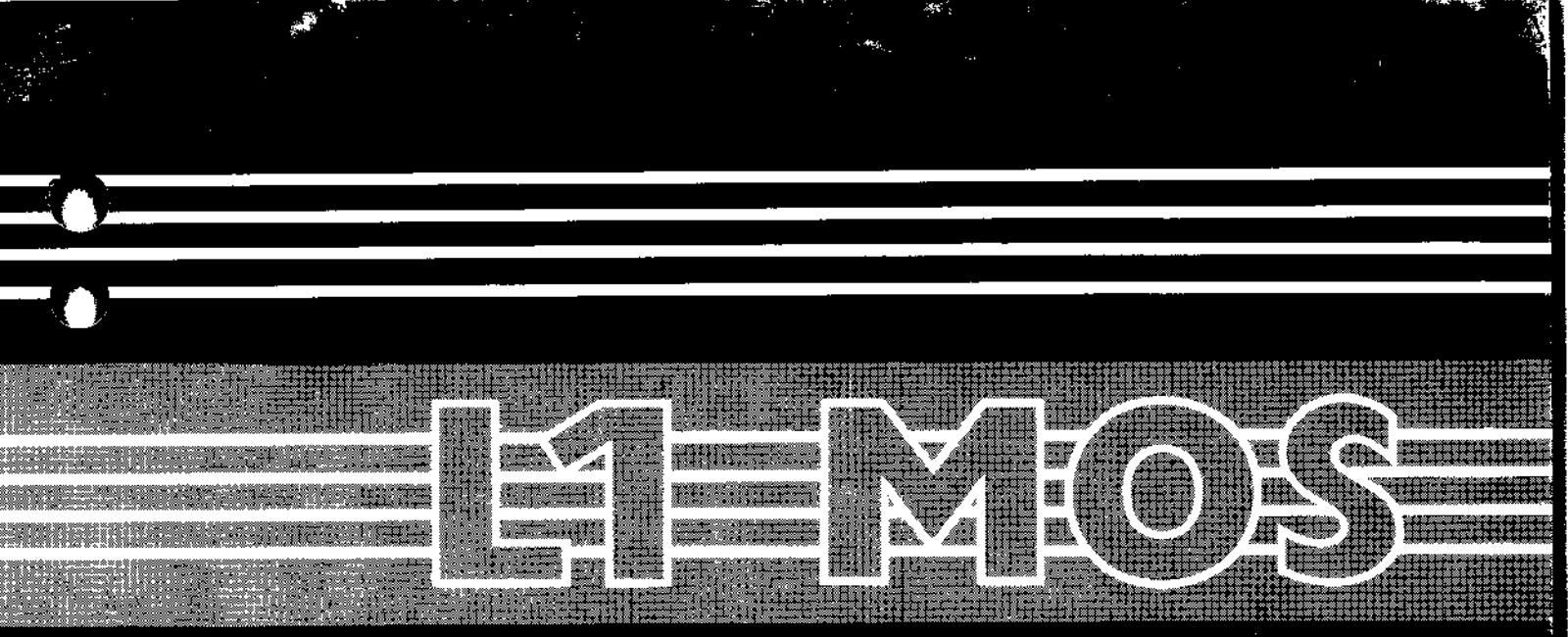
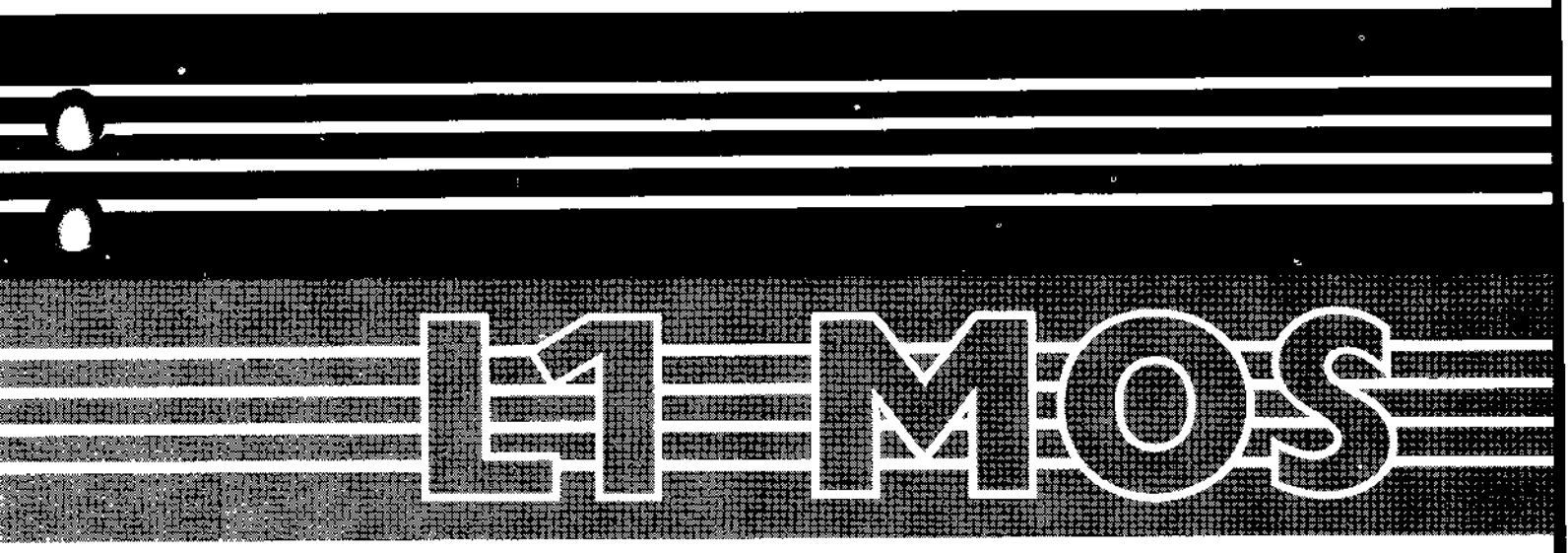

L1MOS



DMS - DATA MANAGEMENT SYSTEM
Features and Functions

olivetti

PREFACE

This manual describes the DMS features and function in general.

This information gives a basic initial approach to the DMS environment which is also necessary to gain a deeper understanding of the package.

The manual assumes that the reader has no previous knowledge whatsoever of the subject.

SUMMARY

The manual describes the following subjects:

- definition of DMS and general features
- functioning mechanisms
- the functions.

The last section includes a list of the main DMS commands whith a brief description of the corresponding function, giving the reader a more concrete understanding of the capabilities of DMS.

REFERENCES

Read first...

Introduction to MOS - Code 4002130 G

For further information, read...

DMS - User Guide - Code 4002360 F (vol. 5)

DMS - Data Definition Language - User Guide - Code 4004570 D (vol. 5)

First Edition: January, 1983

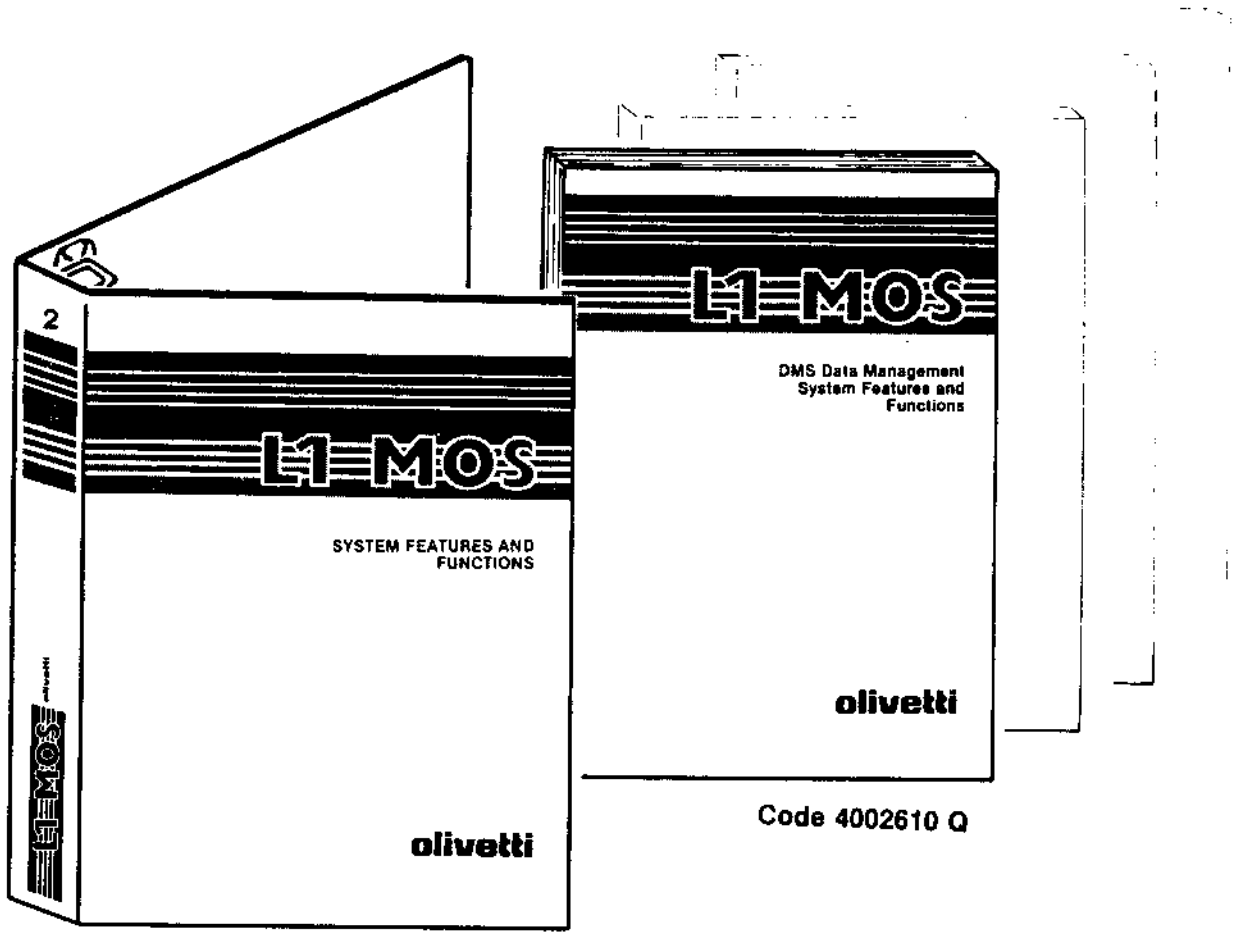
Second Edition: June, 1983

Third Edition: August, 1984

Release: 3.1, 4.0

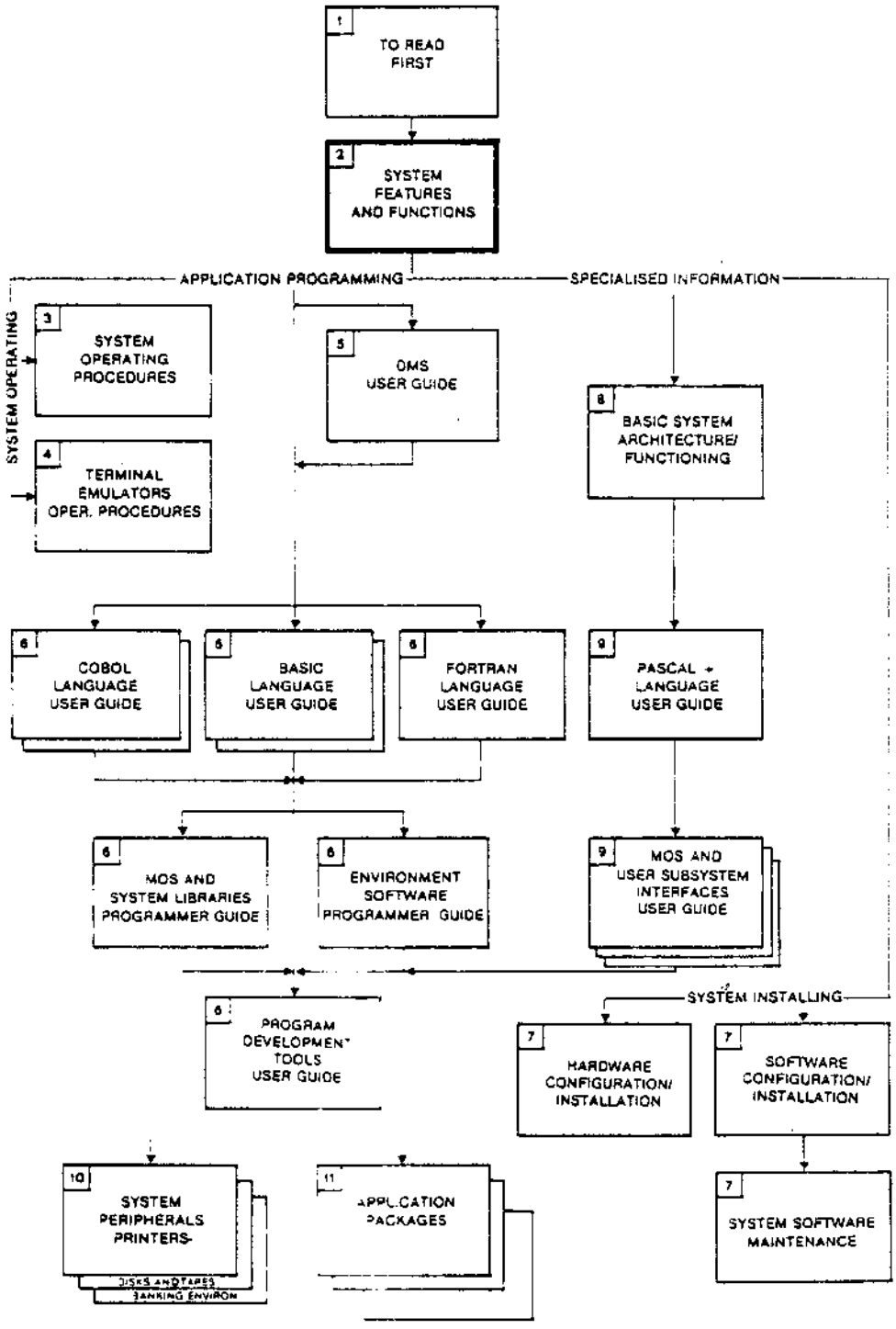
PUBLICATION ISSUED BY:

Ing. C. Olivetti & C., S.p.A.
Direzione Documentazione
77, Via Jervis - 10015 IVREA (Italy)



Code 4002610 Q

Code 4004750 F


1
TO READ FIRST
2
SYSTEM FEATURES AND FUNCTIONS
APPLICATION PROGRAMMING
SPECIALISED INFORMATION
SYSTEM OPERATING
3
SYSTEM OPERATING PROCEDURES
4
TERMINAL EMULATORS OPER. PROCEDURES
5
OMS USER GUIDE
8
BASIC SYSTEM ARCHITECTURE/ FUNCTIONING
6
COBOL LANGUAGE USER GUIDE
6
BASIC LANGUAGE USER GUIDE
6
FORTRAN LANGUAGE USER GUIDE
9
PASCAL + LANGUAGE USER GUIDE
6
MOS AND SYSTEM LIBRARIES PROGRAMMER GUIDE
6
ENVIRONMENT SOFTWARE PROGRAMMER GUIDE
9
MOS AND USER SUBSYSTEM INTERFACES USER GUIDE
6
PROGRAM DEVELOPMENT TOOLS USER GUIDE
SYSTEM INSTALLING
7
HARDWARE CONFIGURATION/ INSTALLATION
7
SOFTWARE CONFIGURATION/ INSTALLATION
10
SYSTEM PERIPHERALS PRINTERS-
11
APPLICATION PACKAGES
7
SYSTEM SOFTWARE MAINTENANCE

1. INTRODUCTION 1.0.1

2. GENERAL FEATURES 2.0.1

PRODUCT FEATURES 2.1.1

USER FEATURES 2.2.1

3. DMS MECHANISM 3.0.1

4. DMS LOGIC 4.0.1

5. THE DMS FUNCTIONS 5.0.1

THE DATA STRUCTURES DEFINITION FUNCTION 5.1.1

THE PROCEDURE DEFINITION FUNCTION 5.2.1

THE APPLICATION FUNCTION 5.3.1
Reference to the Existing Collections 5.3.1
Access to the Data File Records 5.3.1
Generation of Reports and Graphs 5.3.4
Definition of Variables and Transfer of Values . . . 5.3.12
Control of the Command Sequence 5.3.13
Execution of a Command File 5.3.13
Execution of a Procedure 5.3.14

THE DATA PROTECTION FUNCTION 5.4.1

THE EDITING FUNCTION 5.5.1

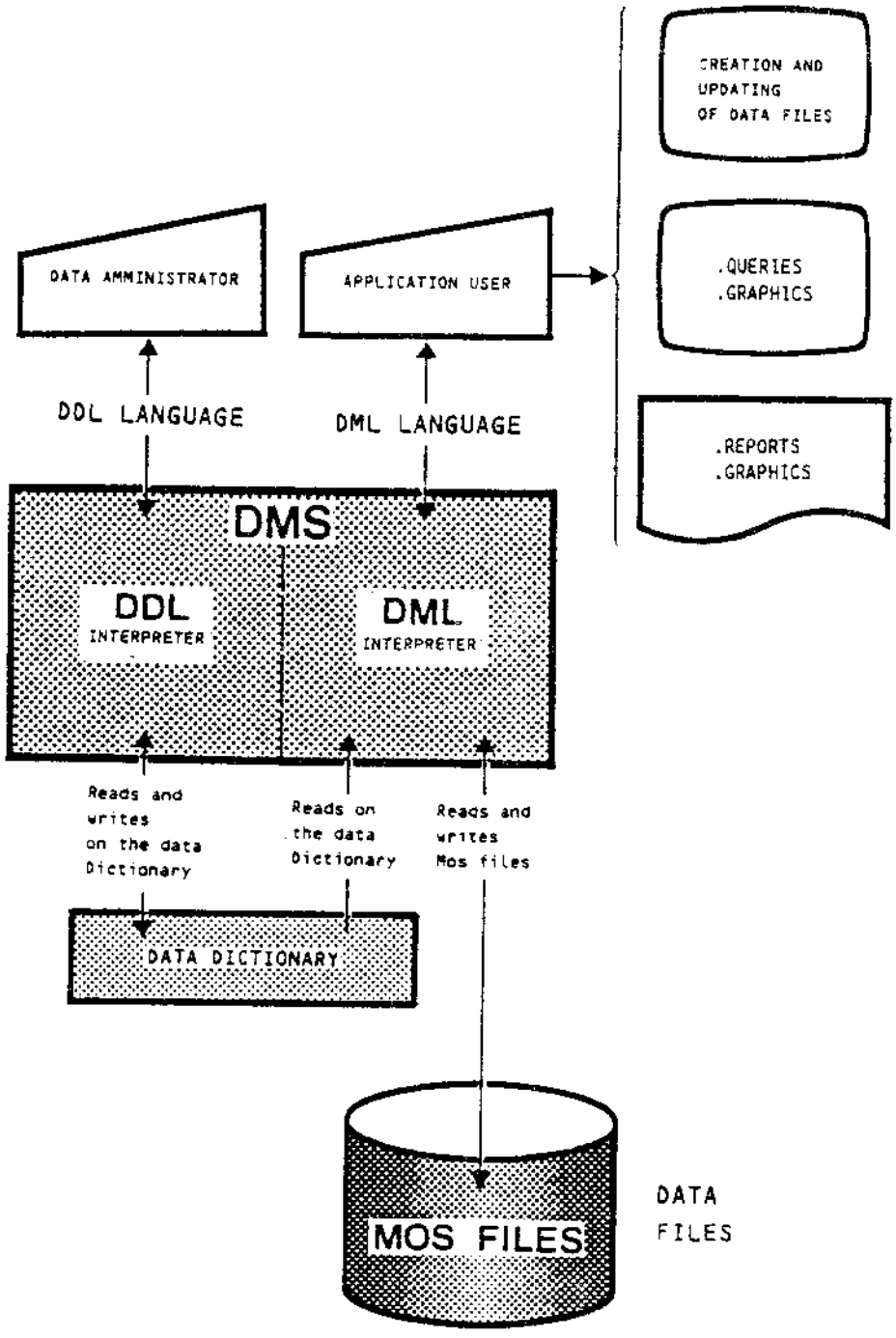


Fig. 1. 1 - DMS Scheme

1. INTRODUCTION

The DMS package provides the Data Manipulation Language (DML) which is used to access files to carry out the following functions in the MOS environment:

- Data Entry (create, modify, delete records).
- Inquiries.
- Reports.
- Graph generation.

In order to access physical files and carry out the above mentioned functions the DML interpreter needs specific information which describes the files to the DMS.

The data administrator, responsible for the DMS installation, creates and stores this information in particular libraries of the data Dictionary using the Data Definition Language (DDL).
The DMS is therefore composed of:

- DDL language interpreter.
- DML language interpreter.
- Data Dictionary.

2. GENERAL FEATURES

Product Features

The DMS:

- Can provide the user with a personalised vision of the physical data by using particular descriptors which are stored in the Data Dictionary.
  Moreover, the display and the input output handling of the data can be performed by using the VISA and TFORM packages.
- Minimises both data and software redundancy by defining various virtual files on the same physical data structure.
- Disengages the user from the traditional programming environment facilitating the development of different management applications.
- Is strongly oriented towards interactive use.
- Protects the Data Dictionary and data files by controlling the right to access them.

User Features

The DML (Data Manipulation Language):

- Allows the writing of commands without specific formal restrictions.
- Is flexible.
- Allows the storage of a sequence of commands to be executed at a later stage.
- Presents a simple operating interface.
- Allows creating and updating of files, reports and graphs generation.

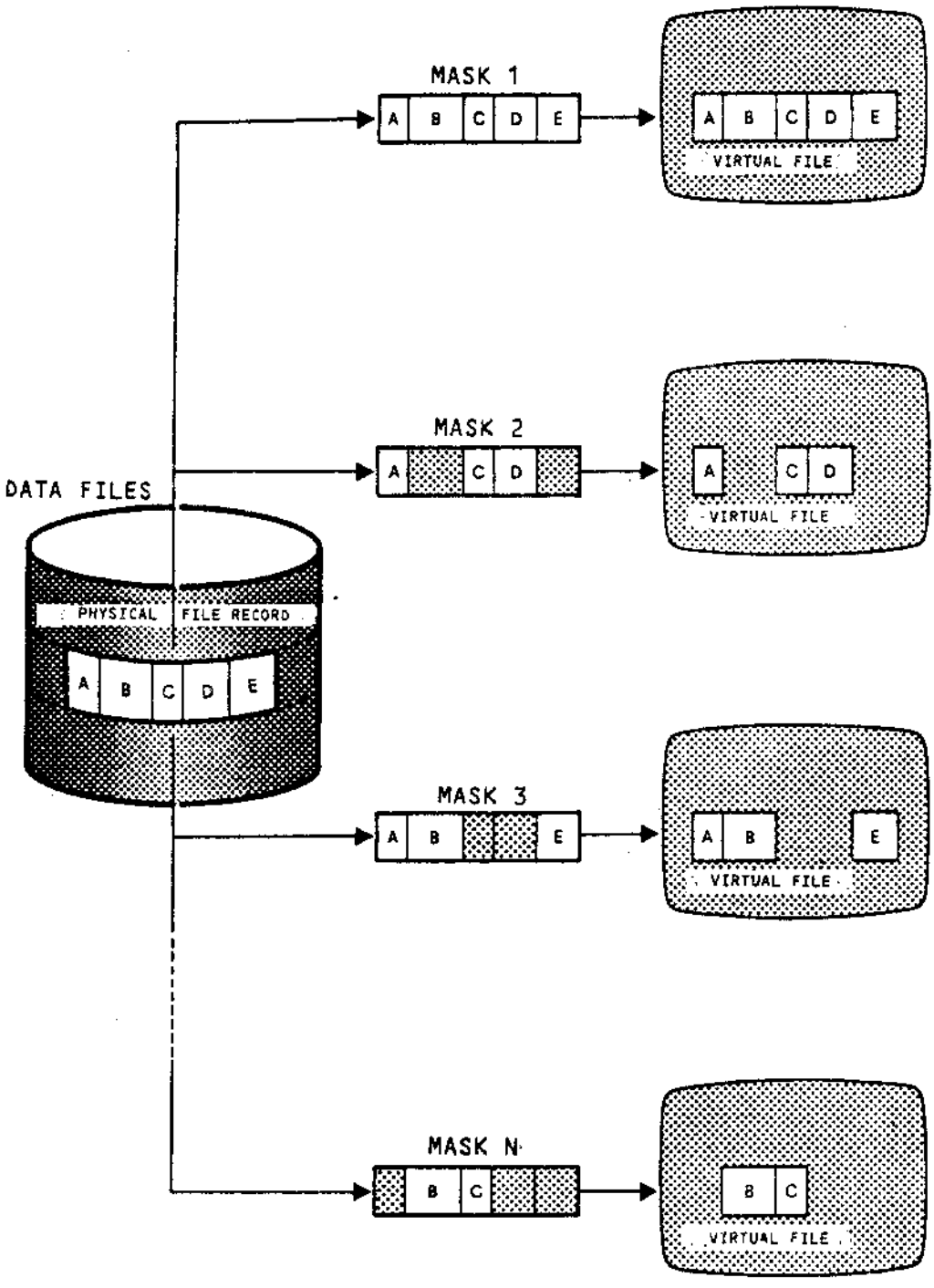


Fig. 2. 1 - Virtual File Representation

Each file used in the DMS environment has its record structure described to the DMS.

It should be noted that the physical structure of a record may be described in different ways, depending on the needs of the various categories of users. In other words, the user "views" the file he wishes to access through his own mask which is able to present the user with only the data he requires, in the most appropriate format.
This is the concept of the virtual file.
With reference to the handling of the values contained in the file records, the DMS may use the VISA and TFORM packages. Thus the user, by creating forms, can define the characteristics of each record file and the may of displaying the record.

The possibility of defining several virtual files from the same physical data structure is important because it:

- Avoids data redundancy.
  Since the same physical data may be described in different ways allowing many representations, the data does not have to be duplicated and reorganised according to the different application needs.

- Minimises software redundancy.
  Besides the functions normally handled by the Operating System (input-output, physical memory allocation) the DMS handles collection, verification, storage and updating of data as system services rather than program functions.

The development of management applications in the traditional programmer language assumes various phases (codify, compile, link-edit, program set up) which are a constant inconvenience, even in the case of very simple programs.
The DMS allows the execution of complex operations through the use of powerful commands which may be executed in an interactive manner.

Thus, it is possible to satisfy "ad hoc" requests which are a consequenc of unforseen decisions through the independence of application programs from the physical data structure, the possible use of common software resources, and, appropriate techniques for the management of data files. The DMS data protection allows for:

- The protection of the contents of the Data Dictionary.
  Only the data administrator may connect to the module that interprets the DDL commands, through which the descriptors used by the DMS and the DMS procedures are created and managed.

- The protection of specific record fields against certain users. This is a consequence of the possibility to define as many virtual files as needed.

- Two different methods for accessing data:

  . read only

  . read with the possibility to modify.

The salient DMS language features from the user point of view are:

- The command names imply the actual operation which they execute; eg. OPEN, CLOSE, MODIFY, CREATE...

- The commands are introduced via the keyboard and displayed on screen. There are no formal restrictions in writing single commands or a sequence of commands:
  the words are separated from each other by at least a blank.
  A sequence of commands may be written on the screen lines.

- It is a flexible language.
  When using certain commands, one may use them in their simple form or may specify any of the available options.

- The execution of DMS commands is interactive, and errors are returned as soon as they occur.
  It is however possible to store a sequence of DMS commands in order to execute them later.
  Each of these programs can be either:

  . a command file, stored on disk, or

  . a procedure, stored in the DMS procedure library.

  A procedure differs from a command file in that it allows the definition of symbolic parameters to which the user assigns an actual value when the procedure is executed.
  The procedure library is allocated on the Data Dictionary.
  The DMS, when executing either a command file or a procedure, reads the commands from the disk and interprets them in the same manner as those entered on the keyboard by the user.

The DMS makes use of the screen to communicate with the user.
The information available is displayed on screen; the user may then operate on it displacing the cursor as required.

- It is oriented towards data file management and the generation of reports and graphs presenting an alternative to traditional programming.

  DMS commands exist to:

  . open and close DMS files

  . create, modify and cancel records

  . display record fields in a selective manner

  . generate output on printer

  . draw graphs on the screen or with the printer

  . make choices at a logical level

  . define a sequence of commands as a sequence to be repeated cyclically.

3. DMS MECHANISM

THE DDL INTERPRETER

The DDL interpreter accepts and executes the Data Definition Language (DDL) commands by which the data administrator creates and handles:

- the descriptors used by the DMS : USER IDENTIFIERS, FILE DESCRIPTORS, VIEWS, COLLECTIONS
- the user defined DMS procedures, if any.

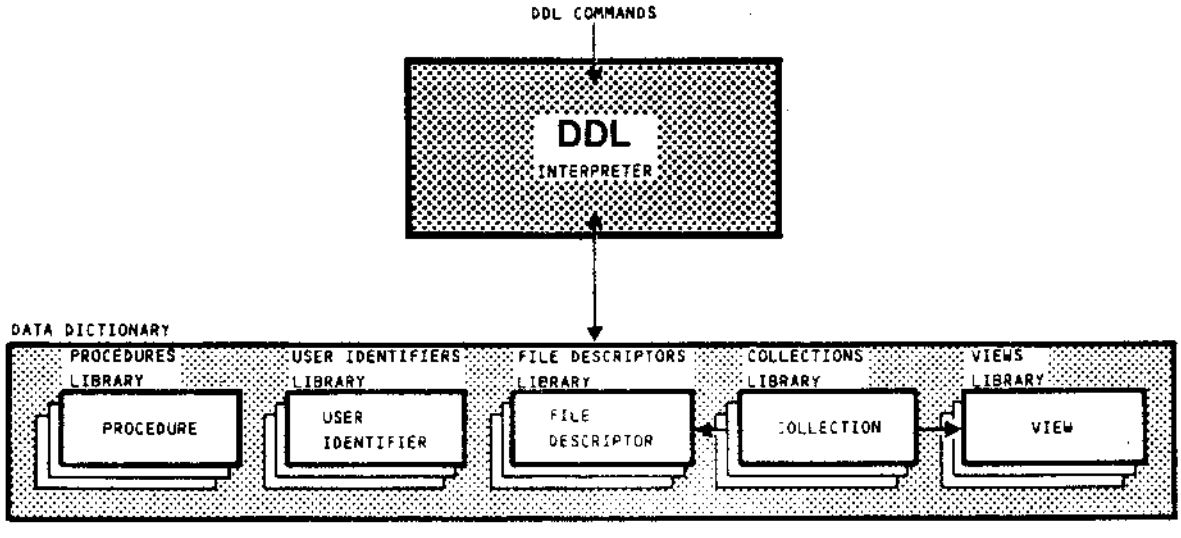


Fig. 3. 1 - DDL Commands Read and Write on the Data Dictionary

The logical link between the DMS descriptors is expressed by the type of information that they contain, as indicated in the following diagrams.

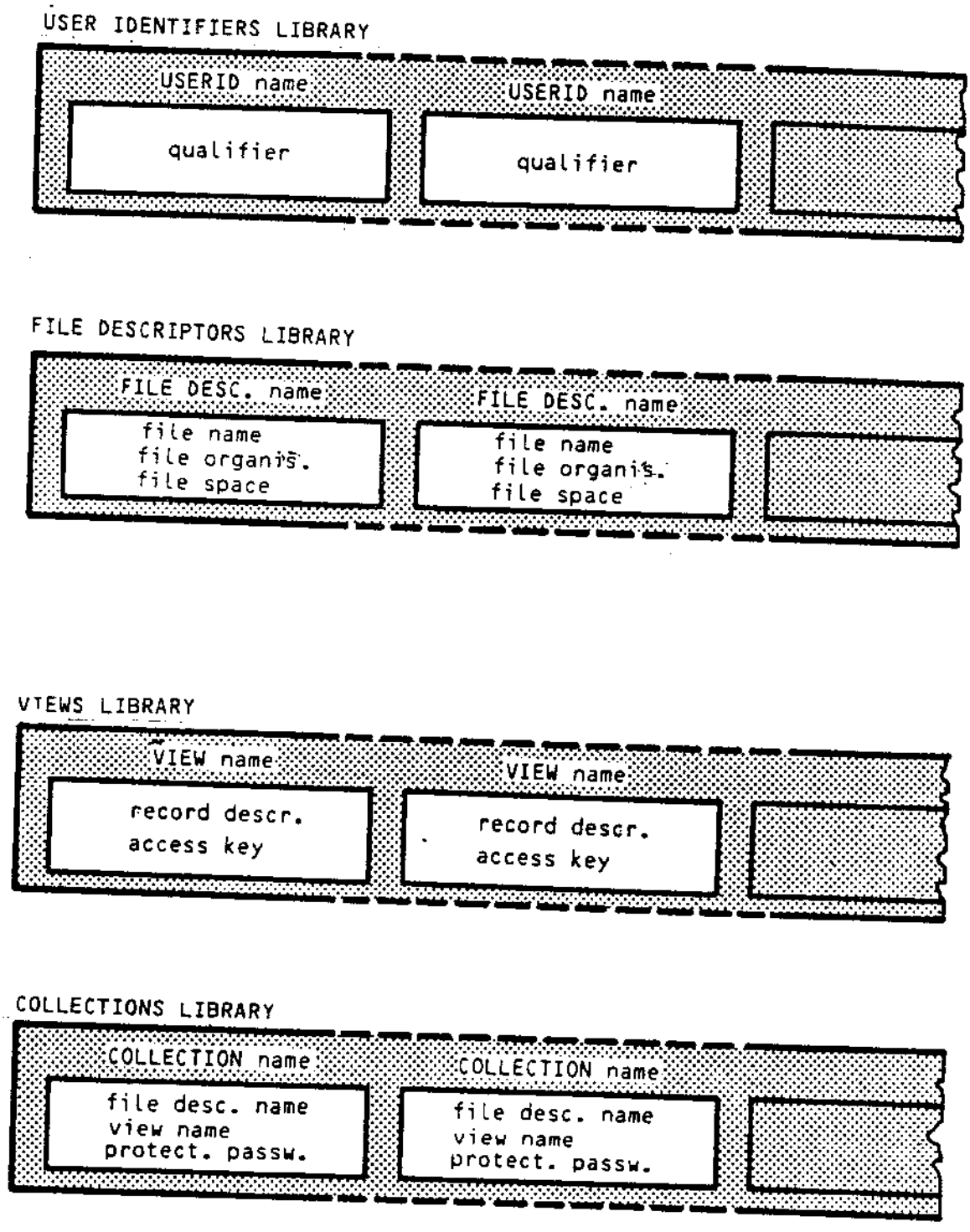


Fig. 3. 2 - DMS Descriptors

USERID name	Is the name of a library element.
qualifier	Indicates the qualifier (data administrator or application programmer) paired to that USERID. In this way only the data administrator (or persons at his level) may create and manage DMS descriptors using the DDL language.

FILE DESC. name	Is the name of a library element.
file name	Specifies the name of the file to which the FILE DESCRIPTOR refers.
file organisation	Specifies the way in which the physical file is organised (sequential or keyed).
file space	Specifies the space which must be reserved for the physical file if it is created using DMS.

VIEW name	Is the name of a library element.
record description	Is the description of the physical file record.
access key	Specifies the record field to be used as an access key if the file is keyed.

COLLECTION name	Is the name of a library element.
file desc. name	Is the name of the FILE DESCRIPTOR to which the COLLECTION makes reference.
view name	Is the name of the VIEW to which the COLLECTION makes reference.
protection password	Is the password to the COLLECTION.

Thus, the information is structured in such a way that reference to a COLLECTION means automatic reference to the contents of the FILE DESCRIPTOR and of the VIEW indicated in the COLLECTION itself.
In fact, the DML language commands available to the user refer solely to COLLECTIONS.

The four types of descriptors are stored in the corresponding Data Dictionary libraries.
For the DMS to operate, a standard Data Dictionary must be set up. At DMS start, that Data Dictionary is automatically connected to it.
The user may nevertheless create and utilise other Data Dictionaries.
Both the standard and any other Data Dictionary are created and handled by the user who executes an appropriate utility program in SHELL environment.

THE DML INTERPRETER

The DML interpreter accepts and executes the Data Manipulation Language commands through which the application user accesses files.

The DML looks for the COLLECTION specified by the user on the Data Dictionary connected to the DMS and, then reads the information held in the FILE DESCRIPTOR and in the VIEW pointed to by that COLLECTION.

At this point the DML has all the information necessary to access the data files.

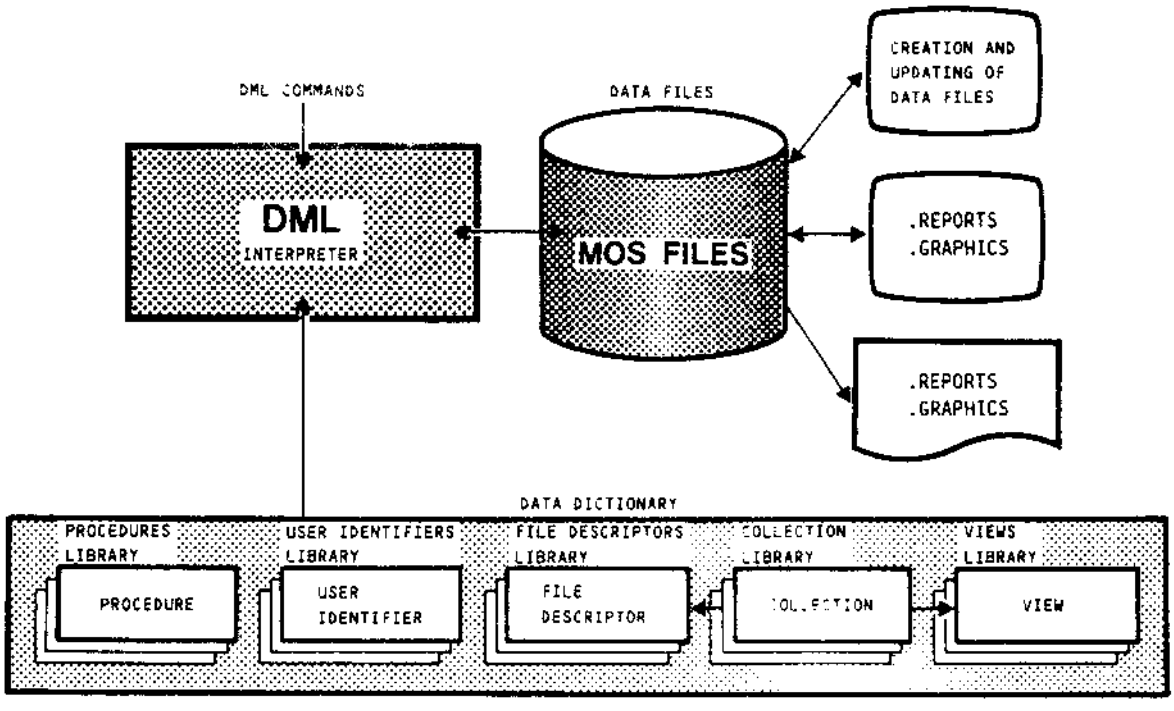


Fig. 3. 3 - DML Commands Access DMS Files

The following figure shows how the information held in a COLLECTION represents the information held in the corresponding FILE DESCRIPTOR and VIEW elements.

This structure provides the virtual file mechanism.

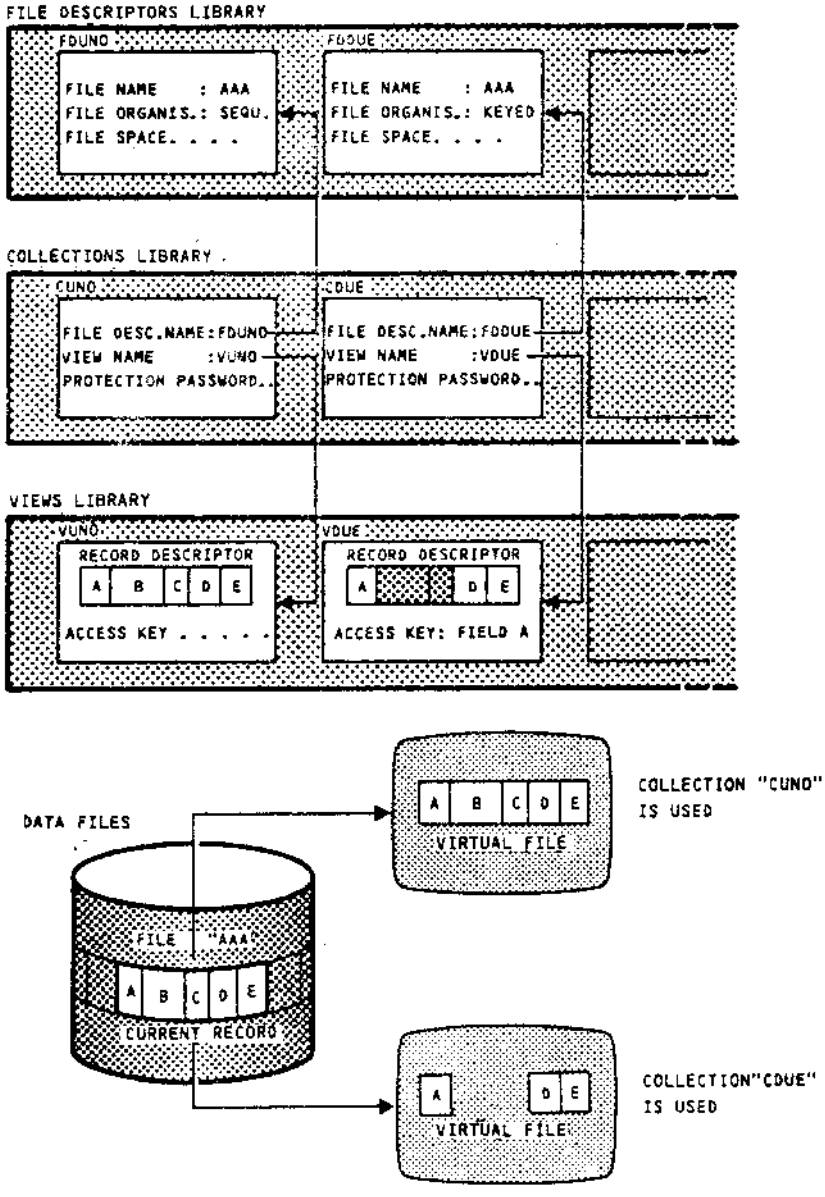


Fig. 3. 4 - Link of the Information to Access a File

THE DMS AND THE MOS ENVIRONMENT

The DMS interacts with the following MOS components as shown in the figure below.

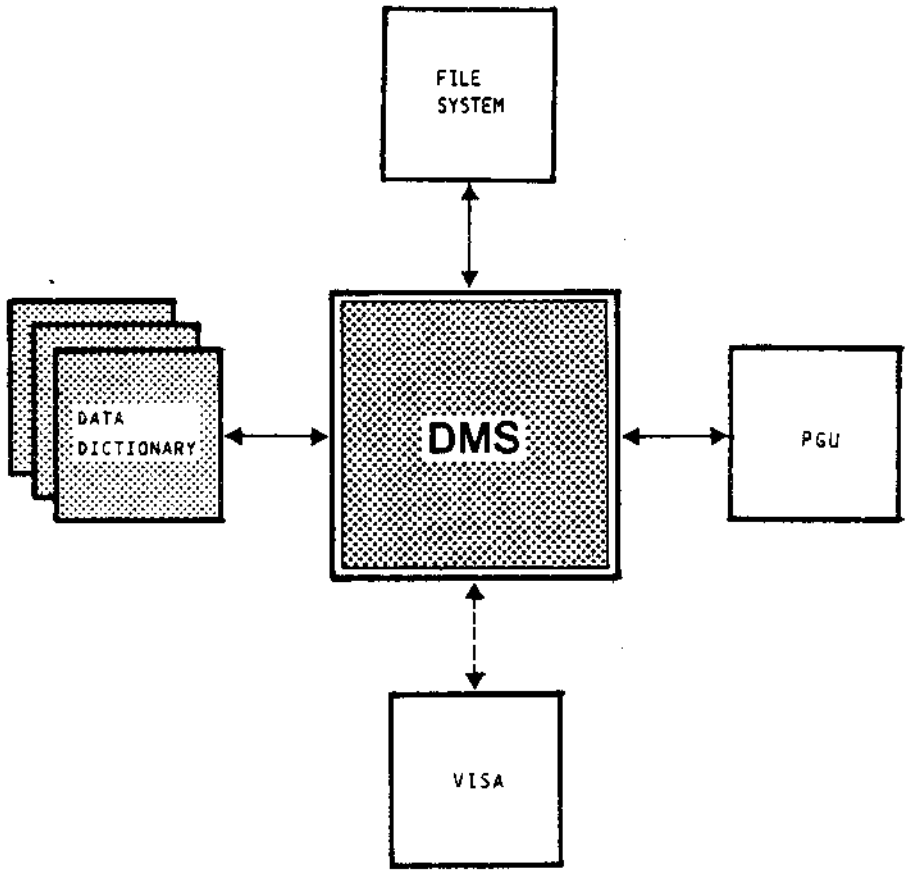


Fig. 3. 5 - Software Objects Interacting with DMS

- FILE SYSTEM: used by the DMS to search for the files, on disk.
- DATA DICTIONARY which contains the DMS descriptors and the user defined DMS procedures. Only one Data Dictionary may be connected to DMS: it is either the standard Data Dictionary or one of the Data Dictionaries successively created.

- VISA (Video Interface System Analizer), which interpretes the forms, if any, by which the user can define the characteristics of the fields and the way of displaying the records.

- PGU (Unified Graphic Package) to draw graphs on the screen or with the printer.

Moreover, the user can utilize:

SYSTEM EDITOR, to create, modify and cancel the command files

TFORM, to create and modify the forms to be interpreted by VISA.

4. DMS LOGIC

This chapter describes the logical path to be followed when creating and accessing a DMS file. A flow representing the sequence of these operations is shown on the next page; some comments follow to describe the figure.

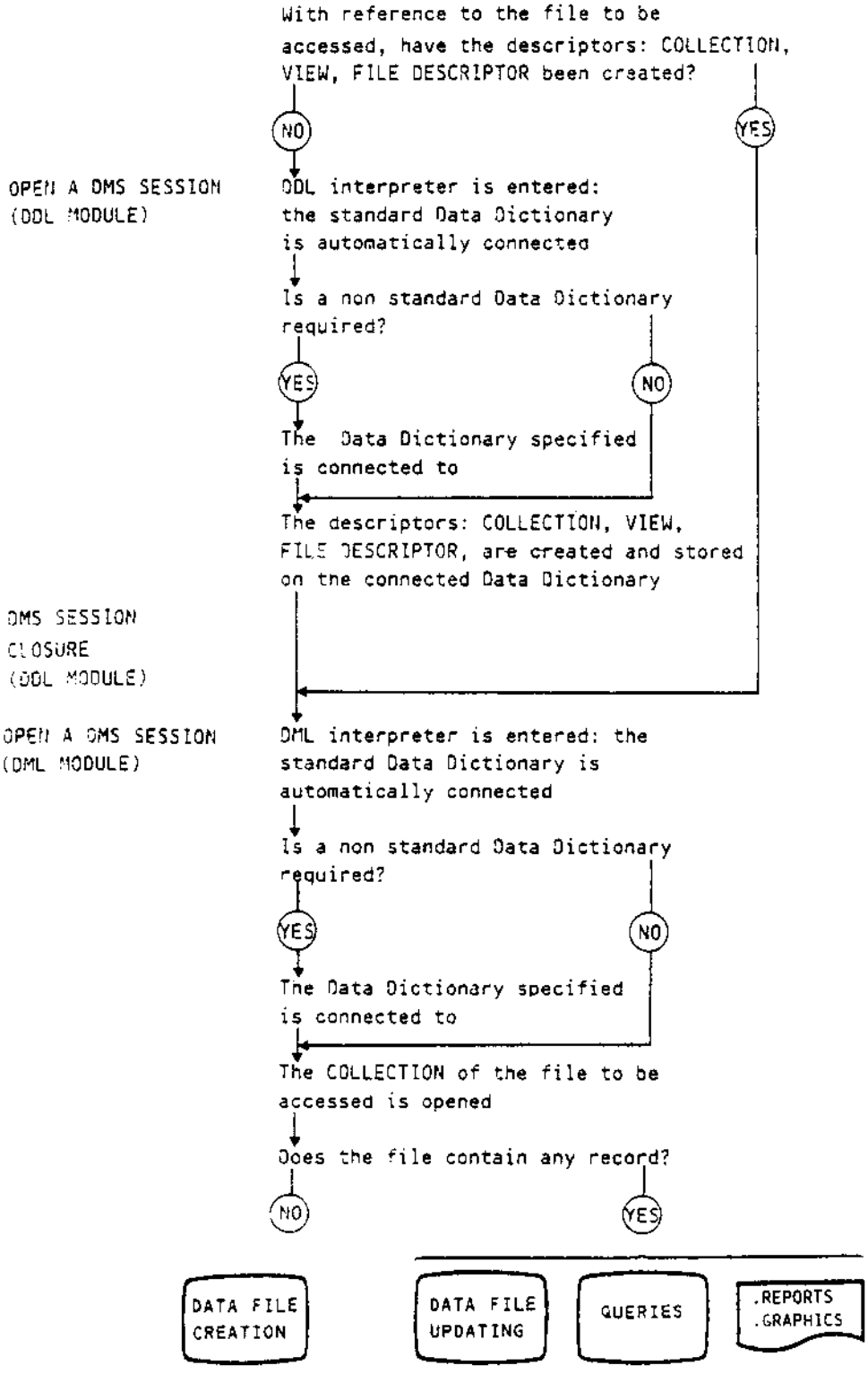


Fig. 4. 1 - Creating and Accessing a DMS File

Some comments follow.

- The creation of the standard Data Dictionary and of any other Data Dictionary is performed by the user in SHELL environment using an appropriate utility program.
  For this reason this step has not been included in the flow.

- Opening a DMS session means connecting to the DDL module or to the DML module giving the appropriate user identifier: at this moment the standard Data Dictionary is automatically connected to DMS.

- The creation of data file descriptors using DDL is reserved for the data administrator.
  As the user refers to a physical file through one of the possible sets of descriptors of that file, (COLLECTION, VIEW, FILE DESCRIPTOR), these objects must exist in the corresponding Data Dictionary libraries.

- Opening a COLLECTION means that the DMS reads in the Data Dictionary the information contained in the COLLECTION indicated and in its corresponding FILE DESCRIPTOR and VIEW.
  This information is held by DMS and used to access the file.
  One or more collections may be opened for the concurrent use of the corresponding files: in this case the last collection opened is known as the "current COLLECTION"
  Each open COLLECTION has a "collection cursor" pointing to a file record known as the "current record".
  When a COLLECTION is opened the first file record is the current record; the user then positions the collection cursor on the records he wishes to access.

The DMS reads the current record from the file corresponding to each COLLECTION opened and copies it into its allocated memory area; the user may therefore work on any of the opened COLLECTIONS, as shown in the following figure.

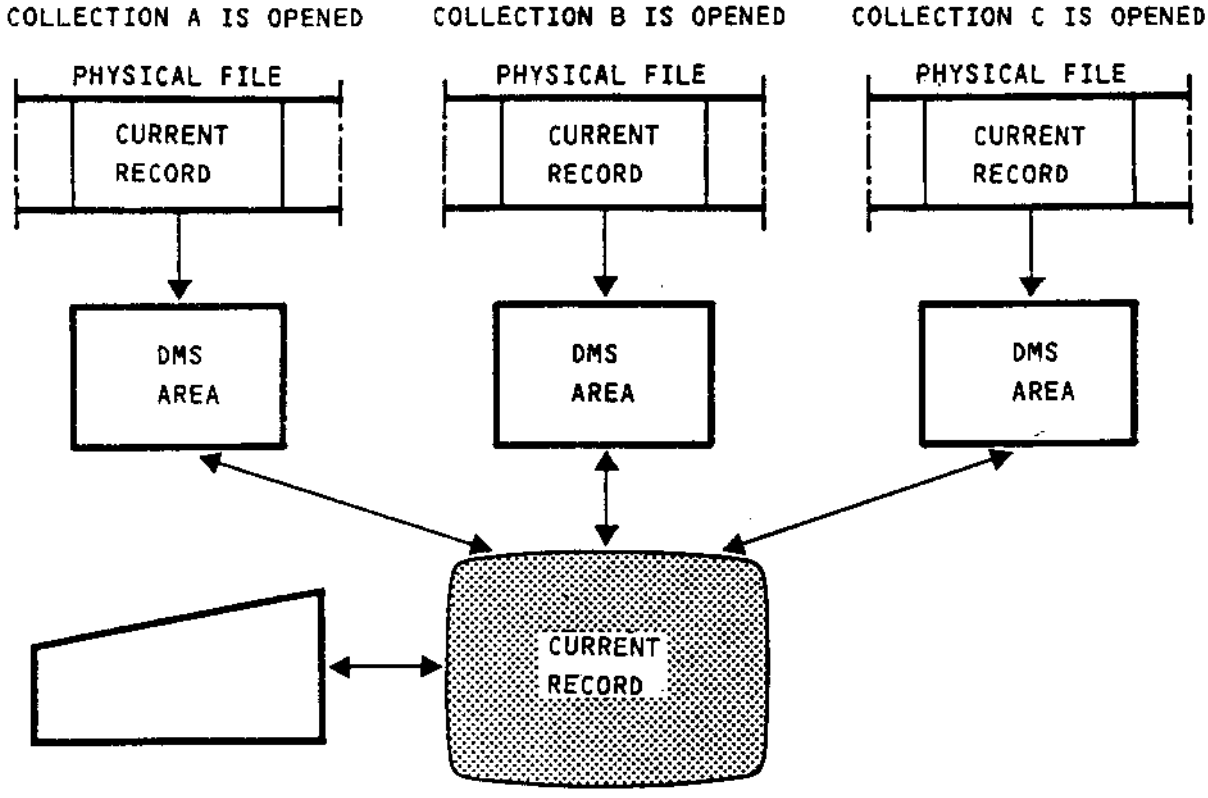
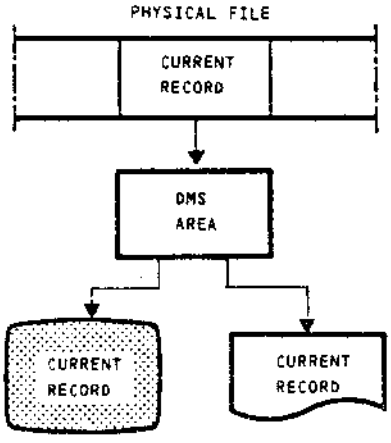


Fig. 4. 2 - More than one File is Accessed at the Same Time

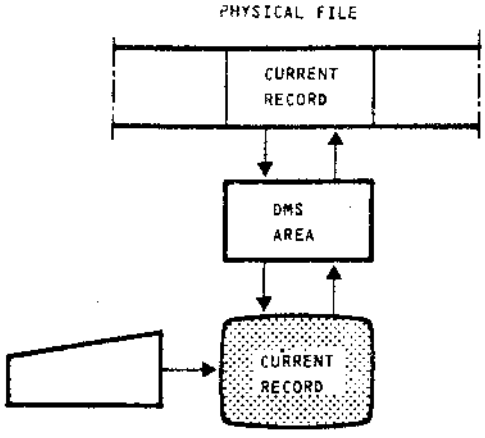
- The physical file may already exist at the opening of a DMS session; if not, it may be created during the session.

  In the first case the file may have been generated in the DMS environment.
  In the second, the opening of the corresponding COLLECTION reserves the physical space on disk for the records to be created.

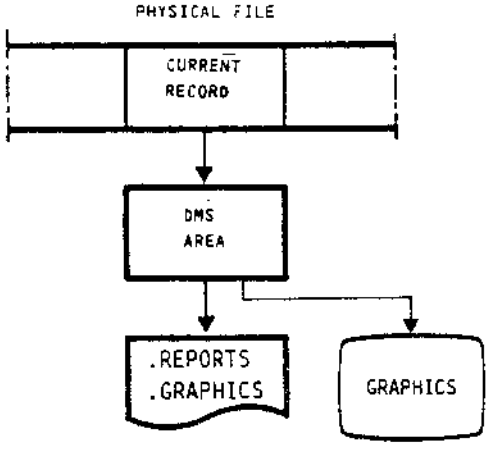
- The use of the record provided by DMS depends on the DML command used, as shown in the following figure.



Display or print.



Display in which field values modifications can be made: changes in the record are re-written in the data file.



Producing Reports or graphics

Fig. 4. 3 - How a File Record can be used in DMS Environment

5. THE DMS FUNCTIONS

This chapter includes a summary of the DMS functions and the following paragraphs:

- The function for definition of data structures.
- The function for definition of DMS procedures.
- The application function.
- The data protection function.
- The editing function.

Because of the nature of the present manual the list of possible operations may be incomplete. Each functional group only mentions those operations characterising the function itself.

Data Structures Definition Function	Creates and manages the following descriptors: USER IDENTIFIERS, FILE DESCRIPTORS, VIEWS, COLLECTIONS It is made available to the data administrator through the DDL language.
Procedure Definition Function	Allows creating and handling of the DMS procedures. It is made available to the data administrator through the DDL language.
Application Function	Allows: - Reference to the existing COLLECTIONS. - Access to the data file records. - Generation of reports and of graphs. - Definition of variables and transfer of values. - Control of the execution sequence of the DMS commands. - Execution of a command file. - Execution of a procedure. It is made available to all application users through the DML language.
Data Protection Function	Manages the various levels of protection offered by DMS using the data structures stored in the Data Dictionary.
Editing Function	Allows the user to move the cursor freely on the screen and to modify the values displayed.

THE DATA STRUCTURES DEFINITION FUNCTION

It is represented by the DDL commands and its use is reserved for the data administrator.
It allows reading and writing on the Data Dictionary for the:

creation, deletion, display

of the file descriptors of the files managed in the DMS environment.

The following is a list of commands related to this function.

CONNECT	Connects the specified Data Dictionary.
USERID	Creates a user identifier-type descriptor.
FILE_DESCRIPTOR	Creates a file descriptor-type descriptor.
VIEW	Creates a view-type descriptor. In order to guide the user in the record description, the VIEW command can be executed in interactive way, driven by the DMS.
COLLECTION	Creates a collection descriptor.
REMOVE	Deletes the descriptor indicated in the corresponding library.
DESCRIBE	Displays the names of the descriptors contained in the specified library.
EXEC	Executes the specified command file. The following should be noted:

- In the data structure definition environment the command file must contain only DDL command (the DDL interpreter is loaded into memory).
- A command file may call another command file.
- When the command file has been executed the DMS passes control to the next instruction, if any.

THE PROCEDURE DEFINITION FUNCTION

It is made available to the data administrator through the following DDL command:

PROCEDURE — Creates the procedure and memorizes it on the Data Dictionary.

A procedure is a sequence of DML commands stored in the procedure library of the Data Dictionary.
The following characterizes a procedure:
it is possible to define and use symbolic parameters to which the user assigns an actual value when the procedure is executed.

A very simple example;
the procedure named, for instance, 'sphere', which calculates and displays the volume of a sphere is:

```
PROCEDURE sphere (r)
SHOW (4*3,14*r*r*r)/3
ENDPROC
```

where 'r' indicates the radius of the sphere.
To calculate the volume of a sphere whose radius is 5 the user simply provides the procedure name and value 5 (see Execution of a Procedure in THE APPLICATION FUNCTION).

THE APPLICATION FUNCTION

Application users (and usually the data administrator) may invoke application functions through the DML commands.

The main application functions are described briefly subdivided into functional groups.

Reference to the Existing Collections

The commands for this group allow:

- The connection to a specified Data Dictionary, making the libraries in which the descriptors are stored accessible to the DMS.

- The opening and closing of the COLLECTIONS used in the DMS session.
  When a COLLECTION is opened, the user must declare if the corresponding file is reserved for read only use or if it may also be modified; the DMS effects protection controls in this phase.

The following is a list of commands related to this group:

CONNECT	Connects the specified Data Dictionary.
OPEN	Opens the specified COLLECTION.
CLOSE	Closes the specified COLLECTION.

Access to the Data File Records

When a COLLECTION is open, a data file record is present in the DMS memory area reserved for that COLLECTION; this is the "current record".
It is pointed to by the collection cursor and is available to the user for:

- Display.

- Modification.

- Creation of a new record.

- Deletion.

The following is a list of commands related to this group.

POSITION Positions the collection cursor under a given record. The user may specify the algorithm needed, if he wishes to search for a specific record.

MODIFY Displays all the current record fields (or certain selected fields) and allows modification of the field values.
The modifications take place in the DMS memory area; when the command has been executed they may be stored in the data file by writing the record back to the file.
This command may require the use of a specified form to display and handle the record data.
In this case DMS uses VISA for interpreting the form.

CREATE Creates a new record; the record created is:

- Added on at the end of the file, if the file organisation is sequential.

- Inserted in the appropriate position according to the value of the key field, if the file organisation is keyed.

The command starts with the display of the record structure in order to guide the user in the introduction of values.
This command may require the use of a specified form to display and handle the record data.
In this case DMS uses VISA for interpreting the form.

DELETE The current record is no longer available as long as the COLLECTION remains opened: it deletes logically but not physically. Physical deletion is caused by the command EXPUNGE which must be executed before closing the COLLECTION.

If this is not done, the record will be available again after the COLLECTION has been closed and then re-opened.

EXPUNGE — Physically deletes the data file records for which the DELETE command has been executed during the current DMS session.

DISPLAY — Displays the fields specified by the user. These fields belong to the current records of the COLLECTIONS open.

File Chaining — This function allows the user to correlate two DMS files in such a way that moving the collection cursor of one file causes an appropriate movement of the cursor in the other file.
The chaining can involve more than two files.
The following is a list of command related to this group.

RELATE — Chains a file (secondary file) to another one (primary file).
The command specifies two fields (one for the secondary file and one for the primary file) which define a relation as follows:

when the user moves the collection cursor of the primary file, the collection cursor of the secondary file moves to the first record which satisfies this condition:

the content of the field specified for the primary file is equal to the content of the field specified for the secondary file.

UNRELATE — Ends a chaining previously set up.

ADVANCE — Executes a forward search in the secondary file for another record which satisfies the above mentioned condiction.

Generation of Reports and Graphs

This function allows information to be outputed; this may be:

- Displayed on screen.
- Printed.
- Stored on magnetic tape.
- Stored on disk.

The data is obtained by the DMS which uses the records of the indicated COLLECTION.

The following is a list of commands related to this group.
At the end of the list a brief description of the DMS graphic feature is provided, to give a more precise idea about its possibilities.

REPORT

Generates output using the records of the indicated COLLECTION. Each record produces a logical output line.
The REPORT command has the following characteristics:

- The user may specify selective criteria which indicate the COLLECTION records that are to be considered when formulating the report.
- It is possible to specify certain options regarding the editing of the report pages.
- Output data may be the original values contained in the file records or the result of calculations carried out by the DMS on these values.

PRINT

Outputs a specified list of elements.
An element may indicate:

- The field contents of the current record in one of the open COLLECTIONS.
- A variable.
- A string of characters.
- An expression, whose value is first calculated and then sent to output.

SHOW Operates in the same way as the PRINT function, except that the specified output data is only displayed on screen.

PLOT Produces histograms, on the screen or with the printer.

PIE Produces piecharts, on the screen or with the printer.

DRAW Allows the execution of graphic operations of different type.

The DMS graphic features are briefly described.

The screen can be subdivided into rectangular regions called windows. Windows are independent from one another and there may be a maximum of 16 on the screen.

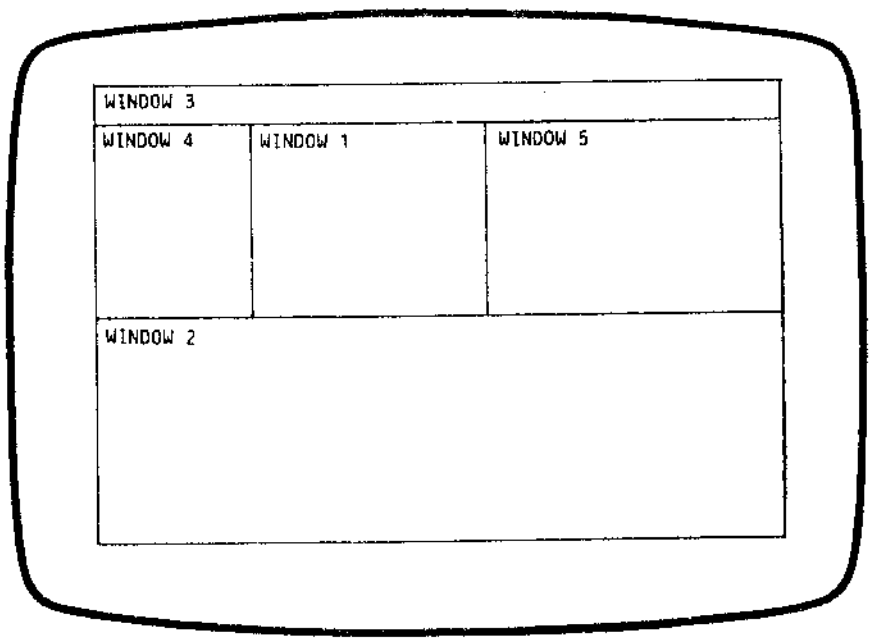


Fig. 5. 1 - Windowing

Single pixels, lines, polylines, arcs, circular segments, circular sectors, circles, ellipses and boxes of varying size may be drawn. Circles, ellipses, boxes and any other closed region may be solid filled in any of eight available colours (black, green, red, yellow, blue, cyan, magenta and white).

Seven linestyles are available. The system provides one of these, the solid line. The other six may be defined by the user. This is useful, for example, when drawing a graph containing more than one curve each one of which must be clearly differentiated form the others.
The user may then choose to represent one curve with a solid line, one with a dashed line, one with a dotted line and so on. The width of lines may be adjusted.

The user can also use a file ("image file") for recording all the commands corresponding to the graphic statements which are executed after the file has been opened, and for redisplaying the graphic image described in the file at any time.

A printed document (hard copy) of the displayed window may be obtained on the Olivetti graphics printers (for example, PR2400, PR1450).

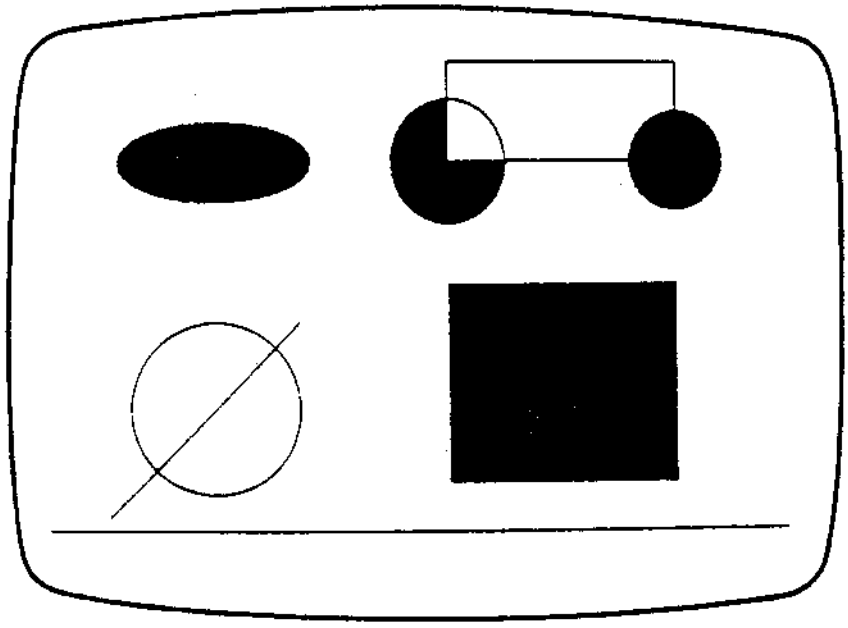


Fig. 5. 2 - Examples of Some Geometrical Figures

As shown in the example above, drawings can be superimposed.

The two following examples represent one histogram and one piechart.

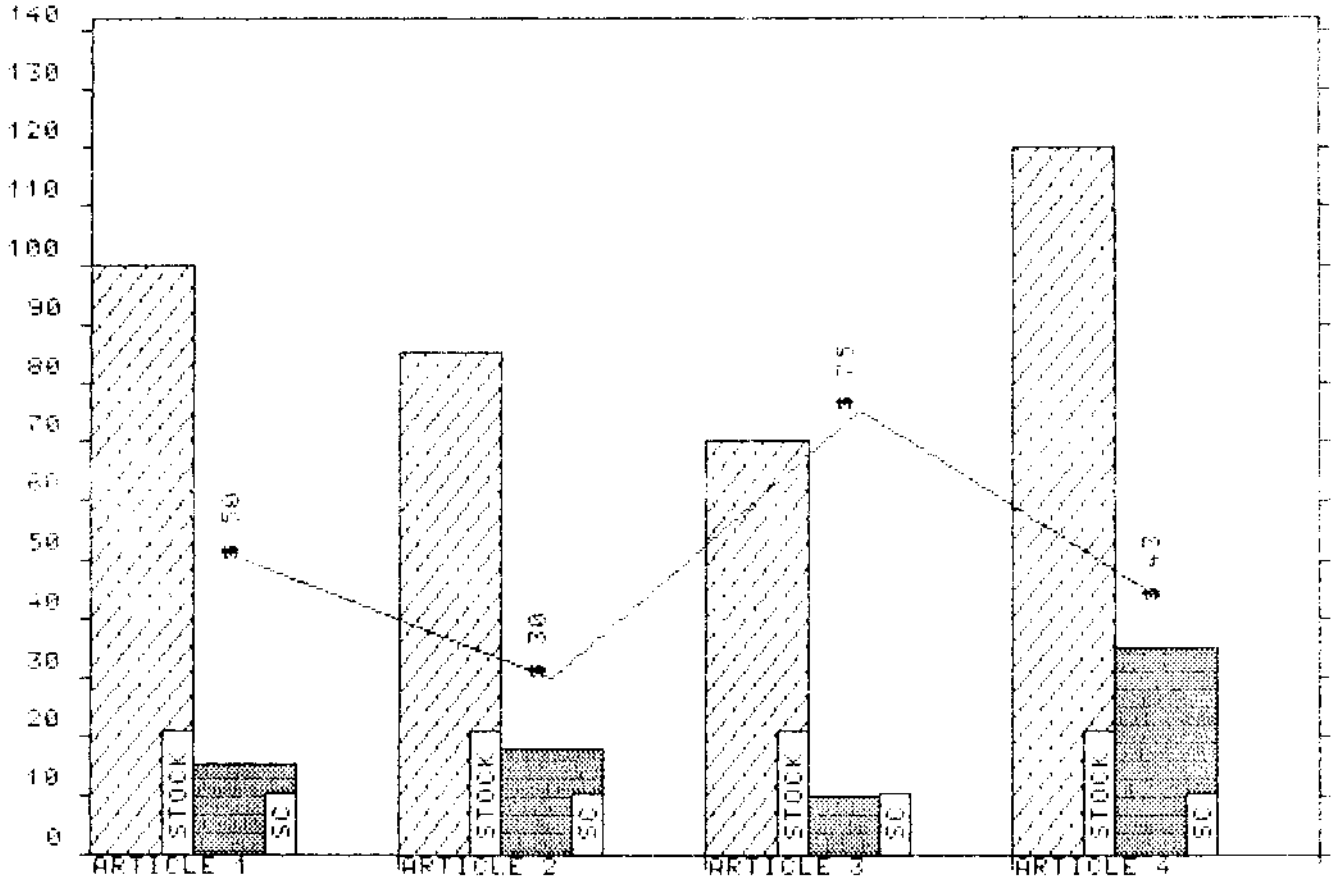


Fig. 5. 3 - Example of an Histogram

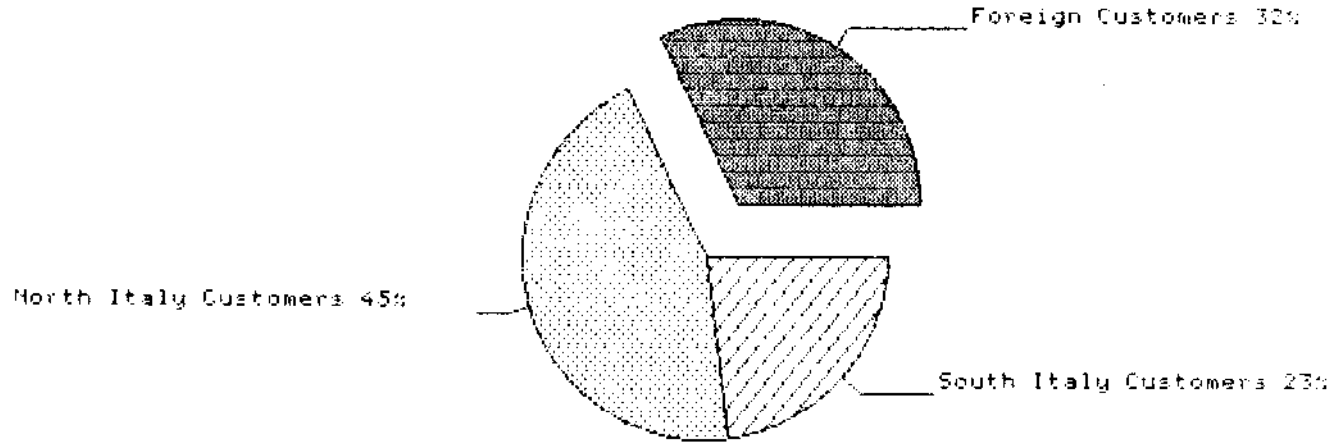


Fig. 5. 4 - Example of a Piechart

The user may rotate and/or translate the graphic axis, and thus also the graphics images contained within them.

Pictures, or parts of pictures, may be stored in a matrix and a file and redrawn when necessary. Text strings may be written on the screen in any direction.
The size of the text strings may also vary.

They can be displayed in one of eight different character set fonts, which are:

- Alphanumeric string precision

- Simple block letters
- Double block letters
- Complex block letters
- Complex italics
- Cyrillic
- Greek
- Gothic

String characters may be scaled up to 16 lines the default value.

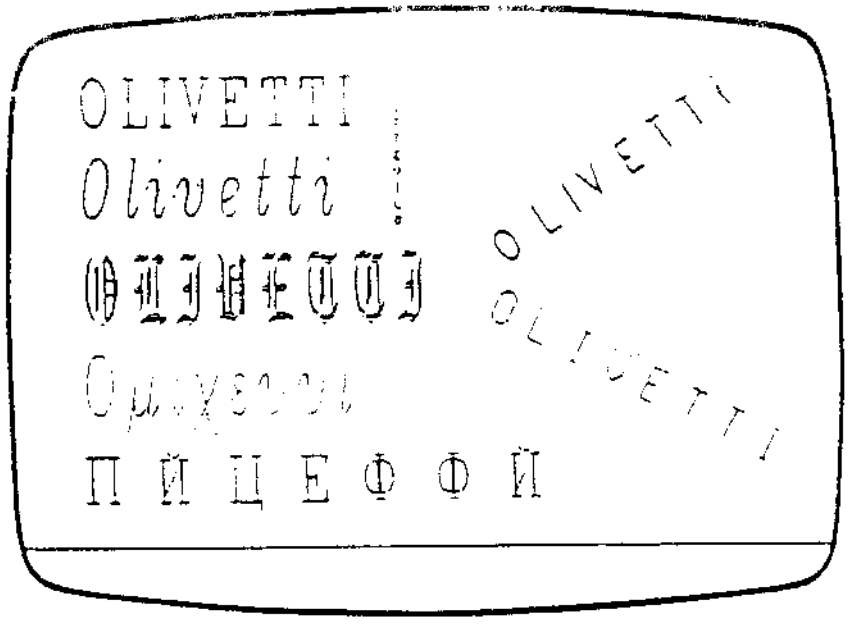


Fig. 5. 5 - Example of Character Strings

Alphanumeric texts and drawings may appear in the same window. This can be usefulm for example, when drawing a graph and labelling its axes.

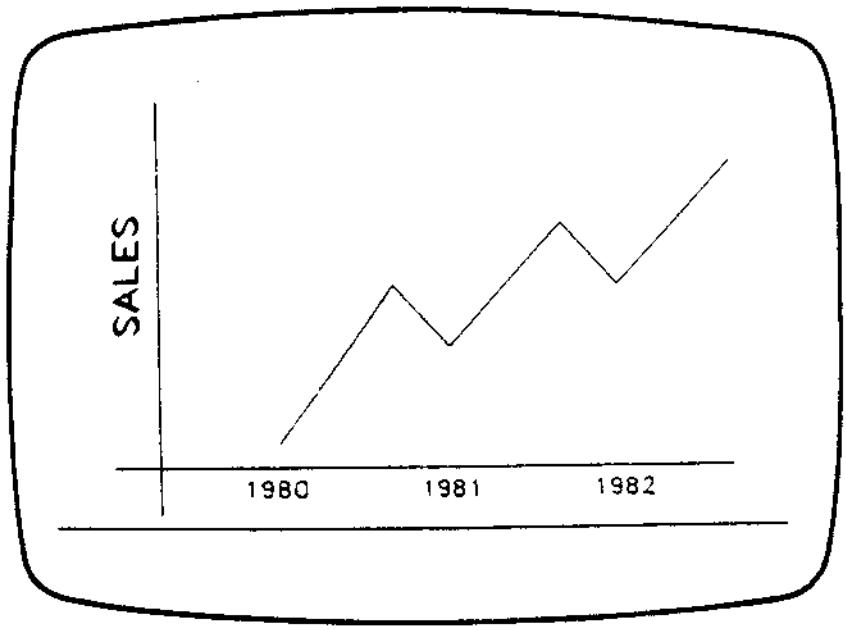


Fig. 5. 6 - Example of a Graph

The user may define particular coordinate systems in order to describe the graphs displayed.

Definition of Variables and Transfer of Values

The DMS allows:

- The allocation of internal variables; these are values defined and used by the programmer in the course of the DMS session.

- Testing of certain boolean variables whose values, handled by DMS, allow the user to know if certain conditions hold.
  For example, the "SEOC" symbol is the boolean variable which indicates if the "end of collection" condition holds or not.

- The assignment of a value to a variable or to a field of the current record.
  This value may be represented by:

  . the contents of a field of the current record

  . a variable

  . a string of characters

  . the value of an expression.

The following is a list of commands related to this group.

ALOC — Allocates the variables whose names are specified in the list.
The variable type is automatically defined by the last value assigned to that variable.
Thus, if a variable contains the numeric value 123 at a certain point in time, it is numeric.
If the string 'ABC' is later transferred to the same variable, that variable becomes a 'string' type variable.
It is important to note that a string of characters may represent an expression. For example, the string "1 + 1".
In such a case it is possible to refer either to the formal contents of the variable (that is the string "1 + 1") or to the calculated variable value (that is the numeric value 2).

DEALOC — Deallocates the variables whose names are specified in the list.

MOVE	Assigns a value to a variable or to a field of the current record.
Control of the Command Sequence	This function comprises certain commands through which it is possible to: - Specify certain conditions and verify their existence. - Define a group of commands which is to be repeated cyclically. - Interrupt the cycle of a group of commands. The following is a list of commands related to this group.
IF.....ENDIF	Establishes a condition indicating the commands which are to be executed if that condition holds.
REPEAT.....END	The commands placed between REPEAT and END are executed in a cyclical manner.
BREAK	Brings the system out of the command cycle.
Execution of a Command File	It allows the execution of the DMS commands stored on file. The command is:
EXEC	Executes the specified command file. The following should be noted: - In the application environment the command file must contain only DML commands (the DML interpreter is loaded into memory). - A command file may call another command file or a procedure. - When the command file has been executed the DMS passes control to the next instruction, if any.

Execution of a Procedure

It allows the execution of the DML commands which constitute the procedure.
The user executes a procedure by indicating the procedure name followed by the values to be assigned to the symbolic parameters, as follows:

```
procedure-name   value 1 value 2 ..... value n
```

For instance, with reference to the example given in THE PROCEDURE DEFINITION FUNCTION, the volume of a sphere whose radius is 5 is obtained by writing

```
sphere 5
```

A procedure may call another procedure or a command file.
When the procedure has been executed the DMS passes control to the next instruction, if any.

THE DATA PROTECTION FUNCTION

The DMS data protection allows:

- Protection of the objects stored in the Data Dictionary (USER IDENTIFIERS, VIEWS, FILE DESCRIPTORS, COLLECTIONS).

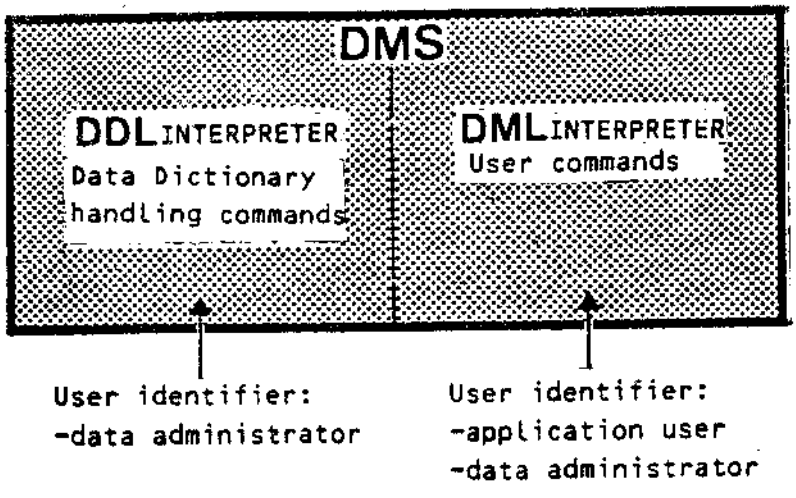


Fig. 5. 7 - Data Dictionary Protection

Only the Data Administrator or authorised persons may connect to the DDL interpreter to manage the DMS objects stored in the Data Dictionary libraries.

In fact, the Data Administrator assigns a user identifier to each DMS user and pairs the identification word to one of the following roles:

. application user

. data administrator.

The protection mechanism is entered when the user, connecting to DMS, must specify both the module to which to connect (DDL or DML interpreter) and its own user identifier.

- The exclusion of certain users from accessing particular record fields.
  This protection, if necessary, assumes that different COLLECTIONS are defined for the same file.

  As each COLLECTION points to a VIEW containing the description of the file record, a complete or partial record description allows the user a complete or partial view of the data.

- Two different methods for accessing data:

  . read only

  . read with the option of modifying.

  The Data Administrator associates a list of passwords with each COLLECTION; each password, in the list, is represented by a pair of names:

  . the password itself

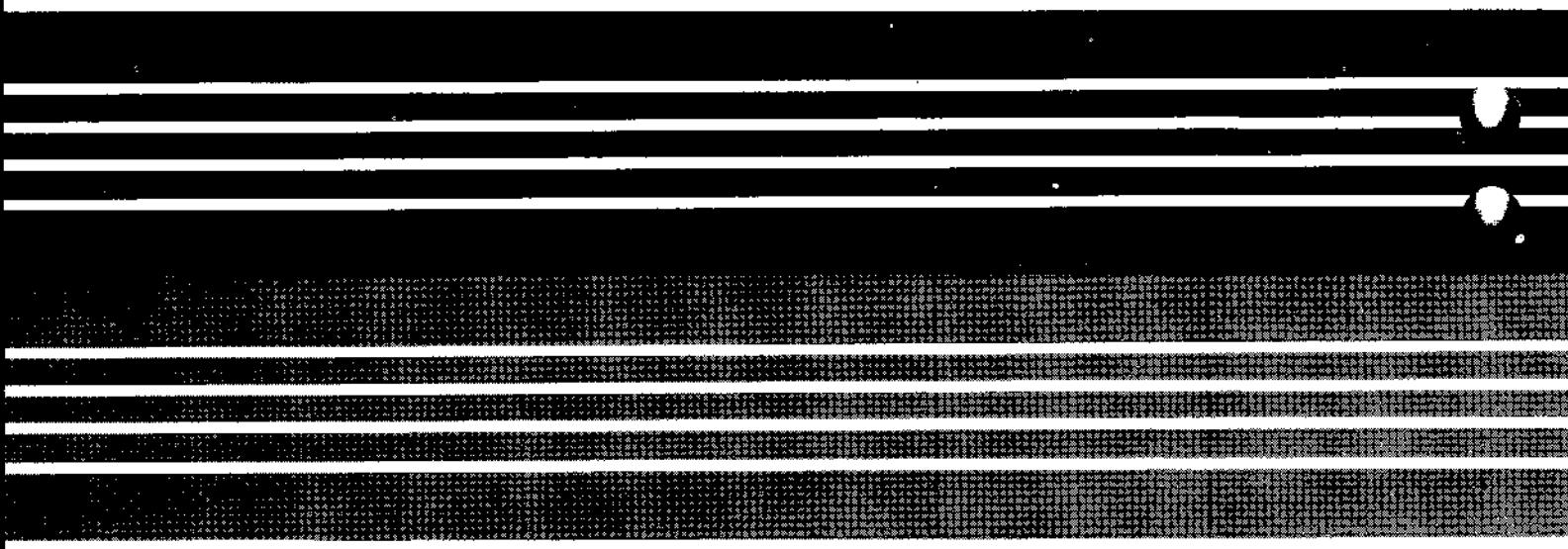
  . the method for accessing data, which can be either READ or MODIFY.

  The user may access a COLLECTION only if he supplies a "password - method of access" contained in the COLLECTION'S protection list.

The DMS is an interactive package which makes use of the screen to communicate with the user.
Screen handling is simple and has the following characteristics:

- Each screen line contains a value (usually the contents of a record field) and the name which identifies that value.
- When a screen page is displayed, the cursor is positioned under the first value and the user may operate on it, after which the cursor moves automatically to the next value; this continues until the end of the page.
- If the information to be displayed cannot be contained on only one page, the DMS automatically displays the next page when the current page has finished.
- The user may move the cursor within the current field or under any displayed field on which he intends to operate, by making use of the appropriate keys.

Note that when DMS uses VISA to interpret a form, the operating rules valid in the VISA environment must be followed.



Code 4002610 Q (2)
Printed in Italy