NUCLEAR ENGINEERING AND DESIGN 14 (1970) 99–103. NORTH-HOLLAND PUBLISHING COMPANY

AN INTEGRATED PROGRAM LIBRARY FOR MATERIAL PROPERTY DATA

A. PEE and U. SCHUMANN

Institut für Reaktorentwicklung, Gesellschaft für Kernforschung m.b.H., Karlsruhe, Germany

Received 30 April 1970

A program system for material property data (MAPLIB) is introduced which is available for incorporating large quantities of material data in a form suitable for computer programs. MAPLIB consists of a multitude of Fortran Function subprograms - one for each material property - which are in accordance with some conventions relative to form, nomenclature and units. The data can be retrieved by problem programs in a flexible mode. As far as possible errors in retrieval requests are supervised. For the integration of new data functions in MAPLIB a utility is available.

1. Introduction

The design and calculation of technical systems, e.g., reactors, requires the knowledge of material data. Usually, material data are obtained from handbooks.

For complicated systems, the calculations are carried out with computer programs. In this case, not man but the computer is looking for material data. The scanning rate of a computer is many times that of a human being. In addition, material data, as a rule, depend on a number of variables, such as temperature and pressure. Since these variables are varied in the design, it is not sufficient to have just one material data value for each item, but rather the dependence of the material data from the appropriate parameters must be available to the problem oriented programs.

At the Karlsruhe Institut für Reaktorentwicklung (Institute of Reactor Development), therefore, the MAPLIB program system (*Ma*terial *P*roperties Program *Lib*rary) was set up to make available material property data in a form suitable for computer programs [1].

The usual way of listing material data in tables is not well suited to this case because interpolations become necessary, while the information contained in the tables can be represented in most cases in computing instructions, such as polynomials, at less expense in terms of storage and evaluation time.

MAPLIB, therefore, consists of a multitude of Function subprograms - one for each material property - which can be called in a flexible mode. MAPLIB is written in FØRTRAN IV because this is the most common programming language for technical applications. The system is implemented on the IBM 360/65 in Karlsruhe.

Within the work carried out at Karlsruhe the main effort was concentrated on setting up a library organization and a utility program for the simple incorporation of new functions in the library and for user information about the existence, accuracy and sources of particular or all material property functions contained in the system. For the time being, it has been left to the user's initiative to himself incorporate those material properties which are of interest to him and not yet included in the system so that, later on, they are available also to other users.

The arrangement of the system will be described below.

2. Basic system conventions

2.1. *Nomenclature*

The individual material data functions are characterized by the material described and by the property. Materials and properties are characterized by symbols, i.e.,

materials up to four characters, e.g., YYYY

properties with two characters, where the first character must be a letter, e.g., XX.

Table 1
Symbols of materials implemented up to now

B4C	boron-4-carbide
HEV	helium gas
NA	sodium, general
NAL	sodium liquid
NAV	sodium vapor
NALS	saturated liquid sodium
NAVS	saturated sodium vapor
PUO	plutonium oxide
UO	uranium oxide
UPUO	uranium-plutonium mixed oxide
4961	4961 grade steel
4981	4981 grade steel
4988	4988 grade steel
Other symbols already defined:	
H2O	water, general
H2OL	water, liquid
H2OV	steam
AIRV	air
CO2V	CO_2 gas
U238	uranium-238 isotope
U235	uranium-235 isotope
PU39	plutonium-239 isotope

Table 2
Symbols of properties implemented up to now

VP	saturated vapor pressure
VT	saturated vapor temperature
FT	melting temperature
FH	heat of fusion
RH	heat of recrystallization
VH	evaporation enthalpy
VS	evaporation entropy
EH	enthalpy
ES	entropy
RO	density
VO	specific volume
CP	specific heat at constant pressure
CV	specific heat at constant volume
PR	Prandtl number
WL	thermal conductivity
ZD	viscosity, dynamic
ZK	viscosity, kinematic
SB	rupture strength
SD	elongation strength
SF	yield strength
EM	modulus of elasticity

The symbols are formed according to certain ground rules which mnemonically facilitate the assignment of the symbols [1]. Tables 1 and 2 show lists of the symbols thus defined. As a rule, the material data function is assigned as a name the combination of the property and material symbols, e.g., XXYYYY.

2.2. *Units*

The units of all quantities in MAPLIB are defined in the M.K.S.A. (meter, kilogram, second, ampere) system, i.e., the units of the arguments and the values of the functions must be indicated in that dimensional system. The temperature unit is °K.

2.3. *Example*

For example, in the library there is one function called RØNAV with the arguments T (temperature) and P (pressure). RØ is the symbol for density, NAV is the symbol for sodium vapor. T must be supplied in °K, P in N/m^2. RØNAV is calculated in the units of $\mathrm{kg/m}^3$.

3. Retrieval of material data

It is the experience obtained with design and analysis codes in the nuclear and mechanical engineering field that in many program applications the material and the property item to be retrieved upon program execution is known at the time when the program is made. In many other cases, however, either the material or the property or both of them are not fixed until the program is actually executed. This fact has been accounted for in the design of MAPLIB, and the programmer now has the choice to assign material and property of his retrieval request at programming time or to leave this the future user of his program. Therefore, access to a material data item S of the material YYYY and the property XX for the parameters $A1$, $A2..., An$ is possible at three levels of the library, according to fig. 1.

At the lowest level the function is called directly:

$$S = \mathrm{XXYYYY}\,(A1, A2,..., An)\,.$$

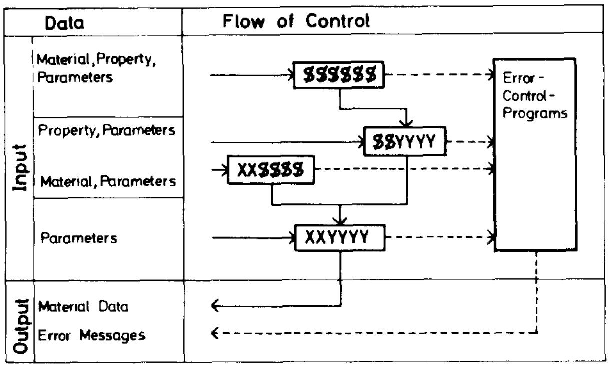

Fig. 1. Retrieval of material data by MAPLIB.

At the intermediate level either the material or the property is a dummy variable.

Material level:

$$S = \$\$\mathrm{YYYY}\,(\mathrm{'XX'}, A1, A2, ..., A\mathrm{n})$$

\$\$ in this case stands for the dummy variable XX which must be indicated as a literal data item.

Property level:

$$S = \mathrm{XX}\$\$\$\$\,(\mathrm{'YYYY'}, A1, A2, ..., A\mathrm{n})$$

\$\$\$\$ stands for the dummy variable YYYY.

Top level:

$$S = \$\$\$\$\$\$\,(\mathrm{'XX'}, \mathrm{'YYYY'}, A1, A2, ..., A\mathrm{n})\,.$$

Here, the names of both the material and the property can be indicated as arguments.

These different possibilities of access allow the programmer to set up his program in a most efficient way for all possible materials and/or properties. He can provide for the names of materials and properties as the input data. For instance, the program needing density as a material data item can be programmed universally for all materials with the function RØ\$\$\$\$ (MATER, *T*, *P*), where MATER is assigned the name of the desired material upon program execution.

The functions of the higher levels have catalogues of all the functions integrated into MAPLIB which are called according to the material and property symbols listed.

4. Error handling

4.1. *Ranges of error*

Whenever a problem program is executed which requests a material property value from MAPLIB, this retrieval request may be in error. For all error classes which can be identified by MAPLIB at execution time, standard reactions are established with respect to:

the answer given to the request in error,
the indication of the error given to the user,
the continuation of program execution.

Five ranges of error as given in table 3 have been defined in the library.

Table 3

Type of error	Meaning and answer to the request
1	The number of parameters supplied is larger than anticipated by the library function. The parameters supplied in excess are not taken into account.
2	Parameter outside the accuracy range of the procedure, while the function is still mathematically defined. MAPLIB furnishes a value which is not accurate and does not even have to be meaningful physically.
3	For the parameters supplied the function is not defined mathematically or physically. MAPLIB returns the value of 1.
4	The function anticipates more parameters then were supplied. The system returns the value of 1.
5	For this material and this property no function has been included in the catalogue. MAPLIB returns the value of 1.

4.2. *Error messages and error sequences*

A distinction must be made between default reactions and optional reactions.

4.2.1. Default reactions

In a case of error, the system normally prints out an error message in English plain text. The first error results in termination of the program execution.

4.2.2. Optional reactions

By calling special system subroutines the user has the possibility of varying some or all the error control parameters. These possibilities include:

modification of the number of permissible errors before termination of the execution (default: 0) for each type of error;

suppression of the error message either wholly or for certain types of error or output to a unit other than the standard system output;

modification of the number of errors which are permissible without error message (default: 0) or of the maximum number of error messages (default: 10) for each range of error.

Moreover, the user has a number of routines available which will enable him to inquire whether there has been an error at all and, if so, how many errors there have been and in what types of error they have occurred so that the user himself can decide how his program should behave in a case of error.

4.3. *Scanning of the number of parameters*

While it was possible to code the identification of error types 2, 3 and 5 in FØRTRAN IV, a small assembler routine had to be set up for assessment of error type 1 and 4, i.e., of the number of parameters supplied, which determines the number of parameters in the internal registers of the computer. However, this assembler routine does not restrict the compatibility of MAPLIB with other computers which use only FØRTRAN IV. If this routine is abandoned, only the control of the number of parameters is no longer possible; hence, the user himself would have to make sure to supply the correct number of parameters.

On the other hand, this routine has the advantage that in the absence of certain parameters the function automatically and independently inserts standard values. This possibility can be employed especially if more recent results of research increase the number of parameters influencing the material property function and if this is to be taken into account in the library by an improved program without having to use a new name for this function and change all previous user programs.

5. **Updating**

5.1. *Incorporation of available material data functions*

If a material data function of the lowest level is to be incorporated in the library, it must be incorporated in the catalogues of the program of higher levels with its property and material names. In addition, the function must be supplemented by the error handling routines. For this implementation, a utility program is available. Input for that program are the FØRTRAN cards of the new function. The completed FØRTRAN higher level programs and the revised function are the output. Both can be routed immediately to the system libraries.

5.2. *Generating material data function of the lowest level*

The basis of the material data functions to be generated are data from the literature or from measurements arranged as functions or in tables, which are often in units different from M.K.S.A. Auxiliary routines are available which generate interpolation or approximation functions to the tabulated data and for the incorporation of the error control ready for use in the library. A program is planned for the automatic conversion of the data, which may be available in different format, into material data functions which, in turn, would constitute the input to the utility program mentioned above.

6. **Setting up individual sublibraries**

In the form described above MAPLIB would have two disadvantages:

(1) Unless comprehensive overlay or other instructions are given to the linkage editor, a call of the function $$$$$$ has the effect of linking the entire library into the program to be executed, i.e., also those functions which are of no interest to the user in this particular application.

(2) The names of the material data functions may collide with the names of functions already provided by the user.

The MAPLIB utility program now offers the possibility of selecting an individual sublibrary out of the available complete catalogue for each user in which, in addition, every other letter in the regions of A–H, Ø–Z can be substituted in the nomenclature instead of "$".

It is further planned to provide the possibility to vary also the names of the material data functions of the lowest level according to the individual users' requests.

7. Conclusion

Presently, 110 material data functions are integrated in MAPLIB. These data refer mainly to materials which are of interest in reactor technology. This scope certainly is still very narrow. However, in MAPLIB a system has been created which is available for incorporating large quantities of material data at small expense in storage requirements and for very short response times to random requests.

References

[1] KFK report, under preparation.