A NEW METHOD FOR GENERATING AND MAINTAINING RIGID FORMATS IN NASTRAN

By

P. R. Pamidi and W. Keith Brown
RPK CORPORATION

SUMMARY

Since all NASTRAN users are most likely to use some of the Rigid Formats it is important that a convenient means of updating these Rigid Formats be available. This benefits both the NASTRAN maintenance contractor and the NASTRAN user community. With this in view, RPK Corporation is currently developing a new method for generating and updating Rigid Formats in NASTRAN. The heart of this method is a Rigid Format data base that is in card-image format and that can therefore be easily maintained by the use of standard text editors. Each Rigid Format entry in this data base will contain the Direct Matrix Abstraction Program (DMAP) for that Rigid Format along with the related restart, subset and substructure control tables. NASTRAN will read this data base directly in every NASTRAN run and perform the necessary transformations to allow the DMAP to be processed and compiled by the NASTRAN executive. This approach will permit Rigid Formats to be changed without unnecessary compilations and relinking of NASTRAN. Furthermore, this approach will also make it very easy for users to make permanent changes to existing Rigid Formats as well as to generate their own Rigid Formats. This new method will be incorporated in a future release of the public version of NASTRAN.

INTRODUCTION

The popularity and longevity of NASTRAN are due to the many types of analyses that it supports and the generality and flexibility that it offers to the user to perform these analyses. The analyses are implemented in NASTRAN by the use of many functional modules each of which can be considered as an independent program. These functional modules interface with one another through executive parameter flags and a local data base called the File Allocation Table (FIAT). The FIAT and the flags are maintained by the NASTRAN executive. The order in which the functional modules are executed and the definition of the files and flags to be read or written by these functional modules are specified by means of a higher level language called the Direct Matrix Abstraction Program (DMAP). The DMAP is compiled and processed by the NASTRAN executive.

The most general way of using NASTRAN is with a user written DMAP program. However, in order to relieve the user of the burden of writing DMAP sequences for the most commonly used analyses (e.g., static analysis, normal mode analysis, etc.), a number of standard DMAP sequences (with sophisticated restart capabilities and subset features) have been developed and included in NASTRAN. These standard DMAP sequences are collectively called the NASTRAN Rigid Formats. There are currently twenty (20) Rigid Formats in NASTRAN. They are very valuable to the user as they are in a readily available and useable form and offer the user the flexibility for performing a variety of analyses. Modifications to the Rigid Formats can be easily made by the user by means of the ALTER feature described in Section 2.2 of the NASTRAN User's Manual (Reference 1).

PRESENT METHOD FOR GENERATING AND MAINTAINING RIGID FORMATS

At present, the Rigid Formats exist in NASTRAN in the form of subroutines. These subroutines are called the LDxx (LD01, LD02, etc.) subroutines and are described in Section 6.6 of the NASTRAN Programmer's Manual (Reference 2). There is one LDxx subroutine for each Rigid Format in NASTRAN. The bulk of the code in these subroutines is in the form of Fortran DATA statements consisting mainly of Hollerith values of the type 4Hxxxx. This format is very awkward. Consequently, making updates to these subroutines is an extremely tedious, time-consuming and error-prone procedure. In order to overcome this serious problem, a utility program called the Rigid Format Generation (RFGEN) program was developed for use by the maintenance contractor to automatically generate the LDxx subroutines. The RFGEN program requires the maintenance of a separate data base of Rigid Formats and their associated restart tables. This data base is read by the RFGEN utility to generate the desired LDxx subroutines.

The current practice of generating and maintaining Rigid Formats has many serious drawbacks. These are discussed in detail below.

1. Because of the nature of the LDxx subroutines, the use of the RFGEN utility is absolutely essential when major changes (e.g., extensive changes to the DMAP) need to be made to any Rigid Format. However, the RFGEN utility is currently operational only on the CDC machine. Hence, the maintenance contractor cannot generate the LDxx subroutines directly on another machine which he may be using as the primary machine for NASTRAN maintenance.

2. The Rigid Format information is currently contained both in the external data base used by the RFGEN utility as well as in the LDxx subroutines of NASTRAN. This is not only wasteful and cumbersome, but has also caused serious problems. Occasionally, updates were

made directly to an LDxx subroutine (for example, correction of a restart table) but never incorporated into the RFGEN data base. Consequently, these updates were lost when the RFGEN utility was used at a later date to generate replacement LDxx subroutines.

3. Permanent changes to a Rigid Format require the updating and recompiling of the appropriate LDxx subroutine as well as the relinking (or rebuilding) of Link 1. This is very inconvenient and annoying both to the maintenance contractor and users.

4. Users do not have access to the RFGEN utility which is available only to the maintenance contractor. They, therefore, cannot make major permanent changes to Rigid Formats at their local sites and are thus forced to use the ALTER capability in order to incorporate the desired changes. However, this has the disadvantage that the ALTER packet has to be included in every run in which the user needs the changes. This can be particularly inconvenient when the ALTERs are extensive.

   An extension of the above problem is that sophisticated users cannot generate new Rigid Formats of their own except by means of inserting the DMAP on a temporary basis through the NASTRAN DMAP approach. However, this has the same limitation that the ALTER cards have in that the DMAP has to be included in every run in which the user needs it. This can be particularly cumbersome when the DMAP contains many statements.

5. The substructuring capability of NASTRAN causes the automatic generation of DMAP ALTERs in those Rigid Formats that support this capability. Currently, Rigid Formats 1, 2, 3, 8 and 9 support this feature. These automatic ALTERs are currently specified in the Fortran subroutines ASCM01, ASCM05, ASCM07 and ASCM08. These ASCMxx subroutines must, therefore, be appropriately updated whenever changes to Rigid Formats 1, 2, 3, 8 and 9 affect their DMAP statement numbers and hence the DMAP ALTERs specified in these ASCMxx subroutines.

6. The data base currently used by the RFGEN utility is not in a very convenient format. Making changes to this data base is therefore not very easy, particularly when changes need to be made to the restart tables.

NEW METHOD FOR GENERATING AND MAINTAINING RIGID FORMATS

All of the above problems can be eliminated by having a Rigid Format data base that is read by NASTRAN directly and that can be maintained easily by means of standard text editors already available on the host computers. The important and distinct advantages to be derived from this approach are discussed below.

1. The need for the RFGEN utility and its associated data base will be eliminated and updating of Rigid Formats will become equally easy on all machines.

2. The LDxx subroutines in their current form will be eliminated from NASTRAN and the entire information about Rigid Formats will be contained in the new Rigid Format data base. This will greatly facilitate the maintenance effort.

3. Permanent changes to a Rigid Format can be incorporated simply by suitably updating the new data base by means of a text editor. The need for compilations and the relinking of Link 1 will thus be eliminated, thereby greatly aiding both users and the maintenance contractor.

4. It will become very easy for users to test and incorporate major changes to existing Rigid Formats at their local sites without having to use the ALTER feature in every run. Similarly, sophisticated users will be able to generate new Rigid Formats (complete with the associated restart tables) with relative ease.

5. By incorporating controls for substructure DMAP ALTERs in the new data base, the need for the possible updating of the ASCMxx subroutines every time the associated Rigid Formats are changed will be eliminated. This will greatly reduce chances for error since all information about the substructure DMAP ALTERs will be contained in the new data base instead of being spread over several ASCMxx subroutines.

6. The format of the new data base will be designed to be user-friendly and to be easily amendable.

IMPLEMENTATION OF THE NEW METHOD

The Rigid Format data base will be in a card-image format and will contain entries for all 20 Rigid Formats in NASTRAN. Each Rigid Format entry will contain the following items of information:

1. DMAP statements
2. Card Name Restart Table
3. File Name Restart Table
4. Rigid Format Change Restart Table
5. Subset flags
6. Controls for substructure DMAP ALTERs

The format of the data base will be designed to be user-friendly and to be easily amendable. All DMAP statements will be in a format similar to the format in which NASTRAN prints the DMAP. All restart tables, subset flags and substructure controls will be defined so as to permit easy editing.

The Rigid Format entries will be implemented as follows on the four computer systems on which NASTRAN is currently supported:

1. IBM - Each entry will be a member of a partitioned dataset
2. UNIVAC - Each entry will be an element of a file
3. CDC - Each entry will be a separate file
4. DEC VAX - Each entry will be a separate file

An I/O interface will be designed on all of the four computer systems to read the Rigid Format entries. This interface will be written in machine-dependent code. Additional subroutines in machine-independent code will be designed to perform the necessary transformations to allow for the processing and compilation of the DMAP by the NASTRAN executive. The current LDxx subroutines will thus no longer be used and will therefore be deleted from the code.

In addition, documentation updates will be provided for the NASTRAN User's Manual and the NASTRAN Programmer's Manual. The User's Manual updates will contain a definition of the format of the new Rigid Format data base and also a description of how to maintain and update the data base. The Programmer's Manual updates will include a description of all subroutines and COMMONs that are added to NASTRAN as part of this new capability.

CONCLUDING REMARKS

The new method of generating and maintaining Rigid Formats described in this paper should be welcomed by both the NASTRAN maintenance contractor and the NASTRAN user community. It will, for the first time, make it possible for the maintenance contractor to update and modify existing Rigid Formats without the use of a utility program and without unnecessary compilations and relinking of NASTRAN. It will also, for the first time, make it possible and practicable for users not only to make permanent changes to existing Rigid Formats, but also to create new Rigid Formats of their own. This will greatly enhance the flexibility and attractiveness of NASTRAN.

REFERENCES

1. The NASTRAN User's Manual, NASA SP-222(05), December 1978.
2. The NASTRAN Programmer's Manual, NASA SP-223(05), December 1978.