data requested or returning to the IMAT EXECUTIVE.

PROCESSOR: FORMAT BULK DATA

PURPOSE: The FORMAT BULK DATA processor creates a complete MSC/NASTRAN Solution 3 data deck for use in computing the Normal Modes of a structure using the Lanczos method.

ENTERING THE FILE NAMES:

You will be asked to enter the name of the database to be used with this processor. The database **must** reside in your current directory. The processor makes no attempt to copy files from other directories. Entering END at this point will cause you to exit from the processor and return to the IMAT EXECUTIVE.

Next, you will be asked to enter the password for the database or a carriage return to indicate the default password, NONE. If an error occurs, you will be given a chance to enter a different database name.

You now will be asked to enter a name for the MSC/NASTRAN bulk data deck which the program will create for you. A carriage return will give the file the default name, NASBLK.

ENTERING THE MAXIMUM FREQUENCY OF INTEREST:

You will be asked to enter the maximum frequency of interest (Hz). EX: 2.0 All undamped natural frequencies below this value will be computed.

ERROR HANDLING:

If an error occurs at any stage of the processing (i.e., missing relation, error retrieving data, element mismatch, etc.), the program will inform you of the error and give you the option to continue processing or to quit. Should you choose to quit, you will return to the IMAT EXECUTIVE. The partial bulk data deck that was created will remain in your directory.

CHOOSING THE CONSTRAINED DEGREES OF FREEDOM:

This option allows you to delete rigid-body degrees of freedom (DOF) from the solution by specifying the deleted DOF, one at a time, using the following menu:

FORMAT BULK DATA: CONSTRAINED DEGREES OF FREEDOM MENU

CHOOSE THE CONSTRAINED DEGREES OF FREEDOM DESIRED:

1. X - DIRECTION
2. Y - DIRECTION
3. Z - DIRECTION
4. X - ROTATION
5. Y - ROTATION
6. Z - ROTATION
7. BEGIN PROCESSING

Select the options one at a time, or menu item 7 to continue with the processing. The program will write the appropriate GRDSET card to the bulk data deck.

PROCESSING THE DATA:

The program now proceeds to process the data in your database and create the bulk data deck. The data types will be processed in the following order:

1. Material property data from relations MAT-PROP and SCRREL
2. Joint location data from relation NODES
3. Node constraint data from relation CONSTRN
4. Element connectivity data from relations SCRREL and BEAMREF
5. Sectional property data from relations SCRREL and BEAMPROP
6. Concentrated mass data from relation CONMASS

PROCESSOR: EXECUTE MSC/NASTRAN

PURPOSE: The EXECUTE MSC/NASTRAN processor will generate a submit file for MSC/NASTRAN.

ENTERING THE NAME OF YOUR MSC/NASTRAN INPUT FILE:

The program will ask you to enter the name of your MSC/NASTRAN input file. The MSC/NASTRAN deck **must** reside in your current directory. The processor makes no attempt to copy files from other directories. If you enter a carriage return for any item, the program will give you a chance to exit the processor or to reenter that item.

ENTERING MSC/NASTRAN OPTIONS :

The program will display the following menu :

EXECUTE MSC/NASTRAN: OPTIONS MENU

CHOOSE THE OPTIONS DESIRED FOR THIS RUN
(NOTE: THE PLOT FILE AND OUTPUT FILE ARE SAVED FOR EVERY RUN)

1. USE OLD PROBLEM TAPE FOR RESTART RUN
2. USE UTILITY FILES FOR RESTART RUN
3. SAVE RESULTING UTILITY FILES
4. CONTINUE PROCESSING [DEFAULT]

Option 1 and option 2 allow you to use an Old Problem Tape (OPTP) and/or utility files that were created by a previously executed MSC/NASTRAN analysis. Option 3 provides the capability to save the utility files created by this analysis. You will be asked to enter names for each of the files you want to save. Option 4 will allow you to enter the required information for the MSC/NASTRAN execution command.

VERIFYING THE SUBMIT FILE:

The program will display the information and ask if it is correct. If you indicate that any of the information is incorrect, the program will give you the opportunity to reenter the information. If you choose not to reenter the information, you will return to the IMAT EXECUTIVE.

SUBMITTING THE BATCH FILE:

The processor will submit your MSC/NASTRAN job to the NAS$BATCH queue by default. If you want to submit your job to a different queue, you may indicate the desired queue at the prompt:

ENTER THE QUEUE TO WHICH THE NASTRAN JOB IS TO BE SUBMITTED [DEFAULT=NAS$BATCH]

TERMINATION:

After you specify the appropriate queue, your MSC/NASTRAN job will be placed on that queue for execution. The processor will ask if you want to submit more MSC/NASTRAN jobs and if not, the processor will return you to the IMAT EXECUTIVE.

PROCESSOR: LOAD ANALYSIS RESULTS

PURPOSE: The LOAD ANALYSIS RESULTS processor transfers analysis data from MSC/NASTRAN to a user specified IMAT-defined RIM database.

ENTERING THE NAME OF THE DATABASE:

You will be asked to enter the name of the database to be used with this processor. The database **must** reside in your current directory. The processor makes no attempt to copy files from other directories. Entering END at this point will cause you to exit from the processor and return to the IMAT EXECUTIVE.

Next, you will be asked to enter the password for the database or a carriage return to indicate the default password, NONE. If an error occurs, you will be given a chance to enter a different database name.

CHOOSING AND LOADING THE DATA:

The program will display:

LOAD ANALYSIS RESULTS: ANALYSIS DATA MENU

1. STRAIN ENERGIES
2. RIGID PROPERTIES
3. VIBRATIONAL EIGENVECTORS
4. BEGIN LOADING

If you choose option 1 and/or option 3, you will be asked to enter the name of the PUNCH file that contains the corresponding data. This file **must** also reside in your current directory. Enter option 4 after you have chosen all the data types desired.

ANALYSIS DATA MENU OPTION 2: RIGID PROPERTIES

If option 2 (RIGID PROPERTIES) is selected, you will be prompted for the name of the (binary) output file that contains the rigid body mass properties. (The FORMAT BULK DATA processor will write the alter needed to obtain this information from MSC/NASTRAN's Grid Point Weight Generator.)

ANALYSIS DATA MENU OPTION 3: VIBRATIONAL EIGENVECTORS

Enter the number of eigenvectors that you desire to load into the database. This option allows you to load, for example, the first n eigenvectors into the database.

NOTE: The eigenvectors must have been normalized such that the generalized mass is unity for each mode. MSC/NASTRAN's Lanczos method automatically performs this "mass normalization".

ERROR HANDLING:

If an error occurs during the processing (i.e., missing relation, error retrieving data, unmatched element type, etc.), the program will inform you of the error and give you the option of continuing to process the remaining data requested or returning to the IMAT EXECUTIVE.

PROCESSOR: RECOVER PHYSICAL OUTPUT

PURPOSE: The RECOVER PHYSICAL OUTPUT processor enables a user to recover physical output using MSC/NASTRAN from a state-space time simulation calculated using MATRIXx/SYSTEM_BUILD.

MAIN MENU OPTIONS:

When you first enter the RECOVER PHYSICAL OUTPUT processor, you will see the MAIN MENU which offers options to let you generate a complete bulk data deck, plot commands only, or a MSC/NASTRAN modal solution (UHV) matrix. To exit the program at any time, enter Q from this or any menu:

RECOVER PHYSICAL OUTPUT: MAIN MENU

CHOOSE THE FUNCTION DESIRED OR "Q" TO QUIT:

1. GENERATE COMPLETE BULK DATA DECK [DEFAULT]
2. INSERT NEW PLOT COMMANDS INTO EXISTING DECK
3. GENERATE NEW PLOT COMMANDS ONLY
4. GENERATE MSC/NASTRAN UHV MATRIX FROM MATRIXX MODAL OUTPUT FILES

MAIN MENU OPTION 1: GENERATE COMPLETE BULK DATA DECK

Option 1 will generate a new MSC/NASTRAN data deck for a modified Solution 31 transient analysis restart.

LOCATING YOUR DATA:

The program will first prompt you to enter the name of your database. If your database is not in your current default directory, you will be prompted to enter the pathname to the directory where the database is located. Enter a VMS pathname for the directory, not including the file name (EX: DUA0:[MYDIR.SUBDIR]).

RECOVER PHYSICAL OUTPUT 2

The program will now display a menu of the control simulations which have been recorded in your database (See RECORD CONTROL SIMULATION processor)

RECOVER PHYSICAL OUTPUT: SIMULATIONS MENU

FOR WHICH SIMULATION SHOULD THIS DECK BE CREATED?

	SIMUL.ID	SYSTEM ID	DESCRIPTION
1.	1	1	Description entered for simulation 1
2.	2	1	Description entered for simulation 2
3.	3	...	...

After you select the simulation desired, the RECOVER PHYSICAL OUTPUT processor will present a summary of the simulation and let you verify that this is the simulation intended.

The program will now attempt to locate the PUNCH file containing the checkpoint dictionary from the normal modes analysis and load time history files recorded in your database for this simulation. If a file cannot be located in the current directory, you will be prompted to enter an alternate pathname for the file.

(EX: DUA0:[MYDIR.MYSUB]MYFILE.DAT)

The program will allow you to continue generating a restart deck if the PUNCH file is missing; however, you must be able to locate time history files for all loads defined in the simulation record.

The RECOVER PHYSICAL OUTPUT processor will now present a summary of the modes used in the control simulation compared to the modes calculated in the finite element normal modes analysis.

INITIALIZING THE CASE CONTROL DECK:

RECOVER PHYSICAL OUTPUT 3

The RECOVER PHYSICAL OUTPUT processor will now generate the initial commands for the MSC/NASTRAN Executive Case Control Decks and will prompt you to enter a title and a subtitle for this analysis (maximum of 70 characters each). If you answer YES to the DMAP option, your MSC/NASTRAN output will contain a DMAP listing. You will also be prompted to enter the set ID that was used in the EIGR or EIGRL card in the normal modes analysis, and if your database contains single point constraints, the program will also prompt you for the SPC set number.

DEFAULT PLOT PARAMETERS:

The program will now display the default plot parameters which will apply to all plot commands generated by this processor:

RECOVER PHYSICAL OUTPUT: PLOT PARAMETER MENU

CHOOSE ANY OF THE CURRENT DEFAULT PLOT PARAMETERS WHICH YOU WOULD LIKE TO CHANGE. THESE PARAMETERS WILL APPLY TO ALL PLOTS GENERATED.

1. XPAPER = 20.0
2. YPAPER = 20.0
3. XAXIS = YES
4. YAXIS = YES
5. XVALUE SKIP PRINT = 0
6. YVALUE SKIP PRINT = 0
7. LOWER TICS = 1
8. UPPER TICS = 0
9. LEFT TICS = 1
10. RIGHT TICS = 0
11. XDIVISIONS = 10
12. YDIVISIONS = 10
13. XGRID LINES = YES
14. YGRID LINES = YES
15. XTITLE = TIME, SECONDS
16. ACCEPT CURRENT PARAMETERS

Choose the parameters that you wish to change one at a time, then select option 16 to accept the parameters you have defined. These parameters, summarized below, are fully explained in the MSC/NASTRAN USER'S MANUAL (Vol. II, 4.3.2):

XPAPER	Plot paper width
YPAPER	Plot paper height

XAXIS	Request to plot X-axis (YES/NO)
YAXIS	Request to plot Y-axis (YES/NO)
XVALUE SKIP PRINT	Number of X-values to skip between labelled tic marks
YVALUE SKIP PRINT	Number of Y-values to skip between labelled tic marks
LOWER TICS	Request for tic marks on lower edge of frame
UPPER TICS	Request for tic marks on upper edge of frame
LEFT TICS	Request for tic marks on left edge of frame
RIGHT TICS	Request for tic marks on right edge of frame
XDIVISIONS	Number of uniform spaces in X-direction
YDIVISIONS	Number of uniform spaces in Y-direction
XGRID LINES	Request for grid lines parallel to Y-axis (YES/NO)
YGRID LINES	Request for grid lines parallel to X-axis (YES/NO)
XTITLE	Label for X-axis

PLOT REQUEST MENU:

The RECOVER PHYSICAL OUTPUT processor will now display the PLOT REQUEST MENU:

RECOVER PHYSICAL OUTPUT: PLOT REQUEST MENU

ENTER THE PLOT TYPES DESIRED FOR THIS ANALYSIS:

1. DYNAMIC LOAD PLOTS
2. MODAL DISPLACEMENT, VELOCITY, ACCELERATION PLOTS
3. PHYSICAL DISPLACEMENT, VELOCITY, ACCELERATION PLOTS
4. ELEMENT FORCE, STRESS PLOTS
5. CONTINUE PROCESSING [DEFAULT]

Select options 1 through 4 to generate the plot commands desired for the MSC/NASTRAN data deck, followed by option 5 which will continue processing by completing the final commands required for the deck. This menu will also display the total number of plots generated as you complete each menu choice.

PLOT REQUEST MENU CHOICE 1: DYNAMIC LOAD PLOTS

This option will allow you to define plots of applied loads at selected nodes or ranges for nodes. The same information will be plotted for all nodes in each range you specify. For each node or range of nodes you enter (a), you will be asked to define the components (b), and the scaling desired (c) for the plots.

(a) The program will first prompt you to enter a node or range of nodes for plots. Enter an individual node number or a range of nodes (EX: 200 or 200-201), or a carriage return if you have defined all load plots desired and wish to return to the PLOT REQUEST MENU.

(b) For each set of plots you request, you will be asked to enter the components desired:

RECOVER PHYSICAL OUTPUT: COMPONENT MENU

1. TRANSLATION-X
2. TRANSLATION-Y
3. TRANSLATION-Z
4. ROTATION-X
5. ROTATION-Y
6. ROTATION-Z

ENTER COMPONENTS DESIRED (1-6) FOR LOAD PLOTS FOR NODES 200-201. (SEPARATE EACH COMPONENT OR RANGE WITH COMMAS.)
EX: 1,3-5 (OR) 1,2,3

Enter the components desired separated by commas. You may indicate a range with a dash.

(c) After you have selected the components for this set of plots, the program will ask you to define the scaling desired:

DO YOU WANT MSC/NASTRAN TO PROVIDE AUTOMATIC SCALING OF DATA? [Y]/N

If you choose MSC/NASTRAN automatic scaling (Y), there will be no minimum or maximum X or Y value set. If you decide to scale the data yourself (N), you will be shown the X/Y MINIMUM/MAXIMUM VALUES MENU:

RECOVER PHYSICAL OUTPUT: X/Y MINIMUM/MAXIMUM VALUES

CHOOSE OPTIONS 1-4 TO CHANGE CURRENT DEFAULTS:

1. XMIN = NONE
2. XMAX = NONE
3. YMIN = NONE
4. YMAX = NONE
5. ACCEPT ABOVE DEFAULTS [DEFAULT]

Choose one at a time the minimum or maximum value you want to change. Enter a new value at the prompt or a carriage return to indicate no minimum or maximum:

NEW VALUE FOR XMIN [<CR> = 'NONE']:

The program will show you the menu again, reflecting the current values you have entered. When you are satisfied with the minimum/maximum values you have set, select option 5.

The program will now display a table of the plots you have selected for these nodes:

PLOTS REQUESTED FOR NODES 200-201:
COMPONENTS 1 2 3

XMIN:	-.100E+03	XMAX:	+.100E+03
YMIN:	NONE	YMAX:	NONE

GENERATE THE PLOTS LISTED ABOVE? [Y]/N

If you agree with the plots chosen (Y), the program will write all of the plot commands necessary to generate these plots. If you want to make changes to the plots requested (N), the program will return you to the prompt for node numbers to define a different set of plots.

From this point on, whenever you enter a new node or range of nodes for plots, the program will display the current default plots which are the set of plots chosen for the last node range. You may choose these same plots for the new range or node, or define a different set. Enter a carriage return at the prompt for nodes to complete all load plots and return to the PLOT REQUEST MENU.

PLOT REQUEST MENU CHOICE 2: MODAL DISPLACEMENT, VELOCITY, ACCELERATION PLOTS

This option will allow you to define displacement, velocity, and acceleration plots for modes or ranges of modes. The same information will be plotted for all modes in each range you specify. For each mode or range of modes you enter (a), you will be asked to define the output types (b), the components desired for each output type (c), and the scaling desired (d) for the plots.

(a) The program will first prompt you to enter a mode or range of modes for plots. Enter an individual mode number or a range of modes (EX: 10 or 6-10), or a carriage return if you have defined all modal plots desired and wish to return to the PLOT REQUEST MENU.

(b) You will now see the MODAL PLOT OUTPUT MENU:

RECOVER PHYSICAL OUTPUT: MODAL PLOT OUTPUT MENU

ENTER OUTPUT TYPES DESIRED ONE AT A TIME FOR MODES 6 - 10:

1. DISPLACEMENT
2. VELOCITY
3. ACCELERATION
4. CANCEL PLOTS FOR THESE MODES
5. CONTINUE PROCESSING [DEFAULT]

Choose one at a time the output types desired for these modes. Select option 5 to continue processing after you have chosen the output types desired. Option 4 will cancel all the plots you have selected for these modes and return you to the prompt to enter a new mode range (a).

(c) For each output type you request, you will be asked to enter the components desired:

RECOVER PHYSICAL OUTPUT: COMPONENT MENU

1. TRANSLATION-X
2. TRANSLATION-Y
3. TRANSLATION-Z
4. ROTATION-X
5. ROTATION-Y
6. ROTATION-Z

ENTER COMPONENTS DESIRED (1-6) FOR DISPLACEMENT PLOTS FOR MODES 6 - 10. (SEPARATE EACH COMPONENT OR RANGE WITH COMMAS.)

EX: 1,3-5 (OR) 1,2,3

Enter the components desired separated by commas. You may indicate a range with a dash. You may change the components selected for an output type by reselecting that type from the MODAL PLOT OUTPUT MENU and choosing different components (<CR> for NONE) as the program will record only the last set of components selected for each output type for any set of modes.

(d) After you have selected all of the output types for these modes (and entered option 5 from the MODAL PLOT OUTPUT MENU), the program will ask you to define the scaling desired for this set of plots:

DO YOU WANT MSC/NASTRAN TO PROVIDE AUTOMATIC SCALING OF DATA? [Y]/N

If you choose MSC/NASTRAN automatic scaling (Y), there will be no minimum or maximum X or Y values set. If you decide to scale the data yourself (N), you will be shown the X/Y MINIMUM/MAXIMUM VALUES MENU and be asked to set X and Y minimum and maximum values by the same method used with Load plots (See section c under LOAD PLOTS above).

The program will now display a table of the plots you have selected for these modes:

PLOTS REQUESTED FOR MODES 6 - 10:
DISPLACEMENT 1 2 3
VELOCITY -
ACCELERATION 1 2 4 5 6

XMIN:	.100E+03	XMAX:	+.100E+03
YMIN:	NONE	YMAX:	NONE

GENERATE THE PLOTS LISTED ABOVE? [Y]/N

If you agree with the plots chosen (Y), the program will write all of the plot commands necessary to generate these plots. If you want to make changes to the plots requested (N), the program will return you to the MODAL PLOT OUTPUT MENU to define a different set of plots.

From this point on, whenever you enter a new mode or range of modes for plots, the program will display the current default plots which are the set of plots chosen for the last mode range. You may choose these same plots for the new range or modes, or define a different set. Enter a carriage return at the prompt for modes or mode range to complete all modal plots and return to the PLOT REQUEST MENU.

PLOT REQUEST MENU CHOICE 3: PHYSICAL DISPLACEMENT, VELOCITY, ACCELERATION PLOTS

This option will allow you to define displacement, velocity, and acceleration plots for nodes or ranges of nodes. The same information will be plotted for all nodes in each range you specify. The set of menus for physical plots exactly parallels those under PLOT REQUEST MENU CHOICE 2 (MODAL PLOTS):

PHYSICAL PLOT OUTPUT MENU (to select output types)
COMPONENT MENU (to select components desired for each output type)
X/Y MINIMUM/MAXIMUM VALUES MENU (to define minimum or maximum X or Y values)

(See the documentation for modal plots above for specific instructions.) Enter a carriage return at the prompt for nodes or ranges of nodes to complete all displacement plot requests and return to the PLOT REQUEST MENU.

PLOT REQUEST MENU CHOICE 4: ELEMENT FORCE, STRESS PLOTS

This option will allow you to define force and stress plots for specific elements or ranges of elements. The same information will be plotted for all elements in each range you specify. For each range you define (a), you will enter the element type (b), give an optional description of the element (c), choose force plots desired (d), choose stress plots desired (e), and have an opportunity to use MSC/NASTRAN automatic scaling or to define you own scaling for data (f).

(a) The program will first prompt you to enter an element number or range of element numbers.

(b) You will then be prompted to enter the MSC/NASTRAN element type for the element or element range:

RECOVER PHYSICAL OUTPUT: ELEMENT TYPE MENU

ENTER ELEMENT TYPE FOR ELEMENT 200:

RECOVER PHYSICAL OUTPUT 11

1. BAR
2. BEAM
3. ROD
4. QUAD4
5. TRIA3
6. CANCEL PLOTS FOR THESE ELEMENTS

Enter the MSC/NASTRAN element type or option 6 to cancel these plots.

(c) You will now be prompted to enter an optional description for this element or range of elements:

ENTER A DESCRIPTION FOR THIS ELEMENT (MAX 37 CHAR) [OPTIONAL]:

Any description you enter will be used in the TCURVE command for this set of plots. Enter a carriage return if you do not want to use a description.

(d) and (e) You will now be shown first a menu of the force plots supported for this element type and then a menu of the supported stress plots (See MSC/NASTRAN USER'S MANUAL, VOLUME II for a description of each plot type.)

EXAMPLE MENU:

RECOVER PHYSICAL OUTPUT: FORCE PLOTS FOR BAR ELEMENTS

ENTER PLOTS DESIRED ONE AT A TIME FOR ELEMENT 200:

1. BENDING MOMENT, PLANE 1 (END A)
2. BENDING MOMENT, PLANE 2 (END A)
3. BENDING MOMENT, PLANE 1 (END B)
4. BENDING MOMENT, PLANE 2 (END B)
5. SHEAR FORCE, PLANE 1
6. SHEAR FORCE, PLANE 2
7. AXIAL FORCE
8. TORQUE

9. CANCEL FORCE PLOTS SELECTED FOR THESE ELEMENTS
10. CANCEL ALL PLOTS SELECTED FOR THESE ELEMENTS
11. CONTINUE PROCESSING [DEFAULT]

As you enter the plots desired (one at a time), the program will mark your choices with an asterisk to the left of the option number. In this example menu, option 9 would cancel all of the force plots you have selected (i.e., clear all the asterisks) and allow you to select a different set of force plots. Option 10 would cancel all plots chosen for these elements and return you to the prompt to enter a new element range. Option 11 would, in this case, pass you on to the stress plot menu to complete the plot selection phase and continue to the MSC/NASTRAN automatic scaling option.

(f) After you have selected all of the plots desired, you will be given a chance to select MSC/NASTRAN automatic scaling or to define your own X or Y minimum or maximum values. (See the discussion of X/Y MINIMUM/MAXIMUM VALUES under PLOT REQUEST MENU CHOICE 2 (MODAL PLOTS)). The program will now write to the data deck all of the commands required to generate the element plots requested. Enter a carriage return at the request for element number or range, when you have completed all the element plots desired. You will return to the PLOT REQUEST MENU.

FINISHING THE BULK DATA DECK:

After you have selected all of the plots desired (Enter option 5 [CONTINUE PROCESSING] from the PLOT REQUEST MENU), the program will write the final commands appropriate to your MSC/NASTRAN data deck including the TABLED1 cards to be formatted from the controls time history files. To execute MSC/NASTRAN with this data deck, use the EXECUTE MSC/NASTRAN processor. This processor will give you the opportunity to retrieve any files necessary to perform a MSC/NASTRAN restart to complete this closed-loop transient analysis. Before executing MSC/NASTRAN, use OPTION 4 from the MAIN MENU of this program to create your MSC/NASTRAN modal solution (UHV) matrix from the MATRIXx modal output (state-space solution) files.

MAIN MENU OPTION 2: INSERT NEW PLOT COMMANDS INTO EXISTING DECK

This option will allow you to define new plot commands to be inserted into an existing deck previously created using option 1. A new version of your deck will be created in which the plot commands you define replace the plot commands in the original deck. Your original deck will not be altered. This option will only work with a deck created using option 1 from the main menu.

You will first be prompted to enter the name of your existing bulk data deck and a name for the new deck to be created. You will then be allowed to reset the default plot parameters (See DEFAULT PLOT PARAMETERS under option 1) and to define new plot commands (See PLOT REQUEST MENU under option 1) much as you would when generating a complete deck from the database. When you complete your plot choices, the plot commands you defined will be inserted into a copy of your original deck.

MAIN MENU OPTION 3: GENERATE NEW PLOT COMMANDS ONLY

Option 3 will allow you to generate plot commands only. This option does not require that you have a bulk data deck present. As in option 2, you will enter a file name for your plot commands and then will define plot parameters and plot commands as you do under option 1.

MAIN MENU OPTION 4: GENERATE MSC/NASTRAN UHV MATRIX FROM MATRIXX MODAL OUTPUT FILES

Option 4 will allow you to generate a MSC/NASTRAN modal solution (UHV) matrix from the three modal output files recorded in your database. You will first select the database and simulation. If you have previously selected a database and simulation, these will be the default choices. The program will now locate the MATRIXx modal output files defined in your database for this simulation. If the files are not located in the current directory, you will be given an opportunity to enter an alternate pathname for each file.

RECOVER PHYSICAL OUTPUT 14

The program will now create a MSC/NASTRAN modal solution (UHV) matrix from the modal output data in these files. When executing a MSC/NASTRAN restart using a bulk data deck from option 1, be certain to define this UHV matrix file as FORTRAN unit 15 (FOR015).

PROCESSOR: EXECUTE SUPERTAB/GEOMOD

PURPOSE: The EXECUTE SUPERTAB/GEOMOD processor allows the user to access the available SDRC I-DEAS modules. The user must be familiar with I-DEAS commands in order to execute this processor.

ENTERING YOUR TERMINAL TYPE:

At the beginning of execution, I-DEAS will prompt you for your terminal type. Enter the correct code for your graphics terminal or an M to see a menu of terminal types supported.

ON-LINE HELP:

ON-line help is featured as a global command in I-DEAS. For further information, consult the I-DEAS USER'S GUIDE, or enter the HELP command from within I-DEAS.

RETURNING TO THE IMAT EXECUTIVE:

Upon completion of I-DEAS, you will be returned to the IMAT EXECUTIVE.

PROCESSOR: LOAD UNIVERSAL FILE

PURPOSE: The LOAD UNIVERSAL FILE processor reads the Universal file created during a SUPERTAB session and loads the data into an IMAT-defined RIM database. This database contains the complete description of a particular finite element model. The processor supports the following data set types:

DATA SET TYPE	DESCRIPTION
15	Node Data
71	Connectivity Data
731	Physical Property Value Entries
747	Material Property Value Entries
752	Permanent Group Data
755	Restraint Data
756	Load Set Data

ENTERING YOUR DATABASE NAME:

You will be asked to enter the name of the database to be used with this processor. The database **must** be in your current directory. The processor makes no attempt to copy databases from other directories. Entering END at this point will allow you to exit the processor and return to the IMAT EXECUTIVE.

You will next enter the password for the database or a carriage return to indicate the default password, NONE. The program will now open your database and if an error occurs, you will have the chance to enter a different database name or exit the processor.

ENTERING THE UNIVERSAL FILE NAME:

You will be asked to supply the name of the SUPERTAB Universal file. The program will retrieve this file for you if it is not already in your current directory.

PROCESSOR LOG FILE:

Many descriptive messages concerning the data in your Universal file may be generated during the execution of LOAD UNIVERSAL FILE. The processor will give you the opportunity to keep a log file of these messages. If you choose to create a log file, you will be asked to enter a name for the log file or you may use the default name offered.

SELECTING DATA SET TYPES FROM THE DATA SET MENU:

The program will now show you:

LOAD UNIVERSAL FILE: DATA SET MENU

ENTER DATA SET TYPES TO BE TRANSLATED INTO RIM:

CHOICE	(DATA SET TYPE)	DESCRIPTION
1.	(15)	NODE DATA
2.	(71)	CONNECTIVITY DATA
3.	(731)	PHYSICAL PROPERTY VALUE ENTRIES
4.	(747)	MATERIAL PROPERTY VALUE ENTRIES
5.	(752)	PERMANENT GROUP DATA
6.	(755)	RESTRAINT DATA
7.	(756)	LOAD SET DATA
8.		ALL SUPPORTED DATA SET TYPES
9.		BEGIN PROCESSING

Enter option 8 if you would like to load all supported data set types into your database; otherwise choose the data set types you want one at a time, followed by option 9 to begin the processing.

PROCESSING THE DATA:

The processor will now read through the Universal data file and load each data set type requested. It will inform you of each data set found with one of three messages, as appropriate:

LOADING DATA SET TYPE xx

UNSUPPORTED DATA SET IN UNIVERSAL FILE IGNORED.
DATA SET TYPE: xx

NON-REQUESTED DATA SET IN UNIVERSAL FILE
IGNORED DATA SET TYPE: xx

If an error occurs during the processing (i.e., missing relation, unmatched element type, etc.), the program will inform you of the error but continue with the processing. After processing the Universal file, the program will remind you of the log file name if you chose to generate one, and then return you to the IMAT EXECUTIVE.

PROCESSING ORDER:

Connectivity Data (Data Set 71) requires property value data from Data Set 731 (Physical Property Value Entries) in order to be correctly loaded into the database. In addition, Permanent Group Data (Data Set 752) refers to element numbers loaded from Data Set 71. The LOAD UNIVERSAL FILE processor will not load either Data Set 71 or Data Set 752 unless its required predecessor was also loaded.

PROCESSOR: FORMAT UNIVERSAL FILE

PURPOSE: The FORMAT UNIVERSAL FILE processor formats finite element model and analysis data stored in an IMAT-defined RIM database into a SUPERTAB (4.0) Universal file. The processor supports the following data set types:

DATA SET TYPE	DESCRIPTION
15	Node Data
55	Data at Nodes
71	Connectivity Data
731	Physical Property Value Entries
747	Material Property Value Entries
752	Permanent Group Data
755	Restraint Data
756	Load Set Data

ENTERING YOUR DATABASE NAME:

You will be asked to enter the name of the database to be used with this processor. The database files **must** reside in your current directory. The processor makes no attempt to copy files from other directories. Entering END at this point will allow you to exit the processor. The program will attempt to open the database and allow you to reenter the information if an error occurs.

ENTERING THE UNIVERSAL FILE NAME:

The program will now prompt you to enter a name for the SUPERTAB Universal data file to be created. You may enter "filename.UNV" or just "filename" as the default subscript is ".UNV". You must enter the full name if you want a subscript other than ".UNV". Entering a carriage return will give you the default name of the database name with a file extension ".UNV".

PROCESSOR LOG FILE:

Many descriptive messages concerning the data in your Universal file may be generated during the execution of LOAD UNIVERSAL FILE. The processor will give you the opportunity to keep a log file of these messages. If you choose to create a log file, it will be given the same name as your Universal file but will end in ".LOG".

ASSIGNING COLORS:

FORMAT UNIVERSAL FILE: COLOR ASSIGNMENT MENU

YOU MAY CHOOSE TO ASSIGN COLOR BASED ON ELEMENT PROPERTIES OR NODE NUMBER, OR ACCEPT THE DEFAULT COLOR ASSIGNMENTS:
ELEMENTS/NODES - GREEN CONSTRAINTS - WHITE LOADS - BLUE

1. ASSIGN COLORS TO ELEMENTS
2. ASSIGN COLORS TO NODES
3. CONTINUE PROCESSING [DEFAULT]

You may select the default color assignments as shown or choose a basis for element or node color assignments to emphasize some particular aspect such as material or physical properties. By choosing option 1 (ASSIGN COLORS TO ELEMENTS), you will **not** make specific color assignments (i.e, red, green, blue, etc.), but will choose the basis for color assignments that the program will automatically generate for you as presented in the ELEMENT COLOR MENU.

FORMAT UNIVERSAL FILE: ELEMENT COLOR MENU

CHOOSE ONE ELEMENT PROPERTY TO BE THE BASIS FOR COLOR ASSIGNMENT:

1. PHYSICAL PROPERTY
2. MATERIAL PROPERTY
3. PERMANENT GROUP
4. ELEMENT TYPE
5. ELEMENT NUMBER RANGE
6. EXIT (USE DEFAULT ELEMENT COLOR)

By choosing option 2 (ASSIGN COLORS TO NODES) FROM THE COLOR ASSIGNMENT MENU, you will select the ranges of nodes which should be assigned different SUPERTAB colors:

SUPERTAB COLORS WILL BE ASSIGNED SEQUENTIALLY BASED ON THE NODE NUMBER RANGES YOU ENTER. ENTER ONE NUMBER RANGE PER LINE (MAX 20 RANGES). EX: 200 250

RANGE 1:

Enter up to 20 ranges (EX:51 100) followed by a final carriage return to return to the COLOR ASSIGNMENT MENU. When you have completed color assignments for elements and/or nodes, select option 3 (CONTINUE PROCESSING) from the COLOR ASSIGNMENT MENU.

SELECTING DATA SET TYPES FROM THE DATA SET MENU:

The program will now show you the DATA SET MENU:

FORMAT UNIVERSAL FILE: DATA SET MENU
ENTER DATA SET TYPES TO BE FORMATTED FROM RIM:

OPTION	(DATA SET TYPE)	DESCRIPTION
1	(15)	NODE DATA
2	(55)	DATA AT NODES
3	(71)	CONNECTIVITY DATA
4	(731)	PHYSICAL PROPERTY VALUE ENTRIES
5	(747)	MATERIAL PROPERTY VALUE ENTRIES
6	(752)	PERMANENT GROUP DATA
7	(755)	RESTRAINT DATA
8	(756)	LOAD SET DATA
9		ALL SUPPORTED DATA SET TYPES
10		BEGIN PROCESSING

Enter option 9 if you would like to format all supported data set types from your database; otherwise choose the data set types you want one at a time, followed by option 10 to begin the processing.

SELECTING MODES:

If you have chosen to format data set 55 (DATA AT NODES), you must now select which modes should be formatted from the eigenvector table (relation EIGNVECT in the IMAT database).

FORMAT UNIVERSAL FILE: MODE NUMBER MENU

WHICH MODES (CONDNUM) SHOULD BE FORMATTED FROM RELATION "EIGNVECT" FROM DATA SET 55 (DATA AT NODES)

1. ALL MODE NUMBERS
2. RANGE OF MODE NUMBERS

Select option 1 (ALL MODE NUMBERS) to format all modes or option 2 (RANGE OF MODE NUMBERS) to enter a specific range of modes desired. Processing time for a large number of modes may be significant. You will also be prompted to enter an optional RUN IDENTIFICATION (maximum of 80 characters) for data set 55.

PROCESSING THE DATA:

The processor will now read through the Universal data file and format each data set type requested. It will inform you of each data set as it attempts to format it. If an error occurs during the processing (i.e., missing relation, unmatched element type, etc.), the program will inform you of the error but continue with the processing.

EX: FORMATTING DATA SET 731

!!! RELATION "BEAMPROP" WAS NOT FOUND
UNABLE TO FORMAT BEAM PROPERTIES

After formatting all requested data sets, the program will remind you of the log file name if you chose to generate one, and then return you to the IMAT EXECUTIVE.

PROCESSING ORDER:

Element Property Value Entries (Data Set 731) must have been formatted in the Universal file in order to reference the properties in Data Set 71 (Connectivity Data). Also, Permanent Group Data (Data Set 752) refers to element numbers which must have appeared previously in Data Set 71. The FORMAT UNIVERSAL FILE processor keeps track of which data sets have been loaded and will only format data for Data Sets 71 and 752 for which all element and property references have been formatted previously in the Universal file.

COLOR TABLE:

After the processing is completed, the log file will contain a listing of the color scheme for the Universal file.

$$[m]\,\ddot{p} + [c]\,\dot{p} + [k]\,p = \bar{f} \qquad (2)$$

where [m], [c] and [k] are the r x r mass, damping and stiffness matrices, respectively, and p is an r-element vector of nodal displacements. The vector $\bar{f}$ contains applied time-dependent forces that can be functions of p and/or $\dot{p}$ when control forces are employed.

In order to reduce the number of degrees of freedom in the problem and decouple the equations of motion, the undamped natural modes of the system are used to define a generalized coordinate system. The natural modes are calculated by ignoring [c] and $\bar{f}$ in equation (2) and solving the resulting homogeneous equations for the first s mode shapes, ϕ_i (r-element vectors), and frequencies, ω_i ($i = 1,2,\ldots,s$).

$$[m]\,\ddot{p} + [k]\,p = 0 \qquad (3)$$

Mode i is called a rigid body mode if $\omega_i = 0.0$ and otherwise it is called a flexible mode. The computer codes which perform this solution (IMAT uses MSC/NASTRAN) may not return exact zeros for rigid body mode frequencies, but in practice these have been at least four orders of magnitude smaller than the smallest flexible mode frequency, so are easily identifiable. Usually there will be six rigid body modes although there can be fewer if rigid body degrees-of-freedom have been constrained during the solution process. Usually, each rigid body mode will incorporate components of translation parallel to all three axes and components of rotation about all three axes. Provision is made for the user who would rather use "pure" rigid body modes than the modes given by MSC/NASTRAN in the linear systems model. However, the rigid body modes calculated by MSC/NASTRAN must be used in the linear systems model if the IMAT physical data recovery procedure is to be used to obtain physical output.

The mode shapes have a very useful property, called orthogonality, that is expressed in the equations

SECTION 6

CONTROLS

Development of Dynamic Equations of Motion for
Flexible Structures with Active Controls - - - - - - - - - - 6- 1
Generation of Linear Systems Matrices - - - - - - - - - - 6- 7

Controls Processors

DEFINE CONTROL SYSTEM - - - - - - - - - - - - 6-14
GENERATE LINEAR SYSTEMS MATRICES - - - - 6-19
EXECUTE MATRIXX - - - - - - - - - - - - - - - - 6-21
RECORD CONTROL SIMULATION - - - - - - - - - 6-22
QUERY EIGENVECTOR FILE - - - - - - - - - - - - 6-29

The partial differential equations governing the small motion of a linear, elastic structure can generally be put in the form:

$$M \frac{\partial^2 w(s,t)}{\partial t^2} + D \frac{\partial w(s,t)}{\partial t} + L_s[w(s,t)] = F(s,t) \tag{1}$$

where the first two terms on the left side of the equation are the dynamic inertia and dissipative forces and the third term on the left is an elastic stiffness force described by a linear differential operator in spatial coordinates acting on the elastic deformation of the structure. These forces are balanced by the applied forces on the right side of the equation.

In practice, means other than the integration of the partial differential equations governing the motion of a continuous complex structure are used to determine the dynamic behavior of the structure since the linear differential operator and spatial boundary conditions become unwieldy. The most common approach to the solution of a complex structure is to represent the structure by an assemblage of smaller structures such as rods, beams, and rectangular and triangular plates with known or assumed solutions. This procedure of discretizing the structure into smaller, simpler elements with known solutions is referred to as the finite element method. The behavior of the structure is now represented by the behavior of the nodes connecting the elements describing the structure and the governing equations have been reduced to a set of ordinary differential equations with a finite number of degrees-of-freedom, r, where r is the minimum number of measurements required to completely describe the motion of the nodes.

The IMAT procedure for analyzing flexible space structures with active controls is based on the principle of modal superposition. The equations of motion for an r-degree-of-freedom structure are written as

$$[m]\ddot{p} + [c]\dot{p} + [k]p = \bar{f} \qquad (2)$$

where [m], [c] and [k] are the r x r mass, damping and stiffness matrices, respectively, and p is an r-element vector of nodal displacements. The vector $\bar{f}$ contains applied time-dependent forces that can be functions of p and/or $\dot{p}$ when control forces are employed.

In order to reduce the number of degrees of freedom in the problem and decouple the equations of motion, the undamped natural modes of the system are used to define a generalized coordinate system. The natural modes are calculated by ignoring [c] and $\bar{f}$ in equation (2) and solving the resulting homogeneous equations for the first s mode shapes, ϕ_i (r-element vectors), and frequencies, ω_i (i = 1,2, ... ,s).

$$[m]\ddot{p} + [k]p = 0 \qquad (3)$$

Mode i is called a rigid body mode if $\omega_i = 0.0$ and otherwise it is called a flexible mode. The computer codes which perform this solution (IMAT uses MSC/NASTRAN) may not return exact zeros for rigid body mode frequencies, but in practice these have been at least four orders of magnitude smaller than the smallest flexible mode frequency, so are easily identifiable. Usually there will be six rigid body modes although there can be fewer if rigid body degrees-of-freedom have been constrained during the solution process. Usually, each rigid body mode will incorporate components of translation parallel to all three axes and components of rotation about all three axes. Provision is made for the user who would rather use "pure" rigid body modes than the modes given by MSC/NASTRAN in the linear systems model. However, the rigid body modes calculated by MSC/NASTRAN must be used in the linear systems model if the IMAT physical data recovery procedure is to be used to obtain physical output.

The mode shapes have a very useful property, called orthogonality, that is expressed in the equations

$$\phi_i^T\,[a]\,\phi_j = 0, \quad i \neq j$$
$$\phi_i^T\,[a]\,\phi_j \neq 0, \quad i = j \tag{4}$$

where [a] represents the mass matrix, [m], or the stiffness matrix, [k]; and the superscript T represents the transpose of the vector. For the present discussion it is assumed that [a] can also represent the damping matrix, [c], although this is not in general true. Equations (4) are used to normalize the mode shapes, ϕ_i, before they are stored in an IMAT database. The mode shapes are normalized such that

$$\phi_i^T\,[m]\,\phi_i = 1 \tag{5}$$

for all modes that are calculated (this is referred to as "normalizing the eigenvectors to the generalized masses").

The equations of motion (2) may now be decoupled to yield a set of single-degree-of-freedom equations if the forces, $\bar{f}$, are not dependent on p and $\dot{p}$. The decoupling is accomplished by a transformation of coordinates. First the generalized (modal) coordinates, q, are defined by the equation

$$p = \Phi\, q \tag{6}$$

where Φ is an r x s matrix defined as

$$\Phi = [\,\phi_1\;\phi_2\;\ldots\;\phi_s\,].$$

Substituting equation (6) into equation (2) and premultiplying by the transpose of the jth mode shape, ϕ_j^T, the generalized-coordinate (modal) form of the equations of motion are obtained as

$$\phi_j^T [m] \Phi \ddot{q} + \phi_j^T [c] \Phi \dot{q} + \phi_j^T [k] \Phi q = \phi_j^T \bar{f} \quad (7)$$

for j=1,...,s.

If the terms on the left hand side of equation (7) are expanded, the orthogonality properties of the modes (equations (4)) will cause all of the terms to disappear except for the terms for mode j. Equation (7) then becomes a set of uncoupled equations:

$$\phi_j^T [m] \phi_j \ddot{q} + \phi_j^T [c] \phi_j \dot{q} + \phi_j^T [k] \phi_j q = \phi_j^T \bar{f} \quad (8)$$

for j=1,...,s.

Equation (8) is simplified by introducing the generalized coordinate (modal) mass, damping, stiffness and force as follows:

$$M_j = \phi_j^T [m] \phi_j = 1 \quad (9)$$

$$C_j = \phi_j^T [c] \phi_j \quad (10)$$

$$K_j = \phi_j^T [k] \phi_j \quad (11)$$

$$\bar{f}_j = \phi_j^T \bar{f} \quad (12)$$

It can be shown that the generalized stiffness and generalized mass are related by the equation

$$K_j = \omega_j^2 M_j = \omega_j^2 \quad (13)$$

Using equations (9) through (13), equation (8) can be rewritten in generalized coordinate (modal) form as

$$\ddot{q}_j + 2\zeta_j \omega_j \dot{q}_j + \omega_j^2 q_j = \bar{f}_j, \; j = 1, 2, \ldots, s \quad (14)$$

where ζ_j is the modal damping ratio defined as

$$\zeta_j = C_j/2\omega_j \tag{15}$$

In matrix notation, equation (14) can be written as

$$\ddot{q} + [\, 2\zeta\omega \,]\, \dot{q} + [\, \omega^2 \,]\, q = \Phi^T \bar{f} \tag{16}$$

where $[\, 2\zeta\omega \,]$ and $[\, \omega^2 \,]$ represent s x s diagonal matrices composed of elements $2\zeta_j\omega_j$ and ω_j^2, respectively. In IMAT, the first order state space form of equation (16) is obtained by defining the state space vector x as $x = [\, q^T, \dot{q}^T \,]^T$. Then the first order state space form can be written as

$$\dot{x} = \begin{bmatrix} 0 & I \\ -[\omega^2] & -[2\zeta\omega] \end{bmatrix} x + \begin{bmatrix} 0 \\ \Phi^T \end{bmatrix} \bar{f} \tag{17}$$

Usually, some of the forces $\bar{f}$ are control forces that are functions of p and/or $\dot{p}$ and are therefore functions of q and $\dot{q}$. The modal equations (14) are therefore coupled through the forcing function on the right hand side and cannot be solved independently. In the IMAT process the finite element code MSC/NASTRAN is used to calculate the mode shapes and frequencies. The state space form of the coupled equations of motion are solved by a controls code such as MATRIXx.

In the preceding discussion, it was assumed that equations (4), the orthogonality conditions, could be used to decouple the damping terms in the equations of motion. This decoupling can actually be achieved only if certain conditions are met. These special cases are listed below:

1. The damping matrix [c] is a linear combination of the mass and stiffness matrices (Rayleigh damping), and therefore

equation (15) can be used to calculate the damping coefficients, ζ.

2. The damping terms in [c] are small, and the non-zero off-diagonal terms in the matrix that results from the product $\phi_j^T [c] \Phi$ in equation (7) can be neglected. Equation (15) is then used to calculate the damping coefficients, ζ.

3. The structure is lightly damped, and the generalized (modal) damping coefficients, ζ, are known for each mode retained in the analysis. In this case the generalized (modal) damping matrix is obtained by direct substitution of known values of the modal damping coefficients. The product $\phi_j^T [c] \Phi$ in v equation (7) is ignored, and equation (15) is not used.

The IMAT process proceeds under the assumptions that damping coefficients are known for each mode used in the analysis, and that there are no significant cross-coupling terms in the modal damping matrix. In actual practice the values of the modal damping coefficients may not be known, and suitable values must be assumed. As unscientific as this may seem, the reader should be aware of the fact that off-diagonal terms of the original [c] matrix are generally not known and are very difficult, if not impossible, to obtain by test in the laboratory for complex structures.

Linear systems theory deals with systems which have the form

$$\dot{x} = \hat{A} x + \hat{B} u + \hat{E} f \tag{18}$$

$$y = \hat{C} x + D u \tag{19}$$

where $\hat{A}$, $\hat{B}$, $\hat{C}$, D, and $\hat{E}$ are constant matrices; and x, u, f, and y are vector functions of time. Elements of x are the states of the system. Elements of u and f are controls and forcing functions of the system, respectively. Elements of y are the outputs or sensor readings.

We deal here with linear systems which model the dynamics of a flexible structure. The states are chosen from the rigid body motions and modal amplitudes and their rates with some restrictions. A rigid body motion or modal amplitude appears if and only if its rate also appears. Rigid body motion can be incorporated in the model either by using some or all of the six rigid body degrees-of-freedom obtained from Newtonian dynamics or by using some or all of the modal amplitudes corresponding to the zero frequency modes obtained from the finite element analysis, but not a mixture of these. However, the rigid body modes calculated by MSC/NASTRAN must be used in the linear systems model if the IMAT physical data recovery procedure is to be used to obtain physical output. We are dealing with the result of finite element modeling, so the dynamics of the structure are represented as the motions of a finite collection of points called nodes and represented in a coordinate system fixed to the structure. Inputs are restricted to be forces and/or torques applied at nodes. Outputs are restricted to be translations parallel to an axis or rotations about an axis at the nodes or the time derivatives thereof. The underlying equations on which the linear systems equations are based are given in equations (20) through (24) for modal amplitude and Newtonian rigid body motions.

IMAT's GENERATE LINEAR SYSTEMS MATRICES (GLSM) processor assumes that rigid body and flexible motion of the structure are governed by the equations:

$$M \ddot{p}_r = \sum_n F_n \tag{20}$$

$$I \ddot{\theta}_r = \sum_n (T_n + L_n F_n) \tag{21}$$

$$\ddot{q}_m + 2\zeta_m \omega_m \dot{q}_m + \omega_m^2 q_m = \sum_{n,c} (\Phi_{mnc} F_{nc} + \Phi'_{mnc} T_{nc}) \tag{22}$$

$$p_{nc} = \sum_m \Phi_{mnc} q_m \tag{23}$$

$$\theta_{nc} = \sum_m \Phi'_{mnc} q_m \tag{24}$$

where

- c — a subscript which varies in the set {x, y, z} (so, e.g., p_c could be p_x , p_y , or p_z).
- F_n — the vector $(F_{nx}, F_{ny}, F_{nz})'$.
- F_{nc} — the force applied at node n in the direction parallel to the c-axis, c = x, y, or z.
- I — mass moment of inertia matrix.
- L_n — lever arm matrix of node n with respect to the center of mass.
- m — a subscript which indexes the structural modes.
- M — mass matrix.
- n — a subscript which indexes the nodes of the structure.
- p_r — the vector $(p_{rx}, p_{ry}, p_{rz})^T = (x_r, y_r, z_r)^T$, the rigid body position.
- p_n — the vector $(p_{nx}, p_{ny}, p_{nz})^T = (x_n, y_n, z_n)^T$, the position of node n.

q_m modal amplitude of mode m.

T_n the vector $(T_{nx}, T_{ny}, T_{nz})^T$.

T_{nc} the torque applied at node n around the c-axis, c = x, y, or z.

z_m damping ratio of mode m.

q_r the vector $(\theta_{rx}, \theta_{ry}, \theta_{rz})^T = (\phi_r, \alpha_r, \beta_r)^T$, the rigid body attitude.

q_n the vector $(\theta_{nx}, \theta_{ny}, \theta_{nz})^T = (\phi_n, \alpha_n, \beta_n)^T$, the attitude of node n.

$\sum_m, \sum_n, \sum_{n,c}$, etc.
summation over all modes, over all nodes, over all nodes and coordinates, etc.

Φ_{mnc} mode shape component of mode m at node n in coordinate c.

Φ'_{mnc} mode slope component of mode m at node n in coordinate c.

w_m frequency of mode m.

Equation (20) describes rigid body translation; (21), rigid body attitude; (22), modal amplitude; (23), nodal position; and (24), nodal attitude.

Of the terms in these equations, the values for M, I, L_n, w_m, Φ_{mnc}, and Φ'_{mnc}, and the ranges of the indices m and n are derived from information extracted from the IMAT formatted data base; z_m, F_n and T_n are supplied by the user; and p_r, q_r, q_m, p_{nc}, and q_{nc} are defined by equations (20)-(24) together with initial conditions on p_r, q_r, q_m and their rates.

As a result of these considerations, the linear system takes on a special form. We take advantage of this form to compress the linear systems matrices information. First, we may take $x = [\, q^T, \dot{q}^T \,]^T$ where q contains the user's selection of Newtonian rigid body modes and/or modal amplitudes and $\dot{q}$ contains the corresponding velocities. We observe that an output which is a translation or rotation is calculated from just the q part of the state by a calculation of the form $c \cdot q$ (vector dot product) and that the associated rate is $c \cdot \dot{q}$. Finally, for the convenience of the controls engineer, we divide the system inputs into two classes, one containing actuators which are the inputs over which the engineer has control and the other containing external forces and torques which are influences acting on the structure which are beyond the engineer's control. We denote the actuator inputs by u, the external force/torque inputs by f, and restrict y to denote position sensors so that rate and acceleration sensors become $\dot{y}$ and $\ddot{y}$. Then equation (17) can be written as

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -[\omega^2] & -[2\zeta\omega] \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ \Phi^T B_c \end{bmatrix} u + \begin{bmatrix} 0 \\ \Phi^T B_f \end{bmatrix} f \qquad (25)$$

where 0 is a null matrix and I is an identity matrix; B_c and B_f are a control influence matrix and a disturbance influence matrix, respectively. The linear system equations then take the form of equations (18) and (19):

$$\begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} 0 & I \\ \mathrm{diag}(A_{21}) & \mathrm{diag}(A_{22}) \end{bmatrix} \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + \begin{bmatrix} 0 \\ B \end{bmatrix} u + \begin{bmatrix} 0 \\ E \end{bmatrix} f \qquad (26)$$

$$y = [\, C \;\; 0 \,] \begin{bmatrix} q \\ \dot{q} \end{bmatrix} + D\,u \qquad (27)$$

where

$$A_{21} = [\, -\omega_1^2 \;\; -\omega_2^2 \;\; . \; . \; . \;\; -\omega_s^2 \,]^T \qquad (28)$$

$$A_{22} = [\ -2\zeta_1\omega_1 \ \ -2\zeta_2\omega_2 \ \ . \ . \ . \ \ -2\zeta_s\omega_s\ ]^T \tag{29}$$

$$B = \Phi^T B_c \tag{30}$$

$$E = \Phi^T B_f \tag{31}$$

Assuming D is a null matrix, velocity and acceleration sensor readings can be obtained by

$$\dot{y} = C\,\dot{q} \tag{32}$$

$$\ddot{y} = C\,\ddot{q} \tag{33}$$

By substituting the $\ddot{q}$ term from equation (26) into the equation (33), we obtain

$$\ddot{y} = C\,\mathrm{diag}(A_{21})\,q + C\,\mathrm{diag}(A_{22})\,\dot{q} + C\,B\,u + C\,E\,f \tag{34}$$

Equation (33) does not fit into the generic linear systems form since $\ddot{q}$ is not part of our system states; but in any application where equation (26) is being solved, $\ddot{q}$ may well be available and equation (34) would be the efficient way to compute the acceleration sensor readings. By comparing equations (18) and (19) with equations (26) and (27), full linear systems matrices can be defined as follows:

$$\hat{A} = \begin{bmatrix} 0 & I \\ \mathrm{diag}(A_{21}) & \mathrm{diag}(A_{22}) \end{bmatrix} \tag{35}$$

$$\hat{B} = \begin{bmatrix} 0 \\ B \end{bmatrix} \tag{36}$$

$$\hat{C} = [\,C \ \ 0\,] \tag{37}$$

$$\hat{E} = \begin{bmatrix} 0 \\ E \end{bmatrix} \tag{38}$$

In the above example matrix $\hat{C}$ was defined to get physical displacement output at the sensor locations. The user can also define the $\hat{C}$ matrix to get velocities and accelerations using equations (32) and (33). Modal displacements, velocities, and accelerations can also be obtained by changing the $\hat{C}$ matrix.

Instead of passing full linear systems matrices to the GLSM data file, the GLSM processor passes the matrix A which contains only the vectors A_{21} and A_{22} in its first two columns, i.e,

$$A = [\, A_{21} \; A_{22} \,] \tag{39}$$

and the matrices B, C, and E. The matrices A and B are generated for most feedback control problems, but the generation of matrices C and E depends on whether the system has sensors and external forces. A quantity of supplemental information describing the modes, actuators, sensors, etc., being modeled is also made available in the list file generated by GLSM.

The compressed linear systems matrices (A, B, C, and E) stored in the GLSM data file are formatted so that the file is readily readable by MATRIXx. At this point, the user can easily recover the full linear systems matrices required for MATRIXx simulation from the compressed GLSM matrices by employing the MATRIXx commands.

For instance, assume that the user generated a GLSM data file named GLSM.DAT which contains A, B, and C matrices. Then the following sequence of MATRIXx commands can be used to recover the full matrices ($\hat{A}$, $\hat{B}$, and $\hat{C}$) required by MATRIXx.

```
LOAD GLSM.DAT          [Read GLSM.DAT into MATRIXx]
[N,NB]=SIZE(B);        [Find the matrix size of B and C
[NC,N]=SIZE(C);         Matrices]
ZA=0*ONES(N);          [Define an N x N null matrix]
IA=EYE(N);             [Define an N x N identity matrix]
ZB=0*ONES(N,NB);       [Define an N x NB null matrix]
AA=[ZA,IA;DIAG(A(:,1)),DIAG(A(:,2))];
                       [Recover Â matrix]
BB=[ZB;B];             [Recover B̂ matrix]
CC=C*[IA,ZA];          [Recover Ĉ matrix]
DD=C*ZB;               [Recover D matrix]
S=[AA,BB;CC,DD];       [Define a MATRIXx system matrix]
NS=N*2;                [Define the size of state variables]
SAVE GLSM.SNS S NS     [Save the system matrix and its size in
                        a file named GLSM.SNS]
```

MATRIXx always requires a D matrix. The user should define a null matrix of appropriate size if the system does not have a D matrix.

PROCESSOR: DEFINE CONTROL SYSTEM

PURPOSE: The DEFINE CONTROL SYSTEM processor facilitates the definition and/or modification of a control system.

ENTERING THE NAME OF THE DATABASE:

You will be asked to enter the name of the database to be used for controls analysis and its corresponding password. The database **must** reside in your current directory. The processor makes no attempt to copy databases from other directories.

CHOOSING A RELATION:

The first menu displayed is the EDIT SELECTION MENU:

DEFINE CONTROL SYSTEM: EDIT SELECTION MENU

1. DESCRIPTION OF ACTUATOR
2. DESCRIPTION OF SENSOR
3. DESCRIPTION OF EXTERNAL FORCE/TORQUE
4. COMPLETE DESCRIPTION OF SYSTEM
5. MODAL FREQUENCIES AND DAMPING RATIOS
6. RETURN TO THE IMAT EXECUTIVE

Selections 1 through 5 correspond to the database relations ACTUATOR, SENSOR, EXTFTAP, SYSTEM, and EIGNVALS, respectively. The processor is designed to permit you to choose control system components, add and/or modify data using the subsequent menus, and return to this menu to select the next component. Option 6 causes you to exit the processor and return to the IMAT EXECUTIVE.

SELECTING THE APPROPRIATE OPERATIONS:

The OPTIONS MENU provides the operations necessary to manipulate the control system data.

DEFINE CONTROL SYSTEM: OPTIONS MENU
Current Component

1. ADD
2. DELETE
3. MODIFY
4. REVIEW DATA
5. LIST DATA DESCRIPTION
6. OUTPUT DATA TO A FILE
7. OUTPUT DATA DESCRIPTION TO A FILE
8. PREVIOUS MENU [DEFAULT]

This menu serves as a basis for the system definition procedure. The various options are discussed on the following pages. Current Component refers to the control system that was chosen from the EDIT SELECTION MENU.

ADDING DATA TO THE DATABASE:

After choosing the ADD option from the OPTIONS MENU, you will be prompted to enter rows of data, one at a time. The first value in each component row serves as a unique identifier for the information in that row. You have the choice of automatic row identification generation or entering identification numbers for each row manually. If you choose automatic generation, the processor will create identification numbers of consecutive integers, beginning with 1. Otherwise, you have the opportunity to enter row numbers beginning with any positive integer that you desire. Next you will be prompted to enter appropriate data for each subsequent component in the system. You will then be asked if it is all right to load the information into the database. If you have made an error or choose not to load the information, you may enter it again when you are asked to enter the next row of data. When you have finished entering information, the program will return to the OPTIONS MENU.

DELETING DATA FROM THE DATABASE:

The DELETE option exhibits the following menu:

DEFINE CONTROL SYSTEM: DELETE MENU
Current Component

1. DELETE BY component row ID NUMBER
2. DELETE BY RANGE OF component row ID NUMBERS
3. PREVIOUS MENU [DEFAULT]

Option 1 allows you to enter the identification number of one component row that you would like to delete. Option 2 asks you to enter a range of identification numbers corresponding to the components that you want to delete. You have the opportunity to review the data in each of the selected rows before they are deleted. You are asked to confirm the deletion of each row from the database. After completing either selection, you are returned to the DELETE MENU. Entering option 3 will cause the program to return to the OPTIONS MENU.

MODIFYING DATA IN THE DATABASE

The first menu shown when you choose the MODIFY option is:

DEFINE CONTROL SYSTEM: MODIFY MENU
Current Component

1. DUPLICATE AN EXISTING component
2. MODIFY BY component ID NUMBER
3. PREVIOUS MENU [DEFAULT]

If you choose to duplicate an existing component row, (option 1) all of the data from the original row is copied into the duplicated row except for the row identification number. You are asked to choose the new identifier number with the default value being the next consecutive integer higher than the last component row identification number.

Option 2 provides the menu:

DEFINE CONTROL SYSTEM: MODIFY BY component ID NUMBER
Current Component

1. MODIFY EACH ITEM IN AN EXISTING component
2. MODIFY ONE ITEM IN AN EXISTING component
3. PREVIOUS MENU [DEFAULT]

Modifying each item in a component row involves re-entering each item of data in the row. Option 2 allows you to modify only one item in a data row. You are asked to enter the name of the value that you want to modify. If you do not know the names of the values, enter LIST at the prompt and a description of the component row will be provided. Option 3 returns to the MODIFY MENU. At this point you may enter a carriage return to view the OPTIONS MENU again.

REVIEWING DATA:

To review data of the current component, select options from the menu:

DEFINE CONTROL SYSTEM: REVIEW MENU
Current Component

1. BRIEF REVIEW
2. REVIEW ONE component row
3. REVIEW BY RANGE OF component row ID NUMBERS
4. REVIEW ALL DATA
5. PREVIOUS MENU [DEFAULT]

The BRIEF REVIEW option displays each component row number and its corresponding row description. To review one component row, enter the appropriate identification number and the data in that row will be displayed. Enter the corresponding row identification numbers to review a range of data rows. Option 4 displays all of the current component data. Option 5, returns to the OPTIONS MENU.

LISTING DATA DESCRIPTIONS:

The LIST DATA DESCRIPTIONS, option (5) of the OPTIONS MENU, lists the descriptions of the current component.

CREATING A PRINT FILE:

Options 6 and 7 of the OPTIONS MENU output the data of the current component and the component description respectively. You may choose the type of data to be written to the file from the following menu:

DEFINE CONTROL SYSTEM: PRINT DATA MENU
Current Component

1. PRINT DATA SUMMARY
2. PRINT DATA
3. PREVIOUS MENU {DEFAULT]

Option 1 prints each row of data and its corresponding description to the print file. Option 2 writes all of the current component data to the file. Option 3 returns to the previous menu.

PROCESSOR: GENERATE LINEAR SYSTEMS MATRICES

PURPOSE: The GENERATE LINEAR SYSTEMS MATRICES processor creates a data file of linear systems matrices for input into the EXECUTE MATRIXX processor.

ENTERING THE NAME OF THE DATABASE AND THE NAME OF THE EIGENVECTOR FILE:

You are asked to enter the name of the database which contains your linear systems information, and the corresponding password for this database. The processor will also ask you to enter the name of the eigenvector file that is associated with the database (see the finite element processor LOAD ANALYSIS RESULTS). If the database and/or the eigenvector file are not in your current directory, you are asked to enter the directory where they reside. The processor will make copies of these files in your current working directory, although they will be deleted upon completion of the processor.

ENTERING A NAME FOR THE MATRIXx DATA FILE:

You are asked to enter a name for the MATRIXx data file that will be created. When prompted to enter a file name, you may choose the default name, GLSMMAX.DAT, by entering a carriage return.

ENTERING A NAME FOR THE LIST FILE

You are asked to enter a name for the list file which will contain the linear matrices data in a specified format. When prompted to enter a file name, you may choose the default name, GLSM.PRN, by entering a carriage return.

GENERATING THE MATRIXx DATA FILE:

This processor creates linear systems matrices according to the specifications described in the database. You are asked to enter the system identification number that corresponds to the entry in relation SYSTEM that will be used to define these matrices. If the data required for each set exists, the processor will generate matrices

A, B, C, and E respectively. Otherwise error messages will be displayed to indicate inconsistencies in your system definition. Although matrices C and E are not required for a valid linear system, there must be sufficient data in the database to generate the A and B matrices. Upon completion of each matrix the following messages will appear:

GENERATING LINEAR SYSTEMS MATRICES

A MATRIX...
B MATRIX...
C MATRIX... { data permitting }
E MATRIX... { data permitting }

EXITING GENERATE LINEAR SYSTEMS MATRICES:

After the data file is generated, the processor will return to the IMAT EXECUTIVE.

EXECUTE MATRIXX 1

PROCESSOR: EXECUTE MATRIXX

PURPOSE: The EXECUTE MATRIXX processor executes Integrated Systems' MATRIXx/SYSTEM_BUILD giving you a choice of memory size. You should be familiar with MATRIXx commands in order to execute this processor.

MAIN MENU OPTIONS:

MATRIXx uses a stack to store variables and temporary results in calculations. The different stack sizes available for MATRIXx at your installation reflect the central memory that will be available for your execution. (The following menu may be different at your installation):

EXECUTE MATRIXX: STACK SIZE MENU

ENTER THE STACK SIZE DESIRED FOR THIS EXECUTION OF MATRIXX OR "Q" TO QUIT:

1. 100,000 ELEMENT STACK [DEFAULT]
2. 250,000 ELEMENT STACK
3. 350,000 ELEMENT STACK
4. 450,000 ELEMENT STACK
5. 1,000,000 ELEMENT STACK

Upon completion of MATRIXx, you will be returned to this menu. Enter a Q to return to the IMAT EXECUTIVE. If MATRIXx issues an error indicating that you need more memory for a particular problem, choose a larger stack size for your execution.

PROCESSOR: RECORD CONTROL SIMULATION

PURPOSE: The RECORD CONTROL SIMULATION processor is designed to enter into the IMAT database all of the information required by the RECOVER PHYSICAL OUTPUT processor when generating an MSC/NASTRAN modified Solution 31 bulk data deck to obtain physical output such as displacements or stresses from a state space simulation.

SELECTING A DATABASE:

You will be asked to enter enter the name of the database to be used with this processor. Because this program writes to the database, your database files **must** reside in your current default directory. If they do not, you will need to exit IMAT and reset your default directory.

MAIN MENU OPTIONS:

After the program has successfully opened your database, you will see the MAIN MENU which offers options to let you enter or change a record of a controls simulation. To exit the program at any time, enter Q at this or any menu:

RECORD CONTROL SIMULATION: MAIN MENU

CHOOSE THE FUNCTION DESIRED OR Q TO QUIT:

1. ENTER RECORD OF NEW SIMULATION INTO DATABASE [DEFAULT]
2. CHANGE/REVIEW EXISTING SIMULATION RECORD IN DATABASE
3. DELETE SIMULATION RECORD

MAIN MENU OPTION 1: ENTER RECORD OF NEW SIMULATION INTO DATABASE

Option 1 will prompt you for all of the information required to enter a full record of a controls analysis simulation into your database. The program automatically generates a new simulation number and then prompts you for:

1. Analyst's name: Enter the name of the controls analyst (40 char. max)

2. Analysis date: Enter the date of the analysis in the format: 01-FEB-1988 [default = today]

3. ID of the control system used in the simulation:

   You will select this from the SYSTEM MENU which presents all of the systems documented in your database:

RECORD CONTROL SIMULATION: SYSTEM MENU

SELECT THE SYSTEM USED IN THIS SIMULATION:

	SYSTEM ID	DESCRIPTION
1.	1	This is your description of system # 1.
2.	2	This is your description of system # 2.
3.	...	...

Selection of a system will determine the structure of the information you enter about your simulation. You will be prompted for time history files to match the system you select for this simulation.

4. Simulation description: An optional description of your simulation (80 char. max)

5. Simulation time in seconds:

Initial time:	A real value [default=0.0]
Final time:	A real value
Time step interval:	A real value

6. The names of the 3 modal output files:

Modal displacement file:	(max 40 char.)
Modal velocity file:	(max 40 char.)
Modal acceleration file:	(max 40 char.)

Enter the file names only. Do not include any directory specification: EX: MYFILE.DAT

7. Information concerning the time history file for every control force/torque and applied load in your control system:

Time history file name:	Give the file name only - include no directory specification (40 char max).
Start time:	A real value [default = 0.0]
End time:	A real value
Uniform time step:	A real value

At this point, the record of your simulation is complete, and the RECORD CONTROL SIMULATION processor will return you to the MAIN MENU.

MAIN MENU OPTION 2: CHANGE/REVIEW EXISTING SIMULATION RECORD IN DATABASE

Option 2 will allow you to inspect the record of a controls simulation and make any corrections required. Use this option only for corrections or inspection, not to enter a record of a new simulation (See MAIN MENU OPTION 1). Under this option you will first choose a simulation to change from the SIMULATIONS MENU:

RECORD CONTROL SIMULATION: SIMULATIONS MENU

SELECT THE SIMULATION RECORD YOU WANT TO CHANGE:

	SIMUL. ID	SYSTEM ID	DESCRIPTION
1.	1	1	Your description of simulation #1
2.	2	1	Your description of simulation #2
3.	...	...	

The RECORD CONTROL SIMULATION processor will now display the CHANGE MENU so that you can quickly access the part of the record you want to change:

RECORD CONTROL SIMULATION: CHANGE MENU

WHAT PART OF THIS SIMULATION RECORD DO YOU WANT TO CHANGE:

1. SIMULATION DESCRIPTION INFORMATION
2. CONTROL FORCE/TORQUE TIME HISTORY FILES
3. APPLIED LOAD TIME HISTORY FILES
4. RETURN TO "SIMULATIONS MENU" [DEFAULT]

CHANGE MENU OPTION 1: SIMULATION DESCRIPTION INFORMATION

This option will allow you to inspect or change the general description of your simulation (i.e., analyst, date, description, simulation times, and modal output file names). You will first see the SIMULATION DESCRIPTION MENU:

RECORD CONTROL SIMULATION: SIMULATION DESCRIPTION MENU

CHOOSE THE DATA ITEM YOU WANT TO CHANGE:

	DATA ITEM	CURRENT VALUE
1.	DESCRIPTION:	Your description of this simulation
2.	SIMULATION DATE:	01-FEB-1988
3.	ANALYST:	JOHN DOE
4.	INITIAL TIME (SECONDS):	0.0
5.	FINAL TIME ("):	200.0
6.	TIME STEP ("):	.5
7.	MODAL DISPLACEMENT FILE:	RQUE.DAT
8.	MODAL VELOCITY FILE:	RVEL.DAT
9.	MODAL ACCELERATION FILE:	RACC.DAT
10.	NO CHANGES - RETURN TO "CHANGE MENU" [DEFAULT]	

Choose options 1-9 to make changes to the general description information you have previously entered about this simulation. You will then be prompted for a new value for the item. You may not change either the simulation ID, which is generated automatically for you, or the control system ID. The time history files entered for this simulation must match the actuators and external forces defined for a system. Consequently, changing the system ID might invalidate this record. If the system ID is incorrect, first delete this record (OPTION 3 from the MAIN MENU) and enter a new record (OPTION 1 from the MAIN MENU).

Select menu item 10 (NO CHANGES) or menu item 11 (SAVE CHANGES TO THE DATABASE - This appears after you have made changes) to return to the CHANGE MENU after you have made all changes desired.

CHANGE MENU OPTION 2: CONTROL FORCE/TORQUE TIME HISTORY FILES

This option will allow you to inspect or change information about the control force/torque time history files including the file name, start and end times, and the time step interval. You will first see the CONTROL FORCE/TORQUE MENU:

RECORD CONTROL SIMULATION: CONTROL FORCE/TORQUE MENU

SELECT THE RECORD YOU WANT TO REVIEW/CHANGE:

	ACTUATOR ID	TIME HISTORY FILE
1.	1	ACC1.DAT
2.	2	ACC2.DAT
3.	RETURN TO "CHANGE MENU" [DEFAULT]	

Make one selection at a time from this list of all the control force/torque time history files recorded for this simulation. You will then be allowed to inspect and/or change the data regarding this file:

RECORD CONTROL SIMULATION: TIME HISTORY CHANGE/REVIEW MENU

CHOOSE THE DATA ITEM YOU WANT TO CHANGE:

	DATA ITEM	CURRENT VALUE
1.	TIME HISTORY FILE:	ACC1.DAT
2.	START TIME (SECONDS):	0.000
3.	END TIME ("):	10.000
4.	TIME STEP ("):	0.100
5.	NO CHANGES [DEFAULT]	

Choose options 1-4 to make changes to the information about the control force/torque time history files that you previously entered. You will then be prompted to supply a new value for the item. You may not change the actuator ID which was previously defined for this control system. Select option 5 (NO CHANGES) or option 6 (SAVE CHANGES TO DATABASE - this appears once you have made changes) when you have completed all changes concerning this time history file.

CHANGE MENU OPTION 3: APPLIED LOAD TIME HISTORY FILES

This option will allow you to change information about your applied load time history files. See CHANGE MENU OPTION 2 for a detailed description of the menus involved.

CHANGE MENU OPTION 4: RETURN TO "SIMULATIONS MENU"

Select this option when you have completed all changes to this simulation. You may return to the MAIN MENU by selecting the last option on the SIMULATIONS MENU.

MAIN MENU OPTION 3: DELETE SIMULATION RECORD

This option will delete the record of a controls simulation from your database. It will **not** delete any of your system, actuator, or loads information or any structural model information. Use this option if you have entered a simulation record in error and want to replace it with a new record.

QUERY EIGENVECTOR FILE 1

PROCESSOR: QUERY EIGENVECTOR FILE

PURPOSE: To display eigenvector data for a specific node number and mode number. This data is stored in the eigenvector file created by the LOAD ANALYSIS RESULTS processor (MSC/NASTRAN menu, option 4).

ENTERING THE NAME OF THE EIGENVECTOR FILE:

You will be asked to enter the name of the eigenvector file that you want to interrogate. If this file is not in your current directory the processor will ask you to enter the full pathname to the directory where the file is located. The processor will copy the file into your current directory; however it will be deleted upon completion.

DISPLAYING EIGENVECTOR DATA:

This processor displays information based on node and mode numbers for a given eigenvector. You are asked to enter the node number and the corresponding mode number (attribute CONDNUM) of the data you want to examine. If data exists in the file coinciding with the values that you entered, the data will be exhibited in the following format:

NODENUM	X-EIGEN	Y-EIGEN	Z-EIGEN
xxxxxxx	xxxxxxx	xxxxxxx	xxxxxxx

X-ROTN	Y-ROTN	Z-ROTN	CONDNUM
xxxxxx	xxxxxx	xxxxxx	xxxxxx

If an entry can not be found matching the given node and mode numbers, the processor will display:

!!! EIGENVECTOR DATA NOT FOUND WITH

NODENUM: xxxxxx
CONDNUM: xxxxxx

QUERY EIGENVECTOR FILE 2

EXITING THE PROCESSOR:

You will be asked if you would like to locate more eigenvector data. If not, the processor will return to the IMAT EXECUTIVE.

APPENDIX A

IMAT DATABASE DEFINITION

Relation Overview - A- 1
Attribute Overview - A- 4
Schema - A-25

Relation Overview 1

ACTUATOR
ACTUATOR DESCRIPTION
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

BEAMPROP
GENERAL BEAM PROPERTY
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

BEAMREF
BEAM ORIENTATION
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

BEAMS
TWO NODE ELEMENT
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

CONMASS
CONCENTRATED MASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

CONSTRN
NODAL CONSTRAINTS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

DBDESC
DATABASE DESCRIPTION
ORIGIN: CREATE DATABASE, DEFINE CONTROL SYSTEM
DESTINATION: USER'S INFORMATION

DYNFORCE
DYNAMIC FORCE DESCRIPTION
ORIGIN: RECORD CONTROL SIMULATION
DESTINATION: RECOVER PHYSICAL OUTPUT

EIGNVALS
NATURAL FREQUENCIES
ORIGIN: MSC/NASTRAN, DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES, SUPERTAB

EIGNVECT
EIGENVECTOR DATA
ORIGIN: MSC/NASTRAN
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES, SUPERTAB

ELEMLOAD
ELEMENT PRESSURE LOADS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

ELEM-MAP
A SCRATCH RELATION
ORIGIN: CREATE DATABASE
DESTINATION: FORMAT BULK DATA, LOAD BULK DATA

EVECINFO
INFORMATION ABOUT EIGENVECTOR FILE
ORIGIN: DEFINE CONTROL SYSTEM, LOAD ANALYSIS RESULTS
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

EXTFTAP
EXTERNAL FORCE/TORQUE APPLICATION POINT DEFINITION
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

MAT-PROP
MATERIAL PROPERTIES
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

MODELDEF
DIRECTORY OF GROUPS
ORIGIN: SUPERTAB
DESTINATION: USER'S INFORMATION, SUPERTAB

NODELOAD
NODAL TEMPERATURES AND LOADS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NODES
NODE NUMBERS AND COORDINATES
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

QUADS
FOUR-NODE ELEMENT
ORIGIN: SUPERTAB, MSC/NASTRAN,
DESTINATION: MSC/NASTRAN, SUPERTAB

RIGIDBAR
MSC/NASTRAN RIGID ELEMENTS (RBARS)
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

RIGPROP
RIGID-BODY MASS PROPERTIES
ORIGIN: MSC/NASTRAN
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

SCRREL
A SCRATCH RELATION
ORIGIN: FORMAT BULK DATA, LOAD BULK DATA
DESTINATION: MSC/NASTRAN

SENSOR
SENSOR DEFINITION
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

SIMULATE
SIMULATION INFORMATION
ORIGIN: RECORD CONTROL SIMULATION
DESTINATION: RECOVER PHYSICAL OUTPUT

STRNENG
STRAIN ENERGY AND STRAIN ENERGY DENSITY FOR EACH ELEMENT
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

SYSTEM
LINEAR SYSTEM DEFINITION
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

TRIANGLS
THREE-NODE ELEMENT
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

ABSORP TYPE = REAL UNITS = DIMENSIONLESS
ABSORPTIVITY
RELATIONS USED IN: MAT-PROP
ORIGIN SUPERTAB, MSC/NASTRAN
DESTINATION CURRENTLY UNUSED

ACCELFILE TYPE = TEXT 40 UNITS = DIMENSIONLESS
NAME OF FILE CONTAINING MODAL ACCELERATIONS
RELATIONS USED IN: SIMULATE
ORIGIN: RECORD CONTROL SIMULATION
DESTINATION: RECOVER PHYSICAL OUTPUT

ACDOC TYPE = TEXT 80 UNITS = DIMENSIONLESS
DESCRIPTION OF ACTUATOR
RELATIONS USED IN: ACTUATOR
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: USER'S INFORMATION

ACID TYPE = INTEGER UNITS = DIMENSIONLESS
ACTUATOR ID NUMBER
RELATIONS USED IN: ACTUATOR
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

ALPHA1 TYPE = REAL UNITS = DIMENSIONLESS
TRANSVERSE SHEAR DEFLECTION CONSTANT, PLANE 1
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

ALPHA2 TYPE = REAL UNITS = DIMENSIONLESS
TRANSVERSE SHEAR DEFLECTION CONSTANT, PLANE 2
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

ANALYST TYPE = REAL UNITS = DIMENSIONLESS
NAME OF CONTROLS ANALYST
RELATIONS USED IN: SIMULATE
ORIGIN: RECORD CONTROL SIMULATION
DESTINATION: USER'S INFORMATION

AREA TYPE = REAL UNITS = L**2
CROSS-SECTIONAL AREA
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

APPLFOR TYPE = REAL UNITS = ML**2/T**2 or ML/T**2
APPLIED FORCE
RELATIONS USED IN: NODELOAD
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

AXIS TYPE = INTEGER UNITS = DIMENSIONLESS
1,2,3 FOR X, Y, Z TRANSLATION 4,5,6 FOR X, Y, Z ROTATION
RELATIONS USED IN: SENSOR
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

BMREFER TYPE = INTEGER UNITS = DIMENSIONLESS
BEAM-ORIENTATION REFERENCE NUMBER
RELATIONS USED IN: BEAMREF, BEAMS, SCRREL
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

CG TYPE = RVEC 3 UNITS = L
CENTER OF MASS (X, Y, Z)
RELATIONS USED IN: RIGPROP
ORIGIN: MSC/NASTRAN
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

CID TYPE = INTEGER UNITS = DIMENSIONLESS
COORDINATE FRAME ID
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

CMPONENT TYPE = INTEGER UNITS = DIMENSIONLESS
GROUP NUMBER
RELATIONS USED IN: BEAMS, BEAMSTRS, ELEMLOAD, MODELDEF, TRIANGLS, QUADS, RIGIDBAR, STRNENG, SCRREL
ORIGIN: SUPERTAB
DESTINATION: MSC/NASTRAN, SUPERTAB

CONDNUM TYPE = INTEGER UNITS = DIMENSIONLESS
LOAD-CASE ID, OR MODE NUMBER
RELATIONS USED IN: EIGNVECT, EIGNVALS, ELEMLOAD, NODELOAD, STRNENG
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

CONDUCT TYPE = REAL UNITS = BTU(IN-SEC-DEG F)
CONDUCTIVITY
RELATIONS USED IN: MAT-PROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: CURRENTLY UNUSED

COS-NAST TYPE = TEXT 8 UNITS = DIMENSIONLESS
COSMIC NASTRAN ELEMENT TYPE
RELATIONS USED IN: ELEM-MAP
ORIGIN: CREATE DATABASE
DESTINATION: CURRENTLY UNUSED

COS-SECT TYPE = TEXT 8 UNITS = DIMENSIONLESS
COSMIC NASTRAN SECTION PROPERTY
RELATIONS USED IN: ELEM-MAP
ORIGIN: CREATE DATABASE
DESTINATION: CURRENTLY UNUSED

CONTROLS TYPE = IVEC IVAR UNITS = DIMENSIONLESS
ID NUMBERS (ACID'S) OF ACTUATORS
RELATIONS USED IN: SYSTEM
ORIGIN DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

CSYSTYPE TYPE = TEXT 80 UNITS = DIMENSIONLESS
TYPE OF CONTROL SYSTEM
RELATIONS USED IN DBDESC
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: USER'S INFORMATION

DAMPRAT TYPE = REAL UNITS = DIMENSIONLESS
MODAL DAMPING RATIO
RELATIONS USED IN: EIGNVALS
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

DATECRE TYPE = DATE UNITS = DIMENSIONLESS
DATE DATABASE WAS CREATED
RELATIONS USED IN: DBDESC
ORIGIN: CREATE DATABASE
DESTINATION: USER'S INFORMATION

DBDESC TYPE = TEXT 80 UNITS = DIMENSIONLESS
DATABASE DESCRIPTION
RELATIONS USED IN: DBDESC
ORIGIN: CREATE DATABASE
DESTINATION: USER'S INFORMATION

DELTAT TYPE = REAL UNITS = T
UNIFORM TIME STEP
RELATIONS USED IN: DYNFORCE, SIMULATE
ORIGIN: RECORD CONTROL SIMULATION
DESTINATION: RECOVER PHYSICAL OUTPUT

DEPDOFA	TYPE = INTEGER	UNITS = DIMENSIONLESS
DEPENDENT DOF AT END A		
RELATIONS USED IN:	RIGIDBAR	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	
DEPDOFB	TYPE = INTEGER	UNITS = DIMENSIONLESS
DEPENDENT DOF AT END B		
RELATIONS USED IN:	RIGIDBAR	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	
DESCRIPT	TYPE = TEXT 50	UNITS = DIMENSIONLESS
TEXT DESCRIPTION FOR THIS ELEMENT TYPE AND GROUP NUMBER		
RELATIONS USED IN:	MODELDEF	
ORIGIN:	SUPERTAB	
DESTINATION:	USER'S INFORMATION	
DIFFCOMP	TYPE = REAL	UNITS = DIMENSIONLESS
DIFFUSE COMPONENT		
RELATIONS USED IN:	MAT-PROP	
ORIGIN:	CURRENTLY UNUSED	
DESTINATION:	CURRENTLY UNUSED	
DISPFILE	TYPE = TEXT 40	UNITS = DIMENSIONLESS
NAME OF FILE CONTAINING MODAL DISPLACEMENTS		
RELATIONS USED IN:	SIMULATE	
ORIGIN:	RECORD CONTROL SIMULATION	
DESTINATION:	RECOVER PHYSICAL OUTPUT	
DLASTMOD	TYPE = DATE	UNITS = DIMENSIONLESS
DATE OF LAST MODIFICATION		
RELATIONS USED IN:	DBDESC	
ORIGIN:	DATABASE OWNER	
DESTINATION:	USER'S INFORMATION	
DLOADFIL	TYPE = TEXT 40	UNITS = DIMESNIONLESS
LOAD/FORCE TIME HISTORY FILE NAME		
RELATIONS USED IN:	DYNFORCE	
ORIGIN:	RECORD CONTROL SIMULATION	
DESTINATION:	RECOVER PHYSICAL OUTPUT	
EAL-ELT	TYPE = TEXT 8	UNITS = DIMENSIONLESS
EAL ELEMENT TYPE		
RELATIONS USED IN:	SCRREL	
ORIGIN:	CREATE DATABASE	
DESTINATION:	CURRENTLY UNUSED	

EAL-SEC	TYPE = TEXT 8	UNITS = DIMENSIONLESS
EAL SECTION TYPE		
RELATIONS USED IN:	SCRREL	
ORIGIN:	CREATE DATABASE	
DESTINATION:	CURRENTLY UNUSED	

EIGENDOC	TYPE = TEXT VAR	UNITS = DIMENSIONLESS
DESCRIPTION OF MODE OR FREQUENCY		
RELATIONS USED IN:	EIGENVALS	
ORIGIN:	DEFINE CONTROL SYSTEM	
DESTINATION:	USER'S INFORMATION	

EL-TYPE	TYPE = TEXT 8	UNITS = DIMENSIONLESS
IMAT ELEMENT TYPE		
RELATIONS USED IN:	BEAMS, ELEM-MAP, MODELDEF, TRIANGLS, SCRREL, STRNENG, QUADS, RIGIDBAR	
ORIGIN:	LOAD UNIVERSAL FILE, LOAD BULK DATA	
DESTINATION:	FORMAT BULK DATA, FORMAT UNIVERSAL FILE	

ELD	TYPE = TEXT 8	UNITS = DIMENSIONLESS
ELEMENT TYPE		
RELATIONS USED IN:	SCRREL	
ORIGIN:	FORMAT BULK DATA, LOAD BULK DATA	
DESTINATION:	INTERNAL USE ONLY	

ELEMENT	TYPE = INTEGER	UNITS = DIMENSIONLESS
ELEMENT NUMBER		
RELATIONS USED IN:	BEAMS, BEAMSTRS, CONMASS, ELEMLOAD, QUADS, STRNENG, TRIANGLS, RIGIDBAR	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	

EMISSIV	TYPE = REAL	UNITS = DIMENSIONLESS
EMISSIVITY		
RELATIONS USED IN:	MAT-PROP	
ORIGIN:	SUPERTAB	
DESTINATION:	CURRENTLY UNUSED	

ENDU-LIM	TYPE = REAL	UNITS = M/(LT**2)
ENDURANCE LIMIT		
RELATIONS USED IN	MAT-PROP	
ORIGIN:	CURRENTLY UNUSED	
DESTINATION:	CURRENTLY UNUSED	

EVECFNAM TYPE = TEXT 40 UNITS = DIMENSIONLESS
NAME OF RAPID-ACCESS EIGENVECTOR FILE
RELATIONS USED IN: EVECINFO
ORIGIN: LOAD ANALYSIS RESULTS
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES, QUERY EIGENVECTOR FILE

EXAPDOC TYPE = TEXT VAR UNITS = DIMENSIONLESS
DESCRIPTION OF EXTERNAL FORCE/TORQUE
RELATIONS USED IN: EXTFTAP
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: USER'S INFORMATION

EXAPID TYPE = INTEGER UNITS = DIMENSIONLESS
EXTERNAL FORCE/TORQUE ID NUMBER
RELATIONS USED IN: EXTFTAP
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

EXFORTOR TYPE = IVEC VAR UNITS = DIMENSIONLESS
ID NUMBERS (EXAPID'S) OF EXTERNAL FORCE/TORQUE
RELATIONS USED IN: SYSTEM
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

F1 TYPE = REAL UNITS = DIMENSIONLESS
NON-UNIFORM TORSION CONSTANT
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: CURRENTLY UNUSED

FF TYPE = REAL UNITS = DIMENSIONLESS
UNIFORM TORSION CONSTANT
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

FMDIRECT TYPE = INTEGER UNITS = DIMENSIONLESS
FORCE OR MOMENT DIRECTION (1-6)
RELATIONS USED IN: NODELOAD
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

FORCID TYPE = INTEGER UNITS = DIMENSIONLESS
ID NUMBER OF CONTROL FORCE (ACID) OR OF APPLIED LOAD (EXAPID)
RELATIONS USED IN: DYNFORCE
ORIGIN: RECORD CONTROL SIMULATION
DESTNATION: RECOVER PHYSICAL OUTPUT

FORCTORQ TYPE = RVEC 6 UNITS = DIMENSIONLESS
X, Y, Z FORCE AND X, Y, Z TORQUE RESULTING FROM A UNIT INPUT
RELATIONS USED IN: ACTUATOR, EXTFTAP
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

FREQHZ TYPE = REAL UNITS = 1/T
CYCLIC FREQUENCY (HERTZ)
RELATIONS USED IN: EIGNVALS
ORIGIN: MSC/NASTRAN, DEFINE CONTROL SYSTEM
DESTINATION: SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES

FREQRPS TYPE = REAL UNITS = RAD/T
CIRCULAR FREQUENCY
RELATIONS USED IN: EIGNVALS
ORIGIN: MSC/NASTRAN, DEFINE CONTROL SYSTEM
DESTINATION: SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES

I1 TYPE = REAL UNITS = L**4
PRINCIPAL MOMENT OF INERTIA, PLANE 1 (IZZ)
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

I11 TYPE = REAL UNITS = ML**2
MASS MOMENT OF INERTIA
RELATIONS USED IN: CONMASS
ORIGIN SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN,SUPERTAB

I12A TYPE = REAL UNITS = L**4
AREA PRODUCT OF INERTIA
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

I2 TYPE = REAL UNITS = L**4
PRINCIPAL MOMENT OF INERTIA, PLANE 2 (IYY)
RELATIONS USED IN: BEAMPROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

I21 TYPE = REAL UNITS = ML**2
MASS MOMENT OF INERTIA
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

I22 TYPE = REAL UNITS = ML**2
MASS MOMENT OF INERTIA
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

I31 TYPE = REAL UNITS = ML**2
MASS MOMENT OF INERTIA
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

I32 TYPE = REAL UNITS = ML**2
MASS MOMENT OF INERTIA
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

I33 TYPE = REAL UNITS = ML**2
MASS MOMENT OF INERTIA
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN,SUPERTAB

INDDOFA TYPE = INTEGER UNITS = DIMENSIONLESS
INDEPENDENT DOF AT END A
RELATIONS USED IN: RIGIDBAR
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

INDDOFB TYPE = INTEGER UNITS = DIMENSIONLESS
INDEPENDENT DOF AT END B
RELATIONS USED IN: RIGIDBAR
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

MASMOMIN TYPE = RMAT 3,3 UNITS = ML**2
RIGID-BODY MASS MOMENT OF INERTIA MATRIX
RELATIONS USED IN: RIGPROP
ORIGIN: MSC/NASTRAN
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

MASS TYPE = REAL UNITS =M
MASS
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

MASSMAT TYPE = RMAT 3,3 UNITS = ML**2
RIGID-BODY MASS MATRIX
RELATIONS USED IN: RIGPROP
ORIGIN: MSC/NASTRAN
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

MATERIAL TYPE = TEXT 8 UNITS = DIMENSIONLESS
MATERIAL NAME OR ID NUMBER
RELATIONS USED IN: BEAMS, MAT-PROP, TRIANGLS, QUADS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

MATNUM TYPE = INTEGER UNITS = DIMENSIONLESS
MATERIAL NUMBER
RELATIONS USED IN: SCRREL
ORIGIN: FORMAT BULK DATA, LOAD BULK DATA
DESTINATION: MSC/NASTRAN

MOD-ELAS TYPE = REAL UNITS = M/(LT**2)
MODULUS OF ELASTICITY
RELATIONS USED IN: MAT-PROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

MOD-RIGI TYPE = REAL UNITS = M/(LT**2)
MODULUS OF RIGIDITY
RELATIONS USED IN: MAT-PROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

MODES TYPE = IVEC VAR UNITS = DIMENSIONLESS
VIBRATION (>0) OR NEWTONIAN RIGID BODY(-1 TO -6) MODES
RELATIONS USED IN: SYSTEM
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

MSC-NAST TYPE = TEXT 8 UNITS = DIMENSIONLESS
MSC/NASTRAN ELEMENT TYPE
RELATIONS USED IN: ELEM-MAP
ORIGIN: CREATE DATABASE
DESTINATION: FORMAT BULK DATA, LOAD BULK DATA

MSC-SEC TYPE = TEXT 8 UNITS = DIMENSIONLESS
MSC/NASTRAN SECTION PROPERTY
RELATIONS USED IN: ELEM-MAP
ORIGIN: CREATE DATABASE
DESTINATION: FORMAT BULK DATA, LOAD BULK DATA

NODE1 TYPE = INTEGER UNITS = DIMENSIONLESS
FIRST NODE IN ELEMENT CONNECTIVITY
RELATIONS USED IN: BEAMS, TRIANGLS, QUADS, RIGIDBAR, SCRREL
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NODE2 TYPE = INTEGER UNITS = DIMENSIONLESS
SECOND NODE IN ELEMENT CONNECTIVITY
RELATIONS USED IN: BEAMS, TRIANGLS, QUADS, RIGIDBAR, SCRREL
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NODE3 TYPE = INTEGER UNITS = DIMENSIONLESS
THIRD NODE IN ELEMENT CONNECTIVITY
RELATIONS USED IN: QUADS, TRIANGLS, SCRREL
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NODE4 TYPE = INTEGER UNITS = DIMENSIONLESS
FOURTH NODE IN ELEMENT CONNECTIVITY
RELATIONS USED IN: QUADS, SCRREL
ORIGIN: LOAD BULK DATA, SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NODENUM TYPE = INTEGER UNITS = DIMENSIONLESS
NODE NUMBER
RELATIONS USED IN: ACTUATOR, CONMASS, CONSTRN, EIGNVECT,NODES,NODELOAD, EXTFTAP, MAT-PROP, SENSOR
ORIGIN: SUPERTAB, MSC/NASTRAN, DEFINE CONTROL SYSTEM
DESTINATION: MSC/NASTRAN, SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES

NODEREF TYPE = INTEGER UNITS = DIMENSIONLESS
NODE NUMBER DEFINING BEAM ORIENTATION PLANE
RELATIONS USED IN: BEAMREF
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NOM-SIZE TYPE = TEXT 8 UNITS = DIMENSIONLESS
PHYSICAL PROPERTY NAME OR ID NUMBER
RELATIONS USED IN: BEAMS,SCRREL
ORIGIN: LOAD BULK DATA, SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NONSTWHB TYPE = REAL UNITS = M/L
NON-STRUCTURAL MASS PER UNIT LENGTH
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NONSTWHT TYPE = REAL UNITS = M/L**2
NON-STRUCTURAL MASS PER UNIT AREA
RELATIONS USED IN: QUADS, SCRREL, TRIANGLS
ORIGIN: LOAD BULK DATA, SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

NUMMODES TYPE = INTEGER UNITS = DIMENSIONLESS
NUMBER OF MODES
RELATIONS USED IN: EVECINFO
ORIGIN: CURRENTLY UNUSED
DESTINATION: CURRENTLY UNUSED

NUMNODES TYPE = INTEGER UNITS = DIMENSIONLESS
NUMBER OF NODES
RELATIONS USED IN: EVECINFO
ORIGIN: CURRENTLY UNUSED
DESTINATION: CURRENTLY UNUSED

OPTP TYPE = TEXT 40 UNITS = DIMENSIONLESS
MSC/NASTRAN OLD PROBLEM TAPE
(NAME OF NPTP CREATED BY NORMAL MODES ANALYSIS)
RELATIONS USED IN: EVECINFO
ORIGIN: LOAD ANALYSIS RESULTS
DESTINATION: USER'S INFORMATION

PA TYPE = INTEGER UNITS = DIMENSIONLESS
PIN FLAG END A
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

PAT-ELEM TYPE = TEXT 8 UNITS = DIMENSIONLESS
PATRAN ELEMENT TYPE
RELATIONS USED IN: ELEM-MAP
ORIGIN: CREATE DATABASE
DESTINATION: CURRENTLY UNUSED

PB TYPE = INTEGER UNITS = DIMENSIONLESS
PIN FLAG END B
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN

PERCENT TYPE = REAL UNITS = DIMENSIONLESS
PERCENT OF TOTAL STRAIN ENERGY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

PHONE TYPE = TEXT 15 UNITS = DIMENSIONLESS
PHONE NUMBER OF RP (RESPONSIBLE PERSON)
RELATIONS USED IN: DBDESC
ORIGIN: CREATE DATABASE
DESTINATION: USER'S INFORMATION

PRESLOAD TYPE = REAL UNITS = M/(LT**2)
ELEMENT PRESSURE LOAD
RELATIONS USED IN: ELEMLOAD
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

PROJECT TYPE = TEXT 12 UNITS = DIMENSIONLESS
PROJECT NAME
RELATIONS USED IN: DBDESC
ORIGIN: CREATE DATABASE
DESTINATION: USER'S INFORMATION

PUNCHFIL TYPE = TEXT = 40 UNITS = DIMENSIONLESS
MSC/NASTRAN PUNCH FILE CONTAINING EIGENVECTORS
AND CHECKPOINT DICTIONARY
RELATIONS USED IN: EVECINFO
ORIGIN: LOAD ANALYSIS RESULTS
DESTINATION: MSC/NASTRAN

RIMNUM TYPE = INTEGER UNITS = DIMENSIONLESS
IMAT ELEMENT NUMBERING COUNTER
RELATIONS USED IN: SCRREL
ORIGIN: FORMAT BULK DATA, LOAD BULK DATA
DESTINATION: INTERNAL USE ONLY

ROWNUM	TYPE = INTEGER	UNITS = DIMENSIONLESS
ROW NUMBER		
RELATIONS USED IN:	EIGNVALS, NODES	
ORIGIN:	MSC/NASTRAN, DEFINE CONTROL SYSTEM	
DESTINATION:	GENERATE LINEAR SYSTEMS MATRICES	
RP	TYPE = TEXT 12	UNITS = DIMENSIONLESS
RESPONSIBLE PERSON		
RELATIONS USED IN:	DBDESC	
ORIGIN:	CREATE DATABASE	
DESTINATION:	USER'S INFORMATION	
SECTTYPE	TYPE = TEXT 8	UNITS = DIMENSIONLESS
SECTION TYPE		
RELATIONS USED IN:	SCRREL	
ORIGIN:	FORMAT BULK DATA, LOAD BULK DATA	
DESTINATION:	MSC/NASTRAN	
SECNUM	TYPE = INTEGER	UNITS = DIMENSIONLESS
SECTION NUMBER		
RELATIONS USED IN	SCRREL	
ORIGIN:	FORMAT BULK DATA, LOAD BULK DATA	
DESTINATION:	MSC/NASTRAN	
SENDOC	TYPE = TEXT	UNITS = DIMENSIONLESS
DESCRIPTION OF SENSOR		
RELATIONS USED IN:	SENSOR	
ORIGIN:	DEFINE CONTROL SYSTEM	
DESTINATION:	USER'S INFORMATION	
SENSID	TYPE = INTEGER	UNITS = DIMENSIONLESS
SENSOR ID NUMBER		
RELATIONS USED IN:	SENSOR	
ORIGIN:	DEFINE CONTROL SYSTEM	
DESTINATION:	GENERATE LINEAR SYSTEMS MATRICES	
SENSORS	TYPE = IVEC VAR	UNITS = DIMENSIONLESS
ID NUMBERS (SENSID'S) OF SENSORS		
RELATIONS USED IN:	SYSTEM	
ORIGIN:	DEFINE CONTROL SYSTEM	
DESTINATION	GENERATE LINEAR SYSTEMS MATRICES	
SIMDATE	TYPE = DATE	UNITS = DIMENSIONLESS
DATE OF SIMULATION		
RELATIONS USED IN:	SIMULATE	
ORIGIN:	RECORD CONTROL SIMULATION	
DESTINATION:	RECOVER PHYSICAL OUTPUT	

SIMDOC TYPE = TEXT 80 UNITS = DIMENSIONLESS
DESCRIPTION OF SIMULATION
RELATIONS USED IN: SIMULATE
ORIGIN: RECORD CONTROL SIMULATION
DESTINATION: USER'S INFORMATION

SIMID TYPE = INTEGER UNITS = DIMENSIONLESS
SIMULATION ID NUMBER
RELATIONS USED IN: DYNFORCE
ORIGIN: RECORD CONTROL SIMULATION
DESTINATION: RECOVER PHYSICAL OUTPUT

SPARNUM TYPE = REAL UNITS = DIMENSIONLESS
ELEMENT NUMBERING COUNTER
REALTIONS USED IN: SCRREL
ORIGIN: FORMAT BULK DATA, LOAD BULK DATA
DESTINATION: INTERNAL USE ONLY

SPEC-WT TYPE = REAL UNITS = M/L**3
SPECIFIC MASS
RELATIONS USED IN: MAT-PROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

SPECHEAT TYPE = REAL UNITS = BTU(LB-DEG F)
SPECIFIC HEAT
RELATIONS USED IN: MAT-PROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: CURRENTLY UNUSED

STENB TYPE = REAL UNITS = ML**2/T**2
BENDING STRAIN ENERGY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

STENBD TYPE = REAL UNITS = M/(LT**2)
BENDING STRAIN ENERGY DENSITY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

STENMB TYPE = REAL UNITS = ML**2/T**2
MEMBRANE-BENDING STRAIN ENERGY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

STENMBD TYPE = REAL UNITS = M/(LT**2)
MEMBRANE-BENDING STRAIN ENERGY DENSITY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

STENMD TYPE = REAL UNITS = M/(LT**2)
MEMBRANE STRAIN ENERGY DENSITY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USERS INFORMATION

STENT TYPE = REAL UNITS = ML**2/T**2
TOTAL STRAIN ENERGY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

STEND TYPE = REAL UNITS = M/(LT**2)
TOTAL STRAIN ENERGY DENSITY
RELATIONS USED IN: STRNENG
ORIGIN: MSC/NASTRAN
DESTINATION: USER'S INFORMATION

STRNCOL TYPE = INTEGER UNITS = DIMENSIONLESS
COLOR FOR A GROUP OF ELEMENTS
RELATIONS USED IN: STRNENG
ORIGIN: CURRENTLY UNUSED
DESTINATION: CURRENTLY UNUSED

SYSDOC TYPE = TEXT 80 UNITS = DIMENSIONLESS
DESCRIPTION OF SYSTEM
RELATIONS USED IN: SYSTEM
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: USER'S INFORMATION

SYSID TYPE = INTEGER UNITS = DIMENSIONLESS
SYSTEM ID NUMBER
RELATIONS USED IN: SIMULATE, SYSTEM
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES,
RECORD CONTROL SIMULATION

TEMP TYPE = REAL UNITS = DEG F
NODAL TEMPERATURE
RELATIONS USED IN: NODELOAD
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

TEMPTYPE	TYPE = TEXT 8	UNITS = DIMENSIONLESS
TEMPORARY TYPE		
RELATIONS USED IN:	SCRREL	
ORIGIN:	FORMAT BULK DATA, LOAD BULK DATA	
DESINATION:	FORMAT BULK DATA, LOAD BULK DATA	

THERMCOE	TYPE = REAL	UNITS = 1/DEG F
COEFFICIENT OF THERMAL EXPANSION		
RELATIONS USED IN:	MAT-PROP	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	

THETA	TYPE = REAL	UNITS = DEG
INCLINATION OF PRINCIPAL AXIS WITH ELEMENT REFERENCE FRAME		
RELATIONS USED IN:	BEAMPROP	
ORIGIN:	CURRENTLY UNUSED	
DESTINATION:	CURRENTLY UNUSED	

THICKNES	TYPE = REAL	UNITS = L
PLATE OR SHELL THICKNESS		
RELATIONS USED IN:	QUADS, SCRREL, TRIANGLS	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	

TFINAL	TYPE = REAL	UNITS = T
SIMULATION END TIME		
RELATIONS USED IN:	DYNFORCE, SIMULATE	
ORIGIN:	RECORD CONTROL SIMULATION	
DESTINATION:	RECOVER PHYSICAL OUTPUT	

TINIT	TYPE = REAL	UNITS = T
SIMULATION START TIME		
ORIGIN:	RECORD CONTROL SIMULATION	
DESTINATION:	RECOVER PHYSICAL OUTPUT	

TYPE	TYPE = INTEGER	UNITS = DIMENSIONLESS
1 = CONTROL FORCE/TORQUE, 2 = APPLIED LOAD (EXTERNAL FORCE/TORQUE)		
RELATIONS USED IN:	DYNFORCE	
ORIGIN:	RECORD CONTROL SIMULATION	
DESTINATION:	RECOVER PHYSICAL OUTPUT	

ULT-COMP	TYPE = REAL	UNITS = M/(LT**2)
ULTIMATE COMPRESSIVE STRENGTH		
RELATIONS USED IN: MAT-PROP		
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	CURRENTLY UNUSED	

ULT-SHER TYPE = REAL UNITS = M/(LT**2)
ULTIMATE SHEAR STRENGTH
RELATIONS USED IN: MAT-PROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: CURRENTLY UNUSED

ULT-TENS TYPE = REAL UNITS = M/(LT**2)
ULTIMATE TENSILE STRENGTH
RELATIONS USED IN: MAT-PROP
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: CURRENTLY UNUSED

UNITS TYPE = TEXT 40 UNITS = DIMENSIONLESS
DESCRIPTION OF UNITS USED IN DATABASE
RELATONS USED IN: DBDESC
ORIGIN: CREATE DATABASE
DESTINATION: USER'S INFORMATION

USAGE TYPE = INTEGER UNITS = DIMENSIONLESS
-1 FOR UNSPECIFIED; OR SUM OF APPLICABLE: 1 FOR POSITION OR ANGLE, 2 FOR VELOCITY OR ANGULAR RATE, 4 FOR ACCELERATION OR ANGULAR ACCELERATION
RELATIONS USED IN: SENSOR
ORIGIN: DEFINE CONTROL SYSTEM
DESTINATION: GENERATE LINEAR SYSTEMS MATRICES

X TYPE = REAL UNITS = L
X-LOCATION
RELATIONS USED IN: NODES
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN,SUPERTAB
ESTINATION: SUPERTAB

X-EIGN TYPE = REAL UNITS = L
X COMPONENT OF THE EIGENVECTOR
RELATIONS USED IN: EIGNVECT
ORIGIN: MSC/NASTRAN
DESTINATION: SUPERTAB, GENERATE LINEAR SYSTEM MATRICES

X-ROTEN TYPE = REAL UNITS = RAD
X-AXIS ROTATIONAL COMPONENT OF THE EIGENVECTOR
RELATIONS USED IN: EIGNVECT
ORIGIN: LOAD ANALYSIS RESULTS
DESTINATION: SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES

XDCON TYPE = TEXT 4 UNITS = DIMENSIONLESS
CONSTRAINED IN X-DISPLACEMENT? Y OR N
RELATIONS USED IN: CONSTRN
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

XRCON	TYPE = TEXT 4	UNITS = DIMESNIONLESS
CONSTRAINED IN X-ROTATION? Y OR N		
RELATIONS USED IN	CONSTRN	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	
Y	TYPE = REAL	UNITS = L
Y- LOCATION		
RELATIONS USED IN:	NODES	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	
Y-EIGN	TYPE = REAL	UNITS = L
Y COMPONENT OF THE EIGENVECTOR		
RELATIONS USED IN:	EIGNVECT	
ORIGIN:	MSC/NASTRAN	
DESTINATION:	SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES	
Y-ROTEN	TYPE = REAL	UNITS = RAD
Y-AXIS ROTATIONAL COMPONENT OF THE EIGENVECTOR		
RELATIONS USED IN:	EIGNVECT	
ORIGIN:	MSC/NASTRAN	
DESTINATION:	SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES	
YDCON	TYPE = TEXT 4	UNITS = DIMENSIONLESS
CONSTRAINED IN Y-DISPLACEMENT? Y OR N		
RELATIONS USED IN:	CONSTRN	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	
YIELD-CO	TYPE = REAL	UNITS = M/(LT**2)
YIELD STRENGTH IN COMPRESSION		
RELATIONS USED IN:	MAT-PROP	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	
YIELD-SH	TYPE = REAL	UNITS = M/(LT**2)
YIELD STRENGTH IN SHEAR		
RELATIONS USED IN:	MAT-PROP	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	
YIELD-TE	TYPE = REAL	UNITS = M/(LT**2)
YIELD STRENGTH IN TENSION		
RELATIONS USED IN:	MAT-PROP	
ORIGIN:	SUPERTAB, MSC/NASTRAN	
DESTINATION:	MSC/NASTRAN, SUPERTAB	

YRCON TYPE = TEXT 4 UNITS = DIMENSIONLESS
CONSTRAINED IN Y-ROTATION? Y OR N
RELATIONS USED IN CONSTRN
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z TYPE = REAL UNITS = L
Z-LOCATION
RELATIONS USED IN: NODES,
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z-EIGN TYPE = REAL UNITS = L
Z COMPONENT OF THE EIGENVECTOR
RELATIONS USED IN: EIGNVECT
ORIGIN: MSC/NASTRAN
DESTINATION: SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES

Z-ROTEN TYPE = REAL UNITS = RAD
Z-AXIS ROTATIONAL COMPONENT OF THE EIGENVECTOR
RELATIONS USED IN: EIGNVECT
ORIGIN: MSC/NASTRAN
DESTINATION: SUPERTAB, GENERATE LINEAR SYSTEMS MATRICES

Z1 TYPE = REAL UNITS = L
SHEAR CENTER OFFSETS
RELATIONS USED IN: BEAMPROP
ORIGIN: CURRENTLY UNUSED
DESTINATION: CURRENTLY UNUSED

Z1A TYPE = REAL UNITS = L
X-OFFSET OF SHEAR CENTER AT END A
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z1B TYPE = REAL UNITS = L
X-OFFSET OF SHEAR CENTER AT END B
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z1M TYPE = REAL UNITS = L
X-OFFSET DISTANCE OF MASS
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z2 TYPE = REAL UNITS = L
SHEAR CENTER OFFSETS
RELATIONS USED IN: BEAMPROP
ORIGIN: CURRENTLY UNUSED
DESTINATION: CURRENTLY UNUSED

Z2A TYPE = REAL UNITS = L
Y-OFFSET OF SHEAR CENTER AT END A
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z2B TYPE = REAL UNITS = L
Y-OFFSET OF SHEAR CENTER AT END B
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z2M TYPE = REAL UNITS = L
Y-OFFSET DISTANCE OF MASS
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z3A TYPE = REAL UNITS = L
Z-OFFSET OF SHEAR CENTER AT END A
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z3B TYPE = REAL UNITS = L
Z-OFFSET OF SHEAR CENTER AT END B
RELATIONS USED IN: BEAMS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Z3M TYPE = REAL UNITS = L
Z-OFFSET DISTANCE OF MASS
RELATIONS USED IN: CONMASS
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

ZDCON TYPE = TEXT 4 UNITS = DIMENSIONLESS
CONSTRAINED IN Z-DISPLACEMENT? Y OR N
RELATIONS USED IN: CONSTRN
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

ZRCON TYPE = TEXT 4 UNITS = DIMENSIONLESS
CONSTRAINED IN Z-ROTATION? Y OR N
RELATIONS USED IN: CONSTRN
ORIGIN: SUPERTAB, MSC/NASTRAN
DESTINATION: MSC/NASTRAN, SUPERTAB

Schema 1

ACTUATOR
ACTUATOR INFORMATION

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	ACID	INTEGER	ACTUATOR ID NUMBER	DIMENSIONLESS
2	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
3	FORCTORQ	RVEC 6	X, Y, Z FORCE; X, Y, Z TORQUE RESULTING FROM A UNIT INPUT	DIMENSIONLESS
4	ACDOC	TEXT 80	DESCRIPTION OF ACTUATOR	DIMENSIONLESS

BEAMPROP
BEAM PROPERTIES

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	NOM-SIZE	TEXT 8	PHYSICAL PROPERTY NAME OR ID NUMBER	DIMENSIONLESS
2	I1	REAL	PRINCIPAL MOMENT OF INERTIA, PLANE 1 (IZZ)	L**4
3	ALPHA1	REAL	TRANVERSE SHEAR DEFLECTION CONSTANT, PLANE 1	DIMENSIONLESS
4	I2	REAL	PRINCIPAL MOMENT OF INERTIA, PLANE 2 (IYY)	L**4
5	ALPHA2	REAL	TRANSVERSE SHEAR DEFLECTION CONSTANT, PLANE 2	DIMENSIONLESS
6	AREA	REAL	CROSS-SECTIONAL AREA	L**2
7	FF	REAL	UNIFORM TORSION CONSTANT	DIMENSIONLESS
8	F1	REAL	NON-UNIFORM TORSION CONSTANT	DIMENSIONLESS
9	Z1	REAL	SHEAR CENTER OFFSETS	L
10	Z2	REAL	SHEAR CENTER OFFSETS	L
11	THETA	REAL	INCLINATION OF PRINCIPAL AXIS WITH ELEMENT REFERENCE FRAME	DEG
12	I12A	REAL	AREA PRODUCT OF INERTIA	L**4

Schema 2

BEAMREF
BEAM REFERENCE

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	------------	-------	-------------------	--------
1	BMREFER	INTEGER	BEAM-ORIENTATION REFERENCE NUMBER	DIMENSIONLESS
2	X	REAL	X-LOCATION	L
3	Y	REAL	Y-LOCATION	L
4	Z	REAL	Z-LOCATION	L
5	NODEREF	INTEGER	NODE NUMBER DEFINING BEAM ORIENTATION PLANE	DIMENSIONLESS

BEAMS
TWO NODE ELEMENTS

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	------------	-------	-------------------	--------
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	ELEMENT	INTEGER	ELEMENT NUMBER	DIMENSIONLESS
3	NODE1	INTEGER	FIRST NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
4	NODE2	INTEGER	SECOND NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
5	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
6	NOM-SIZE	TEXT 8	PHYSICAL PROPERTY OR ID NUMBER	DIMENSIONLESS
7	MATERIAL	TEXT 8	MATERIAL NAME OR ID NUMBER	DIMENSIONLESS
8	BMREFER	INTEGER	BEAM-ORIENTATION REFERENCE NUMBER	DIMENSIONLESS
9	NONSTWHB	REAL	NON-STRUCTURAL MASS PER UNIT LENGTH	M/L
10	Z1A	REAL	X-OFFSET OF SHEAR CENTER-END A	L
11	Z2A	REAL	Y-OFFSET OF SHEAR CENTER-END A	L
12	Z3A	REAL	Z-OFFSET OF SHEAR CENTER-END A	L
13	Z1B	REAL	X-OFFSET OF SHEAR CENTER-END B	L
14	Z2B	REAL	Y-OFFSET OF SHEAR CENTER-END B	L
15	Z3B	REAL	Z-OFFSET OF SHEAR CENTER-END B	L
16	PA	INTEGER	PIN FLAG END A	DIMENSIONLESS
17	PB	INTEGER	PIN FLAG END B	DIMENSIONLESS

Schema 3

CONMASS
CONCENTRATED MASS

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	-------------------	--------
1	ELEMENT	INTEGER	ELEMENT NUMBER	DIMENSIONLESS
2	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
3	MASS	REAL	MASS	M
4	I11	REAL	MASS MOMENT OF INERTIA	ML**2
5	I21	REAL	MASS MOMENT OF INERTIA	ML**2
6	I22	REAL	MASS MOMENT OF INERTIA	ML**2
7	I31	REAL	MASS MOMENT OF INERTIA	ML**2
8	I32	REAL	MASS MOMENT OF INERTIA	ML**2
9	I33	REAL	MASS MOMENT OF INERTIA	ML**2
10	Z1M	REAL	X-OFFSET DISTANCE OF MASS	L
11	Z2M	REAL	Y-OFFSET DISTANCE OF MASS	L
12	Z3M	REAL	Z-OFFSET DISTANCE OF MASS	L
13	CID	INTEGER	COORDINATE FRAME ID	DIMENSIONLESS

CONSTRN
NODAL CONSTRAINTS

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	-------------------	--------
1	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
2	XDCON	TEXT 4	CONSTRAINED IN X-DISPLACEMENT Y OR N	DIMENSIONLESS
3	YDCON	TEXT 4	CONSTRAINED IN Y-DISPLACEMENT Y OR N	DIMENSIONLESS
4	ZDCON	TEXT 4	CONSTRAINED IN Z-DISPLACEMENT Y OR N	DIMENSIONLESS
5	XRCON	TEXT 4	CONSTRAINED IN X-ROTATION Y OR N	DIMENSIONLESS
6	YRCON	TEXT 4	CONSTRAINED IN Y-ROTATION Y OR N	DIMENSIONLESS
7	ZRCON	TEXT 4	CONSTRAINED IN Z-ROTATION Y OR N	DIMENSIONLESS

Schema 4

DBDESC
DATABASE DESCRIPTION

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	PROJECT	TEXT 12	PROJECT NAME	DIMENSIONLESS
2	RP	TEXT 12	RESPONSIBLE PERSON (DATABASE OWNER)	DIMENSIONLESS
3	PHONE	TEXT 15	PHONE NUMBER OF RP	DIMENSIONLESS
4	DATECRE	DATE	DATE DATABASE WAS CREATED	DIMENSIONLESS
5	DLASTMOD	DATE	DATE OF LAST MODIFICATION	DIMENSIONLESS
6	DBDESC	TEXT 80	DATABASE DESCRIPTION	DIMENSIONLESS
7	CSYSTYPE	TEXT 80	TYPE OF CONTROL SYSTEM MODELED	DIMENSIONLESS
8	UNITS	TEXT 40	DESCRIPTION OF UNITS USED	DIMENSIONLESS

DYNFORCE
SIMULATION INFORMATION

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	SIMID	INTEGER	SIMULATION ID NUMBER	DIMENSIONLESS
2	DLOADFIL	TEXT 40	LOAD/FORCE TIME HISTORY FILE	DIMENSIONLESS
3	TYPE	INTEGER	1 = CONTROL FORCE/TORQUE 2 = APPLIED LOAD (EXTERNAL FORCE/TORQUE)	DIMENSIONLESS
4	FORCID	INTEGER	ID NUMBER (ACID) OF CONTROL FORCE OR ID NUMBER (EXPID) OF APPLIED LOAD	DIMENSIONLESS
5	TINIT	REAL	SIMULATION START TIME	T
6	TFINAL	REAL	SIMULATION END TIME	T
7	DELTAT	REAL	UNIFORM TIME STEP	T

EIGNVALS
EIGENVALUE DATA

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	ROWNUM	INTEGER	ROW NUMBER	DIMENSIONLESS
2	CONDNUM	INTEGER	LOAD-CASE ID OR MODE NUMBER	DIMENSIONLESS
3	FREQRPS	REAL	CIRCULAR FREQUENCY	RAD/T
4	FREQHZ	REAL	CYCLIC FREQUENCY (HERTZ)	1/T
5	DAMPRAT	REAL	DAMPING RATIO	
6	EIGNDOC	TEXT 80	DESCRIPTION OF MODE OR FREQUENCY	DIMENSIONLESS

Schema 5

EIGNVECT
EIGENVECTOR DATA

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
2	X-EIGN	REAL	X COMPONENT OF THE EIGENVECTOR	L
3	Y-EIGN	REAL	Y COMPONENT OF THE EIGENVECTOR	L
4	Z-EIGN	REAL	Z COMPONENT OF THE EIGENVECTOR	L
5	X-ROTEN	REAL	X-AXIS ROTATIONAL COMPONENT OF THE EIGNVECTOR	RAD
6	Y-ROTEN	REAL	Y-AXIS ROTATIONAL COMPONENT OF THE EIGENVECTOR	RAD
7	Z-ROTEN	REAL	Z-AXIS ROTATIONAL COMPONENT OF THE EIGENVECTOR	RAD
8	CONDNUM	INTEGER	LOAD-CASE ID OR MODE NUMBER	DIMENSIONLESS

ELEMLOAD
ELEMENT PRESSURE LOADS

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	ELEMENT	INTEGER	ELEMENT NUMBER	DIMENSIONLESS
3	PRESLOAD	REAL	ELEMENT PRESSURE LOAD	M/(LT**2)
4	CONDNUM	INTEGER	LOAD-CASE ID OR MODE NUMBER	DIMENSIONLESS

Schema 6

EVECINFO
EIGENVECTOR INFORMATION

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	NUMNODES	INTEGER	NUMBER OF NODES	DIMENSIONLESS
2	NUMMODES	INTEGER	NUMBER OF MODES	DIMENSIONLESS
3	EVECFNAM	TYPE 40	RAPID ACCESS EIGENVECTOR FILE	DIMENSIONLESS
4	PUNCHFIL	TEXT 40	NASTRAN PUNCH FILE CONTAINING EIGENVECTORS AND CHECKPOINT DICTIONARY FOR RESTART	DIMENSIONLESS
5	OPTP	TEXT 40	NASTRAN OLD PROBLEM TAPE FROM NORMAL MODES (SOL> 3)	DIMENSIONLESS

EXTFTAP
EXTERNAL FORCE/TORQUE DATA

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	EXAPID	INTEGER	EXTERNAL FORCE/TORQUE ID NUMBER	DIMENSIONLESS
2	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
3	FORCTORQ	RVEC 6	COMPONENTS OF FORCTORQ ARE: X,Y,Z FORCE AND X,Y,Z TORQUE RESULTING FROM UNIT INPUT	DIMENSIONLESS
4	EXAPDOC	TEXT VAR	DESCRIPTION OF EXTERNAL FORCE/TORQUE	DIMENSIONLESS

ELEM-MAP
A SCRATCH RELATION

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
2	EAL-ELT	TEXT 8	EAL ELEMENT TYPE	DIMENSIONLESS
3	EAL-SEC	TEXT 8	EAL SECTION TYPE	DIMENSIONLESS
4	COS-NAST	TEXT 8	COSMIC NASTRAN ELEMENT TYPE	DIMENSIONLESS
5	COS SECT	TEXT 8	COSMIC NASTRAN SECTION TYPE	DIMENSIONLESS
6	MSC-NAST	TEXT 8	MSC/NASTRAN ELEMENT TYPE	DIMENSIONLESS
7	MSC-SECT	TEXT 8	MSC/NASTRAN SECTION TYPE	DIMENSIONLESS
8	PAT-ELEM	TEXT 8	PATRAN ELEMENT TYPE	DIMENSIONLESS

Schema 7

MAT-PROP
MATERIAL PROPERTIES

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	--------------------	--------
1	MATERIAL	TEXT 8	MATERIAL NAME OR ID NUMBER	DIMENSIONLESS
2	SPEC-WT	REAL	SPECIFIC MASS DENSITY	M/L**3
3	YIELD-TE	REAL	YIELD STRENGTH IN TENSION	M/(LT**2)
4	YIELD-CO	REAL	YIELD STRENGTH IN COMPRESSION	M/(LT**2)
5	YIELD-SH	REAL	YIELD STRENGTH IN SHEAR	
6	ULT-TENS	REAL	ULTIMATE TENSILE STRENGTH	M/(LT**2)
7	ULT-COMP	REAL	ULTIMATE COMPRESSIVE STRENGTH	M/(LT**2)
8	ULT-SHER	REAL	ULTIMATE SHEAR STRENGTH	M/(LT**2)
9	ENDU-LIM	REAL	ENDURANCE LIMIT	M/(LT**2)
10	MOD-ELAS	REAL	MODULUS OF ELASTICITY	M/(LT**2)
11	MOD-RIGI	REAL	MODULUS OF RIGIDITY	M/(LT**2)
12	THERMCOE	REAL	COEFFICIENT OF THERMAL EXPANSION	1/DEG F
13	SPECHEAT	REAL	SPECIFIC HEAT	BTU/(LB-DEG F)
14	CONDUCT	REAL	CONDUCTIVITY (IN-SEC-DEG F)	BTU
15	EMISSIV	REAL	EMISSIVITY	DIMENSIONLESS
16	ABSORP	REAL	ABSORPTIVITY	DIMENSIONLESS
17	DIFFCOMP	REAL	DIFFUSE COMPONENT	DIMENSIONLESS

MODELDEF
MODEL DEFINITION

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	--------------------	--------
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
3	DESCRIPT	TEXT 50	TEXT DESCRIPTION FOR THIS ELEMENT TYPE AND GROUP NUMBER	DIMENSIONLESS

NODELOAD
NODAL TEMPERATURE AND LOADS

ATTRIBUTE		TYPE	DESCRIPTION	UNITS
-------------		-------	--------------------	--------
1	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
2	FMDIRECT	INTEGER	FORCE OR MOMENT DIRECTION (1-6)	DIMENSIONLESS
3	APPLFOR	REAL	APPLIED FORCE OR MOMENT	ML/T**2 OR ML**2/T**2
4	TEMP	REAL	NODAL TEMPERATURE	DEG F
5	CONDNUM	INTEGER	LOAD-CASE ID OR MODE NUMBER	DIMENSIONLESS

Schema 8

NODES
NODE NUMBERS AND COORDINATES

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	ROWNUM	INT	ROW NUMBER	DIMENSIONLESS
2	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
3	X	REAL	X-LOCATION	L
4	Y	REAL	Y-LOCATION	L
5	Z	REAL	Z-LOCATION	L

QUADS
FOUR NODE ELEMENTS

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	ELEMENT	INTEGER	ELEMENT NUMBER	DIMENSIONLESS
3	NODE1	INTEGER	FIRST NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
4	NODE2	INTEGER	SECOND NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
5	NODE3	INTEGER	THIRD NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
6	NODE4	INTEGER	FOURTH NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
7	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
8	MATERIAL	TEXT 8	MATERIAL NAME OR ID NUMBER	DIMENSIONLESS
9	THICKNES	REAL	PLATE OR SHELL THICKNESS	L
10	NONSTWHT	REAL	NON-STRUCTURAL MASS PER UNIT AREA	M/L**2

RIGPROP
RIGID-BODY MASS PROPERTIES

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	MASSMAT	RMAT 3,3	RIGID-BODY MASS MATRIX	M
2	MASMOMIN	RMAT 3,3	RIGID-BODY MASS MOMENT OF INERTIA MATRIX	ML**2
3	CG	RVEC 3	CENTER OF MASS X,Y,Z	L

Schema 9

RIGIDBAR
MSC/NASTRAN RIGID ELEMENTS (RBARS)

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	ELEMENT	INTEGER	ELEMENT NUMBER	DIMENSIONLESS
3	NODE1	INTEGER	FIRST NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
4	NODE2	INTEGER	SECOND NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
5	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
6	INDDOFA	INTEGER	INDEPENDENT DOF AT END A	DIMENSIONLESS
7	INDDOFB	INTEGER	INDEPENDENT DOF AT END B	DIMENSIONLESS
8	DEPDOFA	INTEGER	DEPENDENT DOF AT END A	DIMENSIONLESS
9	DEPDOFB	INTEGER	DEPENDENT DOF AT END B	DIMENSIONLESS

SCRREL
A SCRATCH RELATION

ATTRIBUTE		TYPE	DESCRIPTION	UNITS
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	RIMNUM	INTEGER	RIM ROW NUMBER	DIMENSIONLESS
3	SPARNUM	INTEGER	MISCELLENEOUS COUNTER	DIMENSIONLESS
4	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
5	ELD	TEXT 8	ELEMENT	DIMENSIONLESS
6	NODE1	INTEGER	FIRST NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
7	NODE2	INTEGER	SECOND NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
8	NODE3	INTEGER	THIRD NODE IN ELEMENT CONNECTIVITY	DIEMSNIONLESS
9	NODE4	INTEGER	FOURTH NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
10	TEMPTYPE	TEXT 8	TEMPORARY TYPE	DIMENSIONLESS
11	MATNUM	INTEGER	MATERIAL NUMBER	DIMENSIONLESS
12	SECTYPE	TEXT 8	SECTION TYPE	DIMENSIONLESS
13	SECNUM	INTEGER	SECTION NUMBER	DIMENSIONLESS
14	NOM-SIZE	TEXT 8	PHYSICAL PROPERTY NAME OR ID NUMBER	DIMENSIONLESS
15	THICKNES	REAL	PLATE OR SHELL THICKNESS	L
16	BMREFER	INTEGER	BEAM-ORIENTATION REFERENCE NUMBER	DIMENSIONLESS
17	NONSTWHT	REAL	NON-STRUCTURAL MASS PER UNIT AREA	M/L**2

Schema 10

SENSOR
SENSOR INFORMATION

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	-------------------	--------
1	SENSID	INTEGER	SENSOR ID NUMBER	DIMENSIONLESS
2	NODENUM	INTEGER	NODE NUMBER	DIMENSIONLESS
3	AXIS	INTEGER	1,2,3 FOR X,Y,Z TRANSLATION 4,5,6 FOR X,Y,Z ROTATION	DIMENSIONLESS
4	USAGE	INTEGER	-1 FOR UNSPECIFIED OR SUM OF APPLICABLE, 1 FOR POSITION, 2 FOR VELOCITY (OR ANGULAR RATE), 4 FOR ACCELERATION (OR ANGULAR ACCELERATION)	DIMENSIONLESS
5	SENDOC	TEXT 80	DESCRIPTION OF SENSOR	DIMENSIONLESS

SIMULATE
SIMUALTION INFORMATION

1	SIMID	INTEGER	SIMULATION ID NUMBER	DIMENSIONLESS
2	SIMDOC	TEXT 80	DESCRIPTION OF SIMULATION	DIMENSIONLESS
3	SYSID	INTEGER	CONTROL SYSTEM ID NUMBER	DIMESNIONLESS
4	SIMDATE	DATE	DATE OF SIMULATION	DIMENSIONLESS
5	ANALYST	TEXT 40	NAME OF CONTROLS ANALYST	DIMENSIONLESS
6	TINIT	REAL	SIMULATION START TIME	T
7	TFINAL	REAL	SIMULATION END TIME	T
8	DELTAT	REAL	UNIFORM TIME STEP	T
9	DISPFILE	TEXT 40	MODAL DISPLACEMENT FILE	DIMENSIONLESS
10	VELOFILE	TEXT 40	MODAL VELOCITY FILE	DIMESNIONLESS
11	ACCEFILE	TEXT 40	MODAL ACCELERATION FILE	DIMENSIONLESS

Schema 11

STRNENG
STRAIN ENERGY DATA

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	-------------------	--------
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	ELEMENT	INTEGER	ELEMENT NUMBER	DIMENSIONLESS
3	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
4	STENT	REAL	TOTAL STRAIN ENERGY	ML**2/T**2
5	STEND	REAL	TOTAL STRAIN ENERGY DENSITY	M/(LT**2)
6	PERCENT	REAL	PERCENT OF TOTAL STRAIN ENERGY FOR ELEMENT	DIMENSIONLESS
7	STENMD	REAL	MEMBRANE STRAIN ENERGY DENSITY	M/(LT**2)
8	STENB	REAL	BENDING STRAIN ENERGY	ML**2/T**2
9	STENBD	REAL	BENDING STRAIN ENERGY DENSITY	M/(LT**2)
10	STENMB	REAL	MEMBRANE-BENDING STRAIN ENERGY	ML**2/T**2
11	STENMBD	REAL	MEMBRANE-BENDING STRAIN ENERGY DENSITY	M/(LT**2)
12	CONDNUM	INTEGER	LOAD-CASE ID OR MODE NUMBER	DIMENSIONLESS
13	STRNCOL	INTEGER	COLOR FOR A GROUP OF ELEMENTS	DIMENSIONLESS

Schema 12

SYSTEM
LINEAR SYSTEM PARAMETERS

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	--------------------	--------
1	SYSID	INTEGER	SYSTEM ID NUMBER	DIMENSIONLESS
2	MODES	IVEC VAR	VIBRATION(>0) OR NEWTONIAN RIGID-BODY(-1 TO -6) MODES	DIMENSIONLESS
3	CONTROLS	IVEC VAR	ID NUMBER(ACID'S) OF ACTUATORS	DIMENSIONLESS
4	SENSORS	IVEC VAR	ID NUMBER (SENSID'S) OF SENSORS	DIMENSIONLESS
5	EXFORTOR	IVEC VAR	ID NUMBERS (EXAPID'S)OF EXTERNAL FORCE/TORQUE	DIMENSIONLESS
6	SYSDOC	TEXT 80	DESCRIPTION OF SYSTEM	DIMENSIONLESS

TRIANGLS
THREE NODE ELEMENTS

	ATTRIBUTE	TYPE	DESCRIPTION	UNITS
	-------------	-------	--------------------	--------
1	CMPONENT	INTEGER	GROUP NUMBER	DIMENSIONLESS
2	ELEMENT	INTEGER	ELEMENT NUMBER	DIMENSIONLESS
3	NODE1	INTEGER	FIRST NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
4	NODE2	INTEGER	SECOND NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
5	NODE3	INTEGER	THIRD NODE IN ELEMENT CONNECTIVITY	DIMENSIONLESS
6	EL-TYPE	TEXT 8	ELEMENT TYPE	DIMENSIONLESS
7	MATERIAL	TEXT 8	MATERIAL NAME OR ID NUMBER	DIMENSIONLESS
8	THICKNES	REAL	PLATE OR SHELL THICKNESS	L
9	NONSTWHT	REAL	NON-STRUCTURAL MASS PER UNIT AREA	M/L**2

APPENDIX B

IMAT EXAMPLE PROBLEM

Introduction		B- 1
Phase I	Structural Analysis	B- 1
Phase II	Control and Design Analysis	B- 4
Phase III	Simulation	B- 11
Phase IV	Data Recovery	B- 13

IMAT Example Problem 1

Introduction

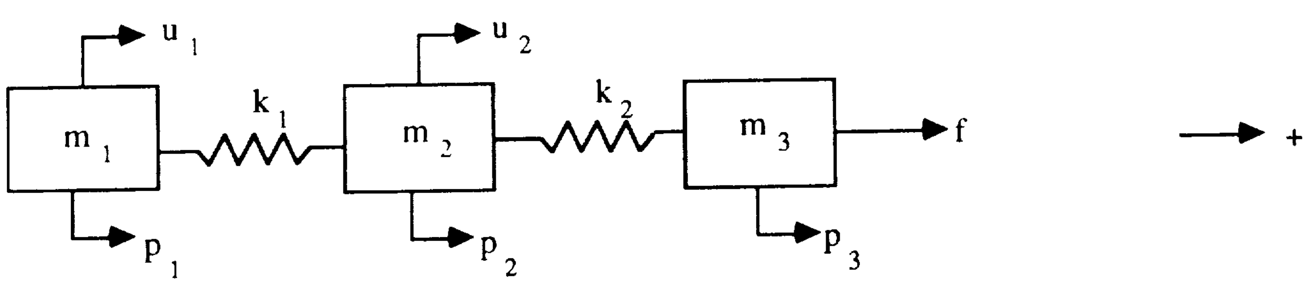
The physical system that is modeled in the IMAT example problem consists of three concentrated masses connected in series by linear springs (modeled with rod elements in IMAT). Modal proportional damping is used to account for the passive damping in the system. Each mass is free to translate along the horizontal axis only, and there are position and rate sensors attached to each mass. Actuators are attached to masses m_1 and m_2. In this example, control gains applied to the state variables are determined by an optimal control law. The objective of the optimal control is to minimize the transient response of the system to a disturbance, f, or to initial displacements without resorting to excessive control forces. Therefore a compromise is required in calculating feedback gains. This leads to an optimal regulator problem in which a set of constant gains are determined to minimize a specified performance index.

Step numbers refer to the steps presented in the Overview of the IMAT Solution Procedure.

Phase I. Structural Analysis

Steps 1-4. Create Finite Element Model

Mass-spring System



Notations

p: physical displacement coordinates (signal observation points)

m: masses

k: spring constants

u: control forces

f: disturbance force

NOTE: damping is accounted for by introducing proportional modal damping later in the problem formulation

Numerical Data:

$m_1 = 2,\ m_2 = 1,\ m_3 = 3$

$k_1 = 1,\ k_2 = 2$

Step 5. EXECUTE MSC/NASTRAN - Normal Modes Analysis

Equations of Motion for Undamped Free Vibration

$$M\ddot{p} + Kp = 0$$

where

$$M = \begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 3 \end{bmatrix}$$

$$K = \begin{bmatrix} k_1 & -k_1 & 0 \\ -k_1 & k_1+k_2 & -k_2 \\ 0 & -k_2 & k_2 \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 \\ -1 & 3 & -2 \\ 0 & -2 & 2 \end{bmatrix}$$

Solutions : ω_i = natural frequencies (rad/sec)

($\omega_i = \sqrt{\lambda_i}$, where λ_i is the ith eigenvalue of $[M - \lambda^2 K] = 0$)

Φ = matrix of undamped mode shapes (each column represents one mode shape, or eigenvector)

Normalize Mode Shapes to Satisfy

$$\Phi^T M \Phi = I$$

Numerical Results

$$[\omega] = \begin{bmatrix} \omega_1 & 0 & 0 \\ 0 & \omega_2 & 0 \\ 0 & 0 & \omega_3 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.7427 & 0 \\ 0 & 0 & 1.8974 \end{bmatrix}$$

$$\Phi = \begin{bmatrix} 0.4075 & 0.5575 & 0.1460 \\ 0.4075 & -0.0597 & -0.9093 \\ 0.4075 & -0.3518 & 0.2057 \end{bmatrix}$$

Step 6. Store Mode Shapes, Frequencies, and Mass Properties in IMAT-defined database.

Phase II. Control Design and Analysis

Step 7. Use DEFINE CONTROL SYSTEM processor to define control system

Define Generalized (Modal) Coordinates, q

$$p = \Phi q$$

Rewrite Equations of Motion, including Modal Damping, Control Forces, and Disturbances

$$\ddot{q} + [2\zeta\omega]\dot{q} + [\omega^2]q = \Phi^T B_c u(t) + \Phi^T B_f f(t)$$

where

$$[2\zeta\omega] = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0.0074 & 0 \\ 0 & 0 & 0.0190 \end{bmatrix} \quad \begin{pmatrix}\text{0.5\% critical damping} \\ \text{for each mode}\end{pmatrix}$$

$$B_c = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix}$$ Control Influence Matrix (controllers at m_1 and m_2)

$$B_f = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$ Disturbance Influence Matrix (applied external force at m_3)

Step 8. Run GENERATE LINEAR SYSTEMS MATRICES (GLSM) to create systems matrices

NOTE: GLSM writes these matrices in compressed form. You must use MATRIXx to expand them to the full form shown below.

$$\dot{x} = \hat{A}x + \hat{B}u$$

$$y = \hat{C}x + Du$$

where the state space vector **must** be defined as

$$x = \begin{bmatrix} q_1 & q_2 & q_3 & \dot{q}_1 & \dot{q}_2 & \dot{q}_3 \end{bmatrix}^T$$

$$\hat{A} = \left[\begin{array}{c|c} 0 & I \\ \hline -[\omega^2] & -[2\zeta\omega] \end{array}\right]$$

$$= \left[\begin{array}{ccc|ccc} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -0.5516 & 0 & 0 & -0.0074 & 0 \\ 0 & 0 & -3.6001 & 0 & 0 & -0.0190 \end{array}\right]$$

$$\hat{B} = \begin{bmatrix} 0 \\ \text{------} \\ \Phi^T B_c \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ \hline 0.4075 & 0.4075 \\ 0.5575 & -0.0597 \\ 0.1460 & -0.9093 \end{bmatrix}$$

$$\hat{C} = [\,\Phi \mid 0\,] = \left[\begin{array}{ccc|ccc} 0.4075 & 0.5575 & 0.1460 & 0 & 0 & 0 \\ 0.4075 & -0.0597 & -0.9093 & 0 & 0 & 0 \\ 0.4075 & -0.3518 & 0.2057 & 0 & 0 & 0 \end{array}\right]$$

$$D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix}$$

Step 9. Run MATRIXx to calculate control gain matrix (Skip this step if gains given or other method is desired)

Form system (S) matrix in MATRIXx as

$$S = \left[\begin{array}{c|c} \hat{A} & \hat{B} \\ \hline \hat{C} & D \end{array}\right]$$

IMAT Example Problem 7

Choose Cost Function

$$P = \int_0^\infty (x^T Q x + u^T R u)\,dt$$

where $Q = I$ (identity matrix)

$R = I$

Use REGULATOR function in MATRIXx to solve Ricatti Equations and obtain Optimal Gain Matrix, G

$$G = \begin{bmatrix} 0.5517 & 0.5090 & -0.1558 & 1.2036 & 1.1926 & 0.2094 \\ 0.8352 & -0.1994 & -0.7753 & 2.0760 & 0.8445 & 0.8281 \end{bmatrix}$$

Step 10. Assemble closed-loop plant (A_c) matrix in MATRIXx

Closed-loop Model

$$\dot{x} = A_c x + \hat{E} f$$

$$y = \hat{C} x$$

where

$$A_c = \hat{A} - \hat{B}G \qquad \text{(G=gain matrix)}$$

$$= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -0.5647 & -0.3022 & 0.4365 & -1.3497 & -0.1291 & 0.3111 \\ -0.2578 & -0.8214 & 0.0322 & -0.5451 & -0.7426 & -0.1748 \\ 0.6778 & 0.1372 & -4.4096 & 1.7413 & -0.9705 & -0.9342 \end{bmatrix}$$

Step 11. Run MATRIXx to calculate the closed-loop eigenvalues and damping ratios

Closed-loop Eigenvalues

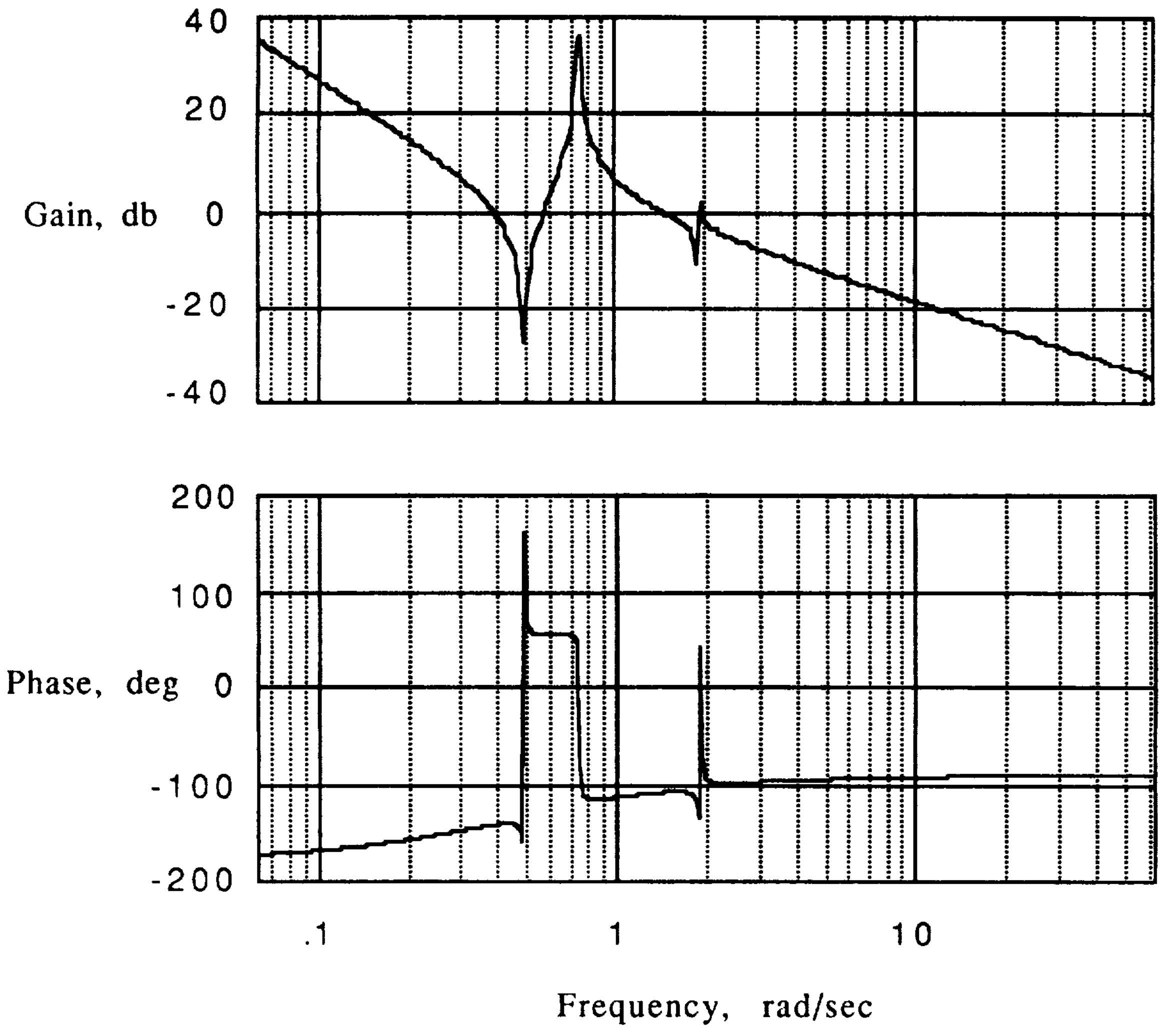
$$\lambda_c = \begin{bmatrix} -0.5091 + 1.8505i \\ -0.5091 - 1.8505i \\ -0.5971 + 0.4928i \\ -0.5971 - 0.4928i \\ -0.3980 + 0.7151i \\ -0.3980 - 0.7151i \end{bmatrix}$$

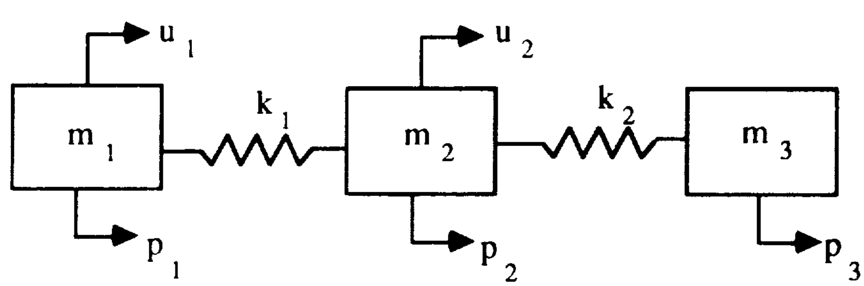
Since the real part is negative for all roots, the system is stable

Closed-loop Damping Ratios

$$[\zeta_c] = \begin{bmatrix} 0.7713 & 0 & 0 \\ 0 & 0.4863 & 0 \\ 0 & 0 & 0.2652 \end{bmatrix}$$

NOTE: Bode plots provide another useful means for determining the stability of a system. Bode plots may be obtained by using the MATRIXx BODE command. The following example shows a MATRIXx Bode plot of the loop transfer function for an input (actuator force) at mass m_1 and an output (position) at mass m_1.





where Gain = $20 \log_{10} [p_1/u_1]$

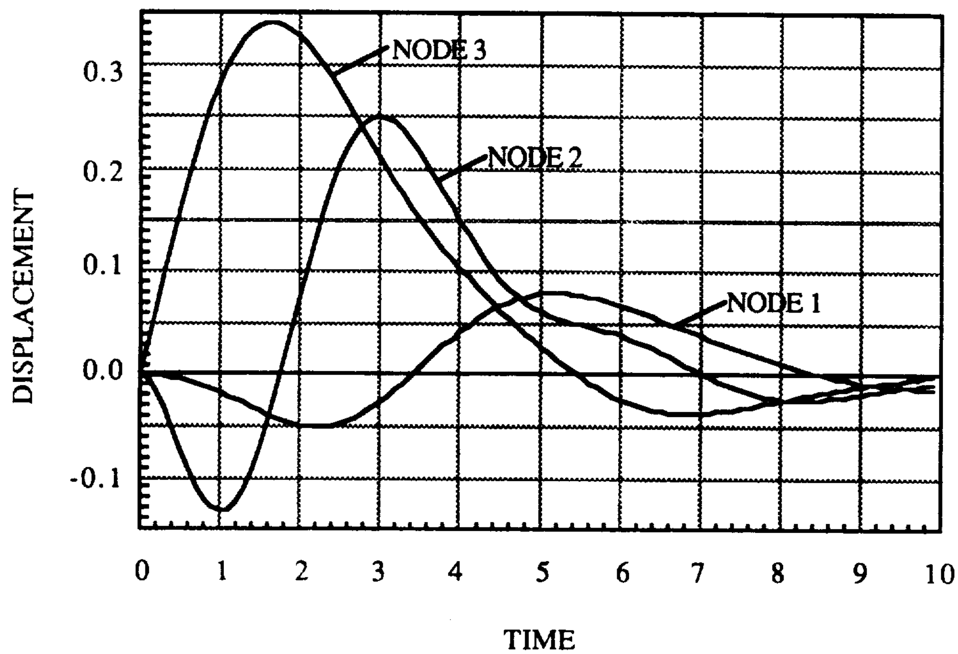
Example Bode Plot

Phase III. Simulation

Step 12. Read disturbance influence matrix, E, created by GLSM into MATRIXx

$$\hat{E} = \left[\begin{array}{c} 0 \\ \hline \Phi^T B_f \end{array} \right] = \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ \hline 0.4075 \\ -0.3518 \\ 0.2057 \end{array} \right]$$

Step 13. Calculate transient response to impulse load applied at m_3 and save modal solution in formatted MATRIXx files. Also save force/time histories of actuators and applied external loads. The following MATRIXx plot shows the physical displacements at the three nodes caused by an impulse applied at node 3. Additional physical quantities such as stresses could be obtained by using the Physical Data Recovery procedure.



Simulation Results for Impulse Load at Node 3

Step 14. Run RECORD CONTROL SIMULATION to store the information describing the simulation.

Phase IV. Data Recovery

Step 15. Run RECOVER PHYSICAL OUTPUT processor to create UHV (modal solution) file in MSC/NASTRAN INPUTT4 format and to create an MSC/NASTRAN Modified Solution 31 restart data deck that contains requests for physical time-history plots.

Contents of MATRIXx Solution Files

$$\begin{bmatrix} q_{11} & q_{12} & \cdots & q_{1n} \\ q_{21} & q_{22} & \cdots & q_{2n} \\ q_{31} & q_{32} & \cdots & q_{3n} \end{bmatrix} \qquad \begin{bmatrix} \dot{q}_{11} & \dot{q}_{12} & \cdots & \dot{q}_{1n} \\ \dot{q}_{21} & \dot{q}_{22} & \cdots & \dot{q}_{2n} \\ \dot{q}_{31} & \dot{q}_{32} & \cdots & \dot{q}_{3n} \end{bmatrix} \qquad \begin{bmatrix} \ddot{q}_{11} & \ddot{q}_{12} & \cdots & \ddot{q}_{1n} \\ \ddot{q}_{21} & \ddot{q}_{22} & \cdots & \ddot{q}_{2n} \\ \ddot{q}_{31} & \ddot{q}_{32} & \cdots & \ddot{q}_{3n} \end{bmatrix}$$

$$\begin{bmatrix} q_{11} & \dot{q}_{11} & \ddot{q}_{11} & q_{12} & \dot{q}_{12} & \ddot{q}_{12} & \cdots & q_{1n} & \dot{q}_{1n} & \ddot{q}_{1n} \\ q_{21} & \dot{q}_{21} & \ddot{q}_{21} & q_{22} & \dot{q}_{22} & \ddot{q}_{22} & \cdots & q_{2n} & \dot{q}_{2n} & \ddot{q}_{2n} \\ q_{31} & \dot{q}_{31} & \ddot{q}_{31} & q_{32} & \dot{q}_{32} & \ddot{q}_{32} & \cdots & q_{3n} & \dot{q}_{3n} & \ddot{q}_{3n} \end{bmatrix}$$

3 x 3n matrix with n time steps

Contents of MSC/NASTRAN UHV File

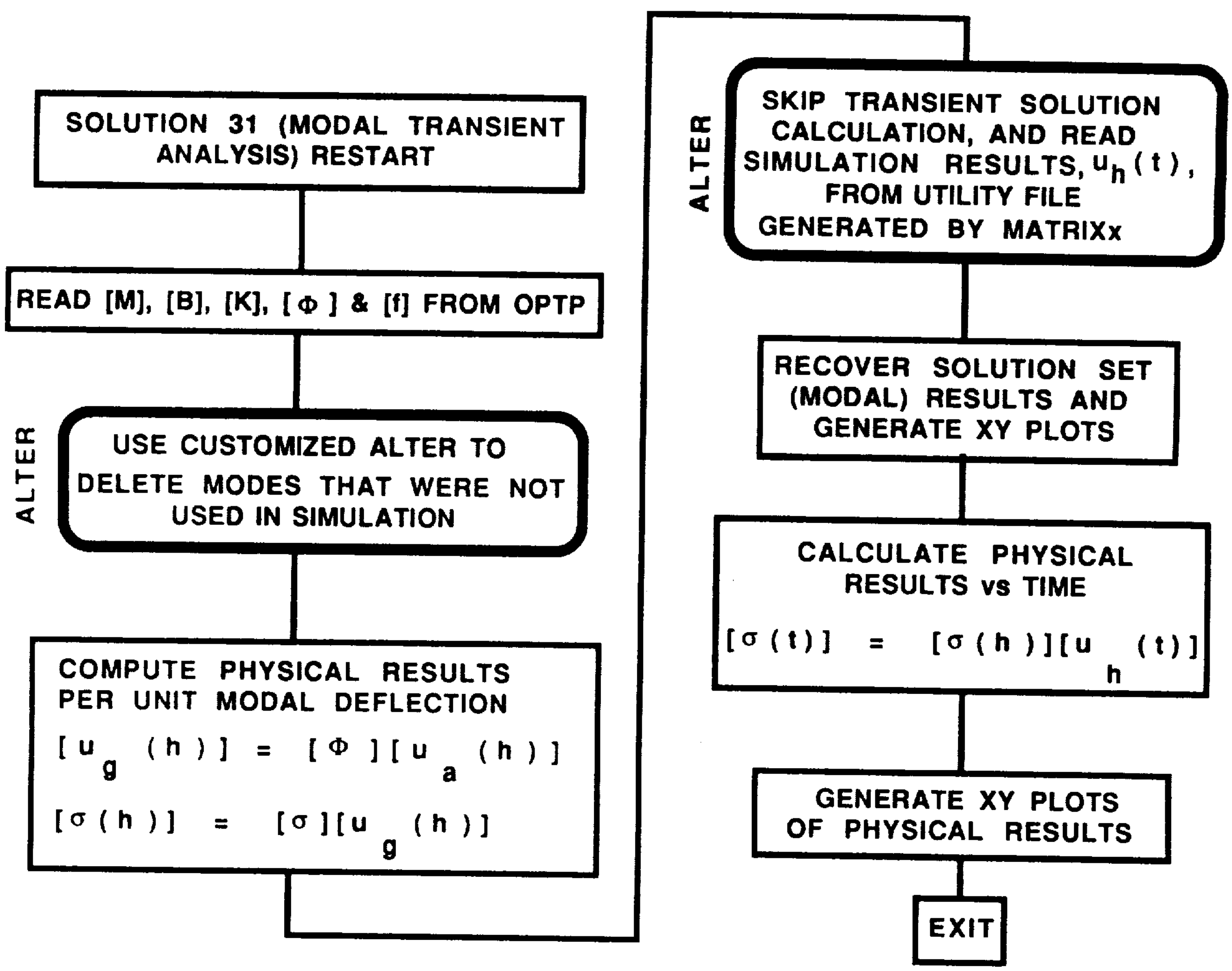
NOTES:

1. q_{ij} = modal coordinate of mode i at time step j
2. $x(t) = (q_1(t)\ q_2(t)\ \ldots\ q_n(t)\ \dot{q}_1(t)\ \dot{q}_2(t)\ \ldots\ \dot{q}_n(t))$
3. $\dot{x}(t) = (\dot{q}_1(t)\ \dot{q}_2(t)\ \ldots\ \dot{q}_n(t)\ \ddot{q}_1(t)\ \ddot{q}_2(t)\ \ldots\ \ddot{q}_n(t))$

Step 16. Run MSC/NASTRAN modified Solution 31 to obtain physical results. Alters shown below are automatically created by the RECOVER PHYSICAL OUTPUT processor.

ALTERATION OF SOLUTION SEQUENCE 31

TO RECOVER PHYSICAL RESULTS



APPENDIX C

RASTER METAFILE TRANSLATOR

Introduction - C- 1
Commands - C- 2

The growing problem of processing raster image data at LaRC is caused by the increasing number of software packages that generate raster data and the increasing number of graphics output devices that process this data. The result of this growth is the continual development of translators which convert a package-specific raster output file into a device-specific raster input file. The problem is further complicated by the requirement to transport these raster image data files (normally written in a binary format) from one host to another in order to access a specific device.

In order to provide some degree of compatibility for locally generated raster image data, LaRC has adopted a generic raster format known as Raster Metafile (RM) format. A prototype version of the Raster Metafile Translator (RMT) interpreter has been designed to permit the display and manipulation of RM formatted images.

The RM format and RMT do not eliminate the requirement for translators since package-specific raster data must be converted into RM format. However, the translation process terminates with RM format from the user's standpoint because the RMT should interface with all supported raster devices.

Since RM formatted images are generated from within IMAT utilities, no further discussion of the specifications of the RM format is included.

The remaining discussion focuses on the functionality of the prototype RMT interpreter. The command-driven RMT, which may be run in the foreground or background, provides for the reading, writing, display, and manipulation of RM format images. Image manipulation operations include format conversions, image composition, revising, clipping, and channel selection. A description of each command, including proper syntax is presented in the following pages.

CONVERT (CONV)

Purpose: To convert an RM formatted image from intensity (color table) to color table (intensity) format.

Parameters:

Required --- input metafile number
output metafile number

Optional --- picture number
window number

If the optional parameters are omitted, the current defaults (see SET/GET default command) are used. A range may be substituted for the picture number (see DIRECTORY command for range specification).

When converting from intensity to color table format, the user is prompted for desired color lookup table (LUT) size information. Defaults are obtained from the device specification (if a device is loaded) or from the number of data bits in the input image. When the black-and-white flag is on (see SET COLOR), the conversion process produces a single channel intensity image.

Usage:

CONVERT METAFILE # <PICTURE #> <WINDOW #> METAFILE #
CONV MF # <P #> <W #> MF #

NOTE: # refers to the index number of the metafile, picture, window, or viewport.

COPY

Purpose: To copy designated RM formatted images from an input metafile to the end of an output metafile.

Parameters:

Required	---	input metafile number output metafile number
Optional	---	picture number

If the picture number is not supplied, the default (see SET/GET default command) is used. A range may be substituted for the picture number (see DIRECTORY command for range specification).

Usage:

COPY METAFILE # <PICTURE #> METAFILE #
COPY MF # <P #> MF #

DIRECTORY (DIR)

Purpose: To display the contents of an RM formatted image file by printing header information for the selected image range.

Parameters:

Required	---	metafile number
Optional	---	image range

If the range is omitted, the headers for all images on the file are displayed. The range specification is available on several commands and takes the form from <first image> to <last image>where the image designators may be the keywords FIRST, CURRENT, or LAST, absolute (integer) image numbers, or relative (integer) offsets designated by + or -.

Usage:

DIRECTORY #
DIR #
DIR # FROM FIRST TO LAST
DIR # FROM 2 TO 4
DIR # FROM -1 TO +2

DRAW (D)

Purpose: To draw RM formatted image to selected graphics output devices.

Parameters:

Required --- none

Optional --- metafile number
picture number
window number
viewport number

If optional parameters are not selected the current default (see SET/GET default command) is used. A range may be substituted for the picture number (see DIRECTORY command for range specification.).

Usage:

DRAW <METAFILE #> <PICTURE #> <WINDOW #> <VIEWPORT #>
D <MF #> <P #> <W #> <V #>

GET

Purpose: To get current options and default values.

The GET subcommands include:

METAFILE (MF)
WINDOW (W)
VIEWPORT (V)
DEFAULT (DEF)
OPTION (OPT)
CHANNEL (CHA)
ERROR (ERR)

GET METAFILE (GET MF)

Purpose: Inquire about metafile name associated with specified metafile number.

Parameters:
Required --- metafile number

Optional --- none

Usage:

GET METAFILE #
GET MF #

GET WINDOW (GET W)

Purpose: Inquire about window coordinates for specified window number.

Parameters:
Required --- window number

Optional --- none

Usage:

GET WINDOW #
GET W #

GET VIEWPORT (GET V)

Purpose: To inquire about viewport coordinates for a specified viewport number.

Parameters:
Required --- viewport number
Optional --- none

Usage:

GET VIEWPORT #
GET V #

GET OPTION (GET OPT)

Purpose: To inquire about supported RMT options including: log file status, alternate command file name (if any), quantization scheme, dithering flag, clipping/resizing flag, and color vs black-and-white flag.

Parameters:
Required --- none

Optional --- none

Usage:

GET OPTION
GET OPT

GET DEFAULT (GET DEF)

Purpose: To inquire about RMT defaults including metafile, picture, window, and viewport numbers.

Parameters:
Required --- none

Optional --- none

Usage:

GET DEFAULT
GET DEF

GET CHANNEL (GET CHA)

Purpose: To inquire about the designated number of channels and channel ordering.

Parameters:
Required --- none

Optional --- none

Usage:

GET CHANNEL
GET CHA

GET ERROR (GET ERR)

Purpose: To inquire about the current error termination level.

Parameters:
Required --- none

Optional --- none

Usage:

GETERROR
GETERR

HELP (H)

Purpose: To access the on-line HELP documentation for individual RMT commands.

Parameters:
Required --- none

Optional --- specific command names

Usage:

HELP
HELP <command>
HELP <command> <SUBCOMMAND>

OVERLAY (OVER)

Purpose: To overlay one or more RM formatted images to produce a composite image.

Parameters:
Required --- input metafile number
output metafile number

Optional --- picture number
window number
viewport number

If optional parameters are omitted, the default (see SET/GET default command) values are assumed. No range specification is allowed with this command.

The resolution of the composite image is determined by the extent of supplied viewport dimensions.

Usage:

OVERLAY METAFILE # <PICTURE #> <WINDOW #> <VIEWPORT #> &
    METAFILE # <PICTURE #> <WINDOW #> <VIEWPORT #> & ...
    METAFILE #

OVERLAY MF # <P #> <W #> <V #> MF # <P #> <W #> <V #> ... MF #

QUIT (Q)

Purpose: To exit the RMT program.

Parameters:

Required --- none

Optional --- none

Usage:

QUIT
Q

SCALE (SCA)

Purpose: To scale the number of data bits in an input intensity image or size of LUT entries in a color table image.

Parameters:

Required	---	input metafile number output metafile number number or bits by which output image is scaled
Optional	---	picture number window number

If optional parameters are not supplied, the defaults (see SET/GET default command) are used. A range may be substituted for the picture number (see DIRECTORY command for range specification).

Usage:

SCALE METAFILE # <PICTURE #> <WINDOW #> METAFILE # NUMBIT #
SCA MF # <P #> <W#> MF # N #

SET

Purpose: To set options or override defaults.
The available SET subcommands are:

METAFILE (MF)
WINDOW (W)
VIEWPORT (V)
LOG (L)
QUANT (Q)
DEFAULT (DEF)
DITHER (DIT)
CLIP (C)
CHANNEL (CHA)
COLOR (COL)
ERROR (ERR)

SET METAFILE (SET MF)

Purpose: To identify an RM for reading or writing.

Parameters:

Required --- metafile number (may range from 1 to 5)

Optional --- write flag

When the write flag is omitted the metafile is opened for reading only.

Caution: when a metafile is opened for writing, no other read processing operations are allowed without first exiting the RMT (i.e., commands like DIR and DRAW are not allowed for output metafiles). Also, the output metafile required for several commands (such as CONVERT or OVERLAY) must be opened for writing.

Usage:

SET METAFILE # <WRITE>
SET MF # <W>

SET WINDOW (SET W)

Purpose: To set an RMT window (used to designate a subset of an RM formatted image).

Parameters:

Required --- window number (may range from 1 to 10)
window dimensions (in terms of minimum and maximum horizontal and vertical extents)

Optional --- none

Usage:

SET WINDOW # (X-MINIMUM X-MAXIMUM Y-MINIMUM
Y-MAXIMUM)
SET W # (X-MINIMUM X-MAXIMUM Y-MINIMUM Y-MAXIMUM)

SET VIEWPORT (SET V)

Purpose: To set an RMT viewport (used to position and size an RM image for display or output).

Parameters:

Required --- viewport number (may range from 1 to 10)
viewport dimensions (minimum and maximum horizontal and vertical extents)

Optional --- none

Usage:

SET VIEWPORT # (X-MINIMUM X-MAXIMUM Y-MINIMUM
Y-MAXIMUM)
SET V # (X-MINIMUM X-MAXIMUM Y-MINIMUM Y-MAXIMUM)

SET LOG (SET L)

Purpose: To toggle the flag to write the RMT log file, RMT.LOG. (This log file may be used as an alternate command file (see SOURCE command).)

Parameters:

Required --- none

Optional --- write flag (ON/OFF or +/-)
default is OFF.

Usage:

SET LOG ON
SET L +

SET QUANTIFY (SET Q)

Purpose: To set the uniform quantization flag used in converting from intensity to color table format.

Parameters:
Required --- none

Optional --- quantify flag (ON/OFF or +/-)

ON implies the use of uniform quantization and OFF implies the use of a color cube quantization scheme (the default is ON).

Usage:

SET QUANTIFY ON
SET Q +

SET DITHER (SET DIT)

Purpose: To set the dither flag used in converting from intensity to color table format. DITHER ON overrides the quantization scheme (see SET QUANTIFY command).

Parameters:
Required --- none

Optional --- dither flag (ON/OFF or +/-)
(the default is OFF.)

If dither is ON, it overrides the uniform quantization flag (see SET QUANTIFY).

Usage:

SET DITHER ON
SET DIT +

SET CLIP (SET C)

Purpose: To set the clip flag used to determine whether to clip image at viewport boundaries or resize image (using a bi-linear interpolation scheme) to fit the viewport.

Parameters:

Required --- none

Optional --- clip flag (ON/OFF or +/_)
(the default is ON implying clipping.)

Usage:

SET CLIP ON
SET C +

SET DEFAULT (SET DEF)

Purpose: To override an RMT default.

Parameters:

Required --- new default name
new default number

Optional --- picture number
window number
viewport number
(the default value for all parameters is 1)

Usage:

SET DEFAULT <METAFILE, PICTURE, WINDOW, VIEWPORT> #
SET DEF <METAFILE, PICTURE, WINDOW, VIEWPORT> #

SET CHANNEL (SET CHA)

Purpose: To override the default number of channels and/or the channel selection order for multi-channel image data.

Parameters:
Required --- none

Optional --- number of channels

By default the RMT assumes 3 channels ordered 1, 2, and 3 corresponding to red, green, and blue channel The user may alter this by supplying a new number of channels and/or channel ordering. (supplying 0 (zero) for the number of channels informs the RMT to use the number of channels in the input metafile image.)

Usage:

SET CHANNEL # (#1 #2 #3 ...)
SET CHA # (#1 #2 #3 ...)

SET COLOR (SET COL)

Purpose: To set color flag used in conversion process when converting from (to) color to (from) black-and-white.

Parameters:
Required --- none

Optional --- color flag (ON/OFF or +/-)
(the default is ON.)

The effect of the color flag being off is to produce a single channel intensity image (corresponding to a black-and-white image) during a conversion operation (see CONVERT command).

Usage:

SET COLOR ON
SET COL +

SET ERROR (SET ERR)

Purpose: To set error termination level.

Parameters:
Required --- none

Optional --- error termination level
S syntax
W warning
F fatal
N none
(default is F - fatal).

Usage:

SET ERROR <SYNTAX,WARNING,FATAL,NONE>
SET ERR <S,W,F,N>

SOURCE (SOU)

Purpose: To read command input from alternate command file

Parameters:
Required --- alternate command file name

Optional --- none

Nesting of alternate command files is not permitted.

Raster Metafile Translator 17

Usage:

SOURCE FILENAME
SOU FILENAME

Report Documentation Page

1. Report No. NASA TM-101500	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle Integrated Multidisciplinary Analysis Tool IMAT Users' Guide for the VAX/VMS Computers		5. Report Date December 1988
		6. Performing Organization Code
7. Author(s) Frances T. Meissner, Editor		8. Performing Organization Report No.
		10. Work Unit No. 585-01-31-06
9. Performing Organization Name and Address NASA Langley Research Center Hampton, VA 23665-5225		11. Contract or Grant No.
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, DC 20546-0001		13. Type of Report and Period Covered Technical Memorandum
		14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The Integrated Multidisciplinary Analysis Tool (IMAT) is a computer software system for the VAX/VMS computer that has been developed at the Langley Research Center. IMAT provides researchers and analysts with an efficient capability to analyze satellite control systems influenced by structural dynamics. Using a menu-driven executive system, IMAT leads the user through the program options. IMAT links a relational database manager to commercial and in-house structural and controls analysis codes. This paper describes the IMAT software system and how to use it.

17. Key Words (Suggested by Author(s)) Computer Software System Structural Analysis, Controls Analysis	18. Distribution Statement Unclassified-Unlimited Subject Category 39

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	232	A11

NASA FORM 1626 OCT 86