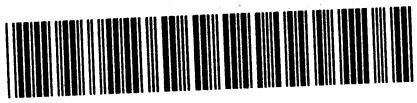
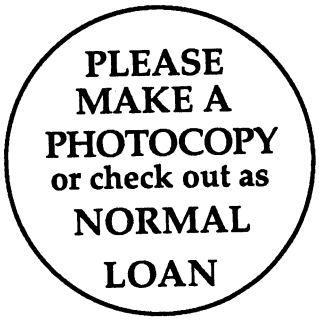
Data Handling Division
DD/87/18
October 1987



CERNDOC - A document filing and retrieval system

L. Esteveny and E. van Herwijnen



(Submitted to Software - Practice and Experience
Presented at the SHARE conference, Chicago, August 1987 and
SEAS Anniversary Meeting, Edinburgh, September 1987)

DD-vf

CERNDOC - A document filing and retrieval system

Eric van Herwijnen, Laure Esteveny
User Support Group
Data Handling Division
CERN, Geneva
Switzerland

Installation Code: CER

Office Integration Project

Session No. G812

Summary

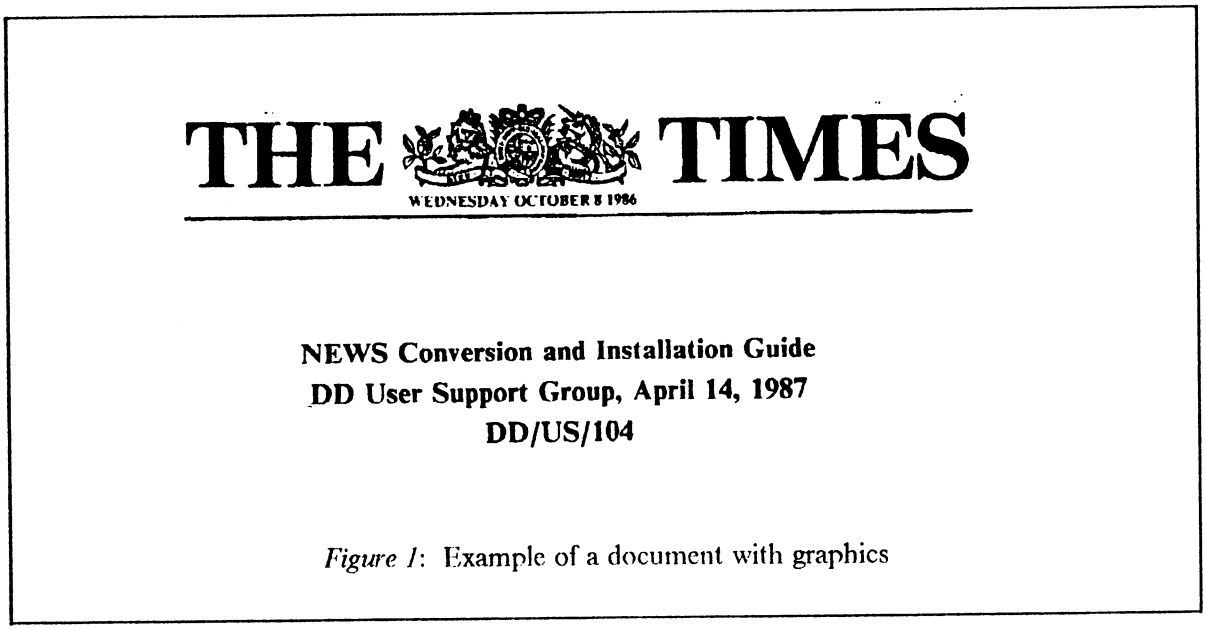
Machine readable documents are produced more and more frequently due to improved word and text processing facilities. Many user communities would benefit from a well thought-out storage scheme for such documents which would also allow for easy access, printing and distribution. The CERNDOC documentation system was developed at CERN, the European Laboratory for Particle Physics, to solve these problems for a large scientific user community. CERNDOC allows the storage and retrieval of any (formatted or marked up) text or binary data file (including MacIntosh 'Macpaint' format bitmaps) and closely interfaces with a local SGML implementation. The system is written in REXX and installed on an IBM 3090-200 running VM/CMS. Non-interactive access may be obtained from network nodes by sending MAIL. Plans for a machine-independent version written in Fortran 77 exist and may be implemented if the system proves to be popular.

(To be submitted to Software - Practice and Experience)

1. Introduction

CERN, the European Laboratory for Particle Physics, is a centre whose facilities are used by many different universities and research institutes all over the world. Although the aim of CERN is high energy physics research, a large amount of activity is centered around internal documentation. A fairly representative cross-section of documents produced by the organisation includes letters to manufacturers, internal memoranda, agendas and minutes of hundreds of working groups and committees, scientific reports and papers, manuals describing use of apparatus or software, and so on.

With the advent of dedicated word processing systems such as WANG, WORDSTAR and MACWRITE, as well as increasingly powerful text processing facilities on mainframe computers, for example IBM's DCF, Waterloo SCRIPT, NROFF/TROFF and TEX, it has become very easy for secretarial and non-secretarial staff alike to produce many of the documents mentioned above in electronic form. This phenomenon has been the cause of a significant increase in the flow of documents in very many different formats. Documents may contain graphics (see figure 1); they may be input for various text processing systems; they may or may not be computer readable, and they may not even be stored at CERN.


THE TIMES

WEDNESDAY OCTOBER 8 1986

NEWS Conversion and Installation Guide
DD User Support Group, April 14, 1987
DD/US/104

Figure 1: Example of a document with graphics

The high energy physics user community is naturally divided into groups associated with the experiments that are in progress or preparation at the laboratory. These groups of users are large and the physical location of their members is very diverse. Indeed, a large part of CERN's user community uses its facilities from home institutes rather than at CERN. To avoid duplication of effort and to share knowledge a single documentation system is required which is central, properly supported, coherent and accessible to as many users as possible.

1.1 General requirements of a documentation scheme

An organisation which fits in well with the structure of the high energy physics community is one where documents are made available for a group of users and which also may be used as a filing system for private individuals. This may be achieved by defining **projects**, containing documents of interest to a (group of) user(s). Informal news-type facilities should be available for informing interested users of the existence of documents. For more formal projects it should be possible to define rules by imposing certain **attributes** on classes of documents such as, for example, reference numbers, or a choice of text formatter to assure a common appearance. It goes without saying that the retrieval facilities should be simple, interactive and fast.

An important issue in the communication between the different groups at CERN is the fact that they will not necessarily share the same hardware. For example, a user whose home institute is the University of X in country Y, may want to receive any updates on a given project at CERN. Frequently this documentation will contain graphics and a device with the capability to process bitmap graphics may be needed to print it (see figure 1). Printers like this are available at some locations in CERN, but not necessarily at other institutes. It should, therefore, not only be possible to receive a copy of the document electronically through the a network, but also to obtain a version that is suitable for displaying/printing on the available hardware.

To enable the inspection of documents stored in the database on simple terminals, it was decided that CERNDOC should format them (whenever necessary) to allow the display on a standard ISO646 ASCII screen. A version of the document in figure 1 formatted for a simple terminal and displayed by a CERNDOC utility may be seen in figure 2.

```
==> USERGUID NEWSINST <======> HELP  INFORMATION <======>line ===> 1 of 821

                    NEWS Conversion and Installation Guide

                     DD User Support Group, April 23, 1987

                                  DD/US/104

        ------------------------------------------------------------------
        NEWS                                                April 23, 1987

        ------------------------------------------------------------------

                                  Preface

        This guide  is intended for people  wishing to install  the new
        (31 March 1987) CERN news system (at CERN or elsewhere)  or to
        convert their existing group news system.

1= All      2= Top     3= Quit   4= Return        5= Clocate      6= ?
7= Backward 8= Forward 9= PFkey 10= Backward 1/2 11= Forward 1/2 12= Cursor

====>                                                  Macro-read 3 Files
```

Figure 2: Example of the same document formatted for a simple terminal

This simplified version of the document may be printed on any lineprinter. However, a document may contain many mathematical formulae which will be omitted in the simplified version and become unreadable without these. The author may wish to indicate explicitly the printer on which the document should be printed. If a document cannot be displayed on the terminal, the user is informed of this fact.

To facilitate the expansion of on-line HELP documentation systems it would be useful to have the formatted documents in the scheme [1] included in the system HELP facility. The retrieval facilities offered by HELP may then be applied to all the items in the database, and the HELP environment will be known to the users. It also provides some facilities for users that log on to VM in line-mode and are unable to scan documents in full-screen mode. Although the use of HELP allows the application of CERNDOC to the documentation of items in local program libraries, the problem of automatic software documentation management such as the SOFTLIB [2] system is not addressed.

1.2 Detailed requirements and accessibility of the system

The interactive classification and retrieval of documents in a central, coherent and easily accessible scheme poses formidable problems. First of all, the database part of such a scheme requires large amounts of storage space. One possibility is to use a large central mainframe, possibly as a network file-server. The mainframe in use at CERN is an IBM 3090 model 200 running VM/CMS, which permits the creation of an interactive system, but the archaic properties of its filing system are well known [3]. The need for confidentiality also complicates matters, particularly on VM/CMS, in the absence of access control lists which do exist in other systems [4]. Despite the disadvantages of VM/CMS, it was decided to implement CERNDOC on this system because of its central position at CERN both in terms of support and accessibility.

Nowadays, large networks such as EARN/BITNET, DECNET and USENET are more and more reliable and accessible, so that the problems concerning distribution are in principle solvable. Two solutions immediately come to mind: one may take the approach of storing the documents in a distributed database or one may use a central database. The former has the advantage of allowing updating and retrieval to take place in the user's local (preferred) environment. However, some non-trivial updating problems remain to maintain integrity of the database in this case [5]. CERNDOC has been implemented using a central database, i.e. the documents are stored and managed on one system, but facilities for access from connected networks are provided. This avoids the difficulty of sending and keeping track of updates to the various distributed parts of the database; on the other hand, it could imply increased network traffic. The system informs specified groups of people via electronic MAIL when changes occur to categories of documents, but no document is sent. If necessary, a command is available to send documents to the same group of people; but any interested user may simply send MAIL to CERNDOC from the remote computer containing a request in a special format.

Before the development of CERNDOC, various systems were in use at CERN, the most popular being DOCP [6]. This system was implemented on the MVS/Wylbur system and written in the Wylbur exec language. DOCP imposed a certain structure on the registration information kept with a document, and in general its users considered it essential that the status, reference numbers, titles and other data be kept in an easily retrievable form. The popularity of this scheme emphasises the importance of being able to define rules for groups of documents.

In this paper, we describe how CERNDOC attempts to provide a solution and a unique CERN interface to the problems mentioned above. As a by-product, a system emerged that provided a very

useful way of tagging CMS filenames with comments (see figure 10) - a feature often present in other operating systems and sorely missed in CMS.

2. Consequences of the use of VM/CMS for the implementation

Many utilities needed for the documentation system were already available on VM/CMS, either provided by IBM or locally written, and it was decided that the best possible use should be made of these. Although the use of the VM/CMS operating system certainly imposes certain restrictions, it was relatively easy to integrate CERNDOC with tools for the classification, formatting and retrieval of documents. CERNDOC is written in the REXX command language and consists of a collection of EXEC files (system commands) and XEDIT macros (editor commands).

A Users' Guide exists, describing the system in more detail [13].

2.1 A service machine for CERNDOC

Part of the system runs in a so called **disconnected service machine**. When a user logs on to VM, a **virtual machine** is allocated to his userid. Each virtual machine may have one or more areas on disk allocated to it where files may be stored. These areas are called **minidisks**. A service machine exploits the very practical feature of VM/CMS allowing userids to be logged on in a special mode which permits the execution of programs (which may interact with other users), without being physically connected to a terminal. Userids that are used like this are also called **batch** machines and may be compared to **background** or **detached** processes in other operating systems.

The idea of a service machine managing documents came from SYSDOC [7], a documentation system for systems group documents of the Computing Division at the Rutherford Appleton Laboratory. The advantage of using a service machine is that it handles all incoming requests automatically: the machine is normally dormant, and is activated when a request arrives (for example MAIL) or when a special message is sent. The incoming messages and files are queued and acted on one-by-one on a first-come first-serve basis. Short tasks, however, for example password validation, are carried out in preference to longer tasks.

Finally, the use of a service machine also permits a safe organisation of minidisks which many users may simultaneously want to read from or write to, traditionally a serious problem in VM/CMS. For these reasons, one is almost forced to use a service machine.

2.2 Portability of the system

The system is transportable to any other VM/CMS site which has the HEPVM tape installed [8], i.e. which has the CERN local REXX function package RXLOCFN, the FIND keyword search facility which runs on top of HELP, and the DIOF (IOS3270) full screen menu package [9]. The MAIL EXEC [10] is not part of the HEPVM software, but having it installed enhances the functionality. CERNDOC was set up using XPRINT, the HEPVM printing facility which allows formatted documents to be printed at any printer in the major laboratories in the HEPVM collaboration.

Due to limitations on disk space, an interface with the VM Software product VM ARCHIVE is provided, but the system will work without this. CERNDOC does not depend on any modifications to the VM operating system.

2.3 Text formatters

To format documents one needs a text formatter. At CERN, the supported text formatter on VM is Waterloo SCRIPT (from version 6.1 upwards). It was therefore decided that CERNDOC should use SCRIPT, but the design does not depend on the details of neither SCRIPT nor XPRINT.

With the acceptance of SGML, the Standard Generalised Markup Language as an ISO standard (ISO8879), a perfect opportunity arose to create a document classification system which would be able to analyse automatically the logical components of documents, and to extract the information needed for book-keeping and retrieval purposes. SGML suggests a standard language to represent the structural elements of a document, using markup tags. A document prepared with SGML may then be processed by the underlying text formatter but also by any other program wishing to extract parts of it for inclusion in a database. Indeed, not only can documents be analysed, but since the layout-styles and markup tags are well-defined, it is also very easy to impose a common format on groups of documents. In principle the portability of documents marked up following the SGML standard is enhanced.

In principle, any generalised markup scheme can be used. CERNDOC also recognizes documents marked up in CERNPAPER, CERN's earlier locally developed markup system [11]. However, documents marked up in CERN's SGML implementation based on the Waterloo SCRIPT GML set [12] are preferred. The use of SGML makes the system independent of the underlying formatter: CERNDOC would work with no or very few modifications if the SGML implementation were, for example, based on TEX.

Using this SGML macro set based on Waterloo SCRIPT has the further advantage that many different output devices are supported, thus solving a large number of the multi-vendor hardware problems. In particular, it permits the creation of output files which may be viewed on simple terminals. It is important not to build a dependency on a particular formatter or macro set into a documentation system, and special care was taken to ensure this was the case.

3. Description of the CERNDOC system

3.1 Structure

The basic entities to be stored and retrieved, i.e. the documents, are CMS files. The names of the documents are therefore limited to eight characters. A cataloguing system would enable synonyming of file names to bypass this problem; this idea would not conflict with CERNDOC and may be added to a future version.

On VM/CMS there are no libraries and the file structure is non-hierarchical which makes any kind of file classification cumbersome. The usual way of grouping files together would be to place them together on one minidisk. Each virtual machine may have several minidisks, and files having something in common e.g. all documents relating to a specific working group would be placed together on one minidisk.

In CERNDOC, documents are filed in a three-level tree-like structure as shown in figure 3.

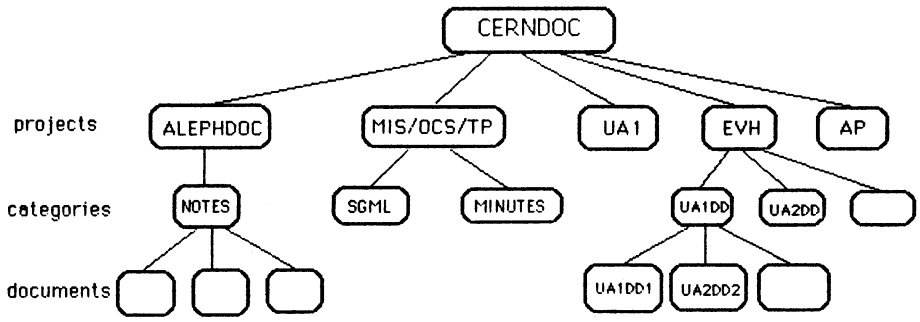


Figure 3: CERNDOC Structure

At the top level the **projects** are defined. A project is a collection of documents pertaining to some subject; two minidisks are used per project. Projects may be set up for 'private' use only, or for use by a 'group' of people.

In addition to this organisation at the minidisk level, a second level exists where **documents** are placed together in **categories**. A category may impose a common structure (reference numbers, authors, dates) on the documents. They may be registered, modified, retrieved via keywords, printed, scanned and distributed. Facilities exist for automatic reference numbering and access restriction (**confidential** documents). All files in a category reside on either of the two minidisks of the project.

More levels could be added later if the need arises.

3.2 Integration with the HELP system

The two-level structure described above permits easy integration into the CMS HELP system. When documents are registered they are both formatted for a simple screen and made into HELP files, i.e. files that may be displayed by using the HELP command. This is done by changing the second part of the file identifier (the **filetype**) to HELP followed by the first 4 characters of the category name. Files which are thus grouped together in the same category are placed in a HELPMENU (see figure 4) and these menu files are added to a menu at the project level. Menus may be traversed by pressing function keys.

Placing the documents together in HELPMENUs has implications for the category and project names. Since all files inside a project appear on the same minidisk, and CMS does not permit files with identical file identifiers, category names must differ in their first four characters. Different projects however, may have categories with the same names. The first eight characters of both the project and the category must be a valid CMS filename (and for other reasons a valid REXX variable name). This excludes for example the '/' character. To enable lists to be displayed in a reasonably orderly way on

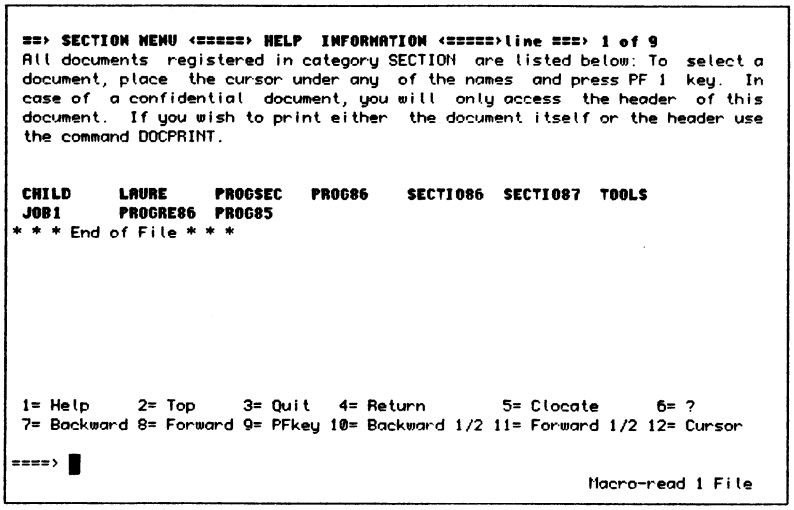


Figure 4: A HELPMENU file of all documents in a category

the screen, a limit to the category and project names of eighteen characters was set, which is enough for most cases.

At CERN, a keyword searching facility on top of HELP called FIND has been developed. It may briefly be described as follows. Every HELP file, either standard IBM or locally written, has a special part containing a list of keywords and a short explanation line associated with it (all the IBM HELP files were modified). At night, a special service machine creates a list of all the keywords in the system plus the address and owner of the minidisk where the associated HELP file may be found. Upon issuing the command **FIND keyword**, the user is presented with a list of all the HELP files on public disks which have the given keyword associated to them. Other disks will be searched and accessed if an option is given. See figure 5 for the result of the command FIND RDR (RDR is short for reader, one of the components of virtual machines, which may be compared to a MAIL box except that it resides on the spool instead of on disk). A choice may then be made by pressing a function key, and the user will be placed in the standard CMS HELP system where both locally written and IBM HELP files may be scanned.

The minimal number of disk accesses assures a very fast response. The application of FIND to documents which have been added to the HELP system and which have been given keywords is a straightforward matter.

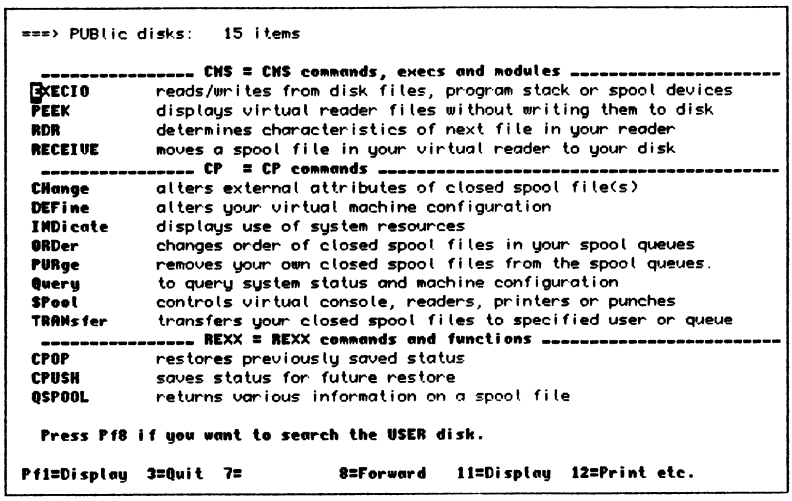


Figure 5: The result of the command FIND RDR

3.3 Security aspects

There are no restrictions on who may or may not use CERNDOC to manage their minidisks. Any user is free to start a large number of projects for private or public use by dedicating minidisks to CERNDOC. It is clear that private projects should be completely hidden from other users, while public projects will sometimes contain confidential documents which may only be seen by selected people. In all cases, write access will have to be restricted to guarantee the integrity of the documents in the database.

In VM/CMS, security is handled at the minidisk level by placing passwords on them. The fact that it is impossible to restrict access to individual files greatly increases the complexity of the security mechanism. The solution adopted by CERNDOC was to allocate two minidisks to each project. One of these minidisks will not be read-password protected and is intended for publicly accessible documents. The other is read-password protected and will contain the confidential documents. In addition, this 'secret' disk will contain passwords and access lists for the individual confidential documents.

Both minidisks are write-password protected, to ensure the documents are not modified by unauthorised people. To simplify matters, it was decided to make the three passwords (write-password of the public disk, read- and write-passwords of the secret disk) identical and to call this the project password. The read password of the public disk should be ALL. For private projects, both disks are secret disks and have read-write-passwords; one minidisk would have sufficed, but to maintain some symmetry it was decided to keep two disks. The project passwords are kept on a CERNDOC disk which has all passwords removed.

The project password will identify those users who are, in principle, authorised to read, add, modify and delete documents. An even higher level of security would be provided if the passwords were removed entirely. However, in this case everyone (including CERNDOC) except the owner is

denied access. This would imply that CERNDOC could only manage its own minidisks, and that solution would probably result in the system being used as a big pool of free disk space. A complicated accounting scheme could be necessary, and it is probably simpler to manage user minidisks directly. A better way would be to add encryption and if needed this could be added to the system at little extra cost.

At logon time, the VM Control Program (CP), configures the CERNDOC virtual machine and establishes a read/write status (a 'R/W link') for all minidisks that belong to it. As it is only safe for one machine to write to a given disk at a time, the write status of the minidisks which are dedicated to CERNDOC should immediately be disabled. This may be achieved by modifying the users PROFILE EXEC (a program which is executed automatically after logging on) to DETACH the R/W links made by CP. Another possibility would be to define read only (so called RR) links in the users directory. By using either of these two methods CERNDOC will at all times be able to safely read and write onto the minidisks which are dedicated to it.

4. The user interface

Because of the large difference in computer literacy among the users of the CERN computers, it is necessary to hide as much as possible any user unfriendly aspect of the operating system. In most cases, the necessary knowledge of VM/CMS can be restricted to the function of the special keys on the keyboard that are necessary to use the full-screen applications. For more sophisticated functions such as manipulating confidential documents or setting defaults for utilities used by CERNDOC, some additional system knowledge may be required. To modify documents one needs to know the editor (XEDIT). On the whole, the system is transparent to the user.

ALL functions of CERNDOC may be carried out from the same environment, i.e. starting from the main CERNDOC menu as shown in figure 6 one may file documents, return to the main menu, retrieve documents, return to the main menu, print documents and so on. However, the separate functions may also be carried out directly without going through the main menu.

4.1 User requirements to be able to use CERNDOC

As mentioned before, any user may start a large number of projects with CERNDOC by dedicating two minidisks per project to CERNDOC. To permit CERNDOC to write to these minidisks, passwords should be set on them and they should be linked read/only by the user as described above. The minidisks which are dedicated to CERNDOC should be accessed only via CERNDOC utilities to avoid the corruption of disk directories.

Once the minidisks have been defined a request for registration of a project is sent to CERNDOC, which will be handled automatically. Then the manager, the name of the virtual machine and the virtual addresses of the minidisks are kept in a NAMES file on one of CERNDOC's disks.

All requests, such as the registration of new projects, the definition of new categories and so on, are sent to the CERNDOC machine via interactive REXX programs using full screen panels (see for example figure 6). The registration of a new project is done using the command **DEFPROJ**.

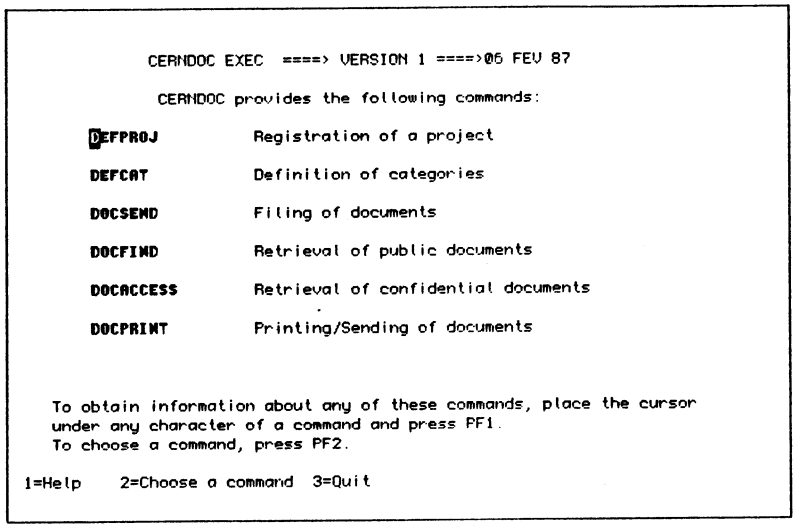


Figure 6: The main CERNDOC menu

4.2 Defining the structure of a project

Once a project has been registered, MAIL is sent to the project manager with the message that the filing of documents may commence. However, before this can be done, the attributes of (at least one) category must be defined (see figure 7) with the command **DEFCAT**. The category defines the logical components that all its documents must contain. These components are extracted and placed together in a special registration file (see figure 8).
For certain categories of documents it may be necessary to insist on a particular reference numbering scheme while for others it may be necessary to require the name of the author on the title page. Notice the field indicating the default printer for printing the document. This is what is meant by the structure of a category. Each document that is submitted to CERNDOC is analysed upon receipt. If the document does not contain the required attributes but it is marked up in a recognised way (i.e. using SGML or CERNPAPER), the relevant tags are added to the document. If it is not marked up recognisably, this information will be added to the registration file only. Many attributes that may be defined for a category of documents correspond to logical components of a document which may be described by tags in the SGML system, and maximum benefit will be derived from CERNDOC if it is used with documents that are marked up using SGML.

4.3 Storing of documents into CERNDOC

CERNDOC accepts any type of documents (e.g. binary files), but will recognise and act on certain properties of a small set of filetypes (SGML, SCRIPT and LISTING i.e. a file that contains printer carriage control characters in the first column). When a document is sent via the **DOCSEND** command to CERNDOC's reader and is received there, three processes take place (after write access to the category has been validated):

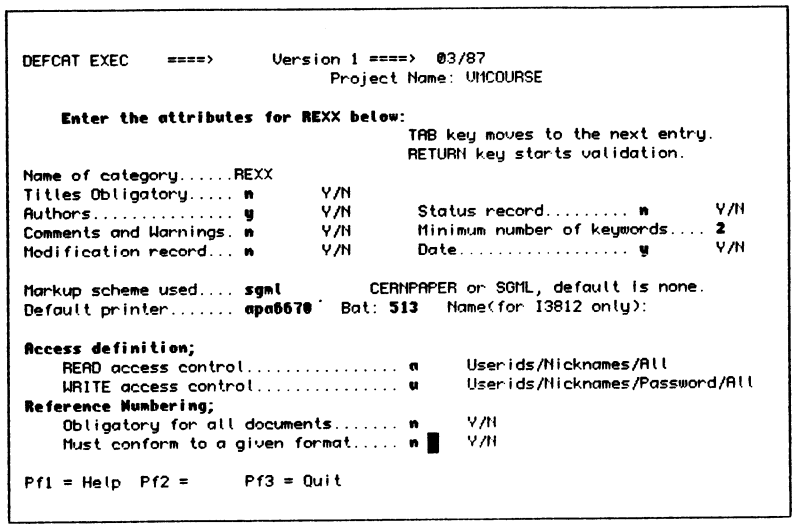


Figure 7: Defining the attributes of a category of documents

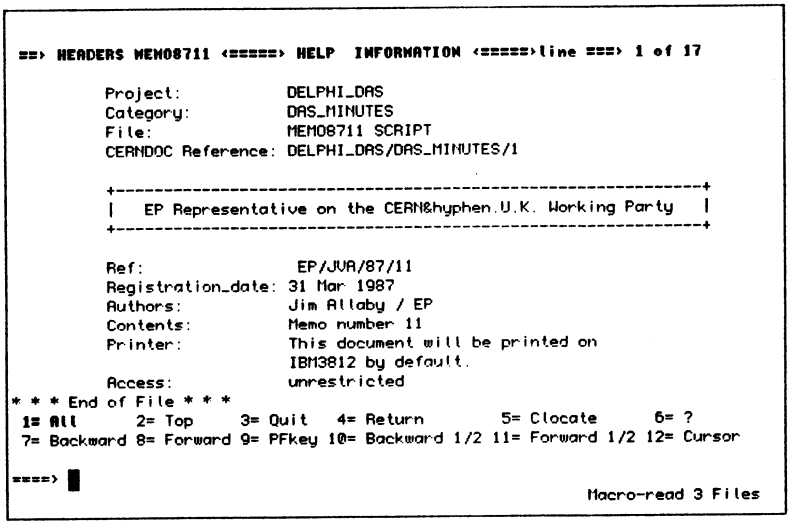


Figure 8: A typical registration information file

1. The file is analysed for the registration information required by the structure of the category. The

user is prompted to supply missing information. A registration file is created (see figure 8), which is referred to as the **header**.

2. A terminal-readable, formatted version of the document is created by possibly invoking the text formatter (see figure 2). This file is integrated with the HELP system by creating and updating the necessary MENU and INDEX files (see figure 9). When the formatting has been completed, the keyword list is analysed and added to the keyword dictionary. The keywords are obtained from the user by prompts when the document is sent to CERNDOC; the document may later be retrieved by giving any of these keywords to the command **DOCFIND**.

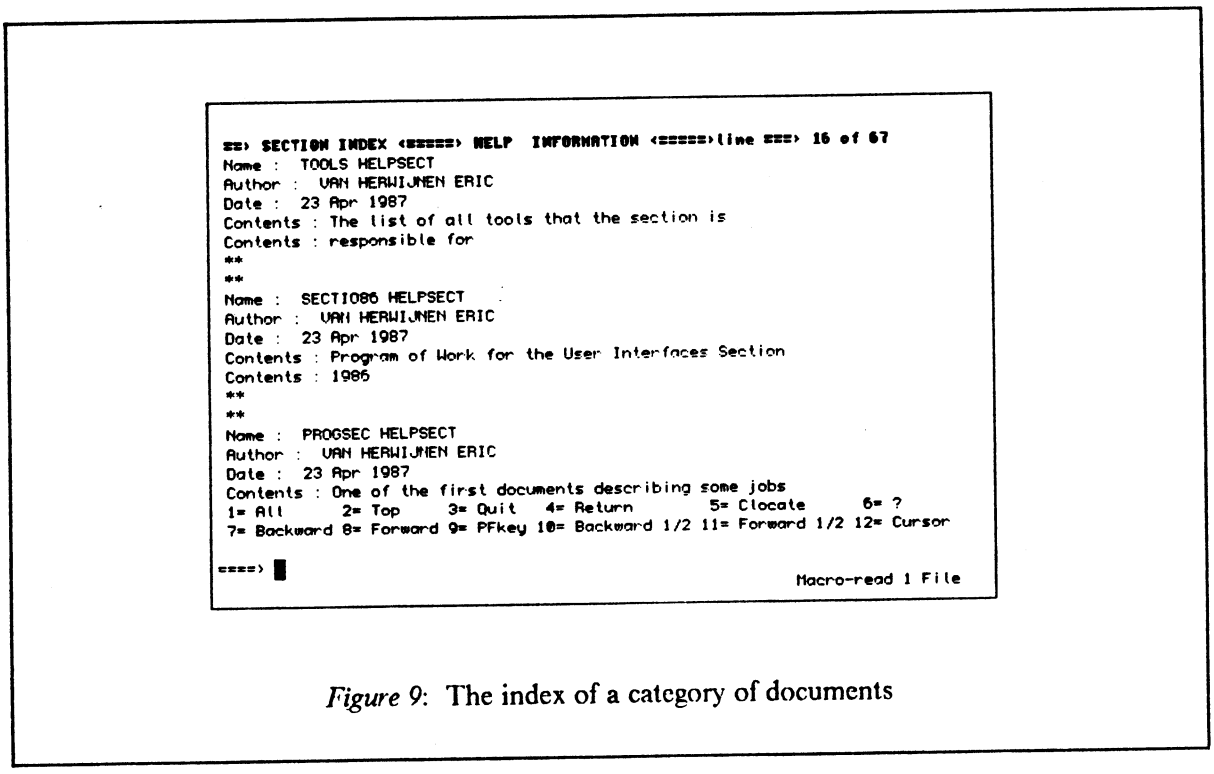


Figure 9: The index of a category of documents

3. The source file is stored, as well as bitmaps and any other files imbedded by it. READ/WRITE access to the file is defined. Various bookkeeping files containing pointers and passwords are also maintained along the way.

These three processes are handled slightly differently according to the filetype of the incoming file. SGML type files may be easily analysed and modified if necessary. CERNPAPER files (and other files which have filetype SCRIPT) are also accepted but as this system is being phased out, they are of lesser importance. Formatted files (files with filetype LISTING) and MacIntosh bitmaps (files with filetype MACPAINT) are treated specially as well, allowing the latter to be imbedded in other documents in the same project. Files that do not have a recognised filetype are considered free format documents unless they are declared to be non terminal displayable by the author. Non displayable documents are simply stored; the formatted document is replaced by its registration information.

Rather than duplicate the document itself, there may be a requirement to simply add a pointer to indicate its location. In other words CERNDOC may simply be used as a catalogue.

There is no reason why CERNDOC should not be programmed to recognise any other existing formatter, but since Waterloo SCRIPT is the only implemented text formatting system on CERN's IBM system, there has not yet been any need to do so. TEX documents for example are stored as free format documents, or they should be run through TEX first to produce a formatted LISTING file which may be stored.

DOCSEND provides many other facilities such as erasing, renaming, replacing, changing keywords, archiving and recalling documents, as well as locking documents (to avoid problems when multiple users work on a single document) and subscribing to a mailing list to receive automatically information about changes to projects.

4.4 Retrieval of documents stored in CERNDOC

4.4.1 Public documents

Once a document is stored in CERNDOC it may be retrieved by using the standard HELP menu structure (see figure 4), or by using keywords (which may include all reference numbers). The names of the file, category, project, author and date are automatically added as keywords when a document is registered. To retrieve public documents a command **DOCFIND** is provided.

If DOCFIND is given only keywords, a completely horizontal search is made through all projects and the results are displayed one screen per project. By giving the project name as an argument, the search will be restricted to just that project (see figure 10). An entry is provided into DOCPRINT, allowing easy printing once the document has been found and scanned.

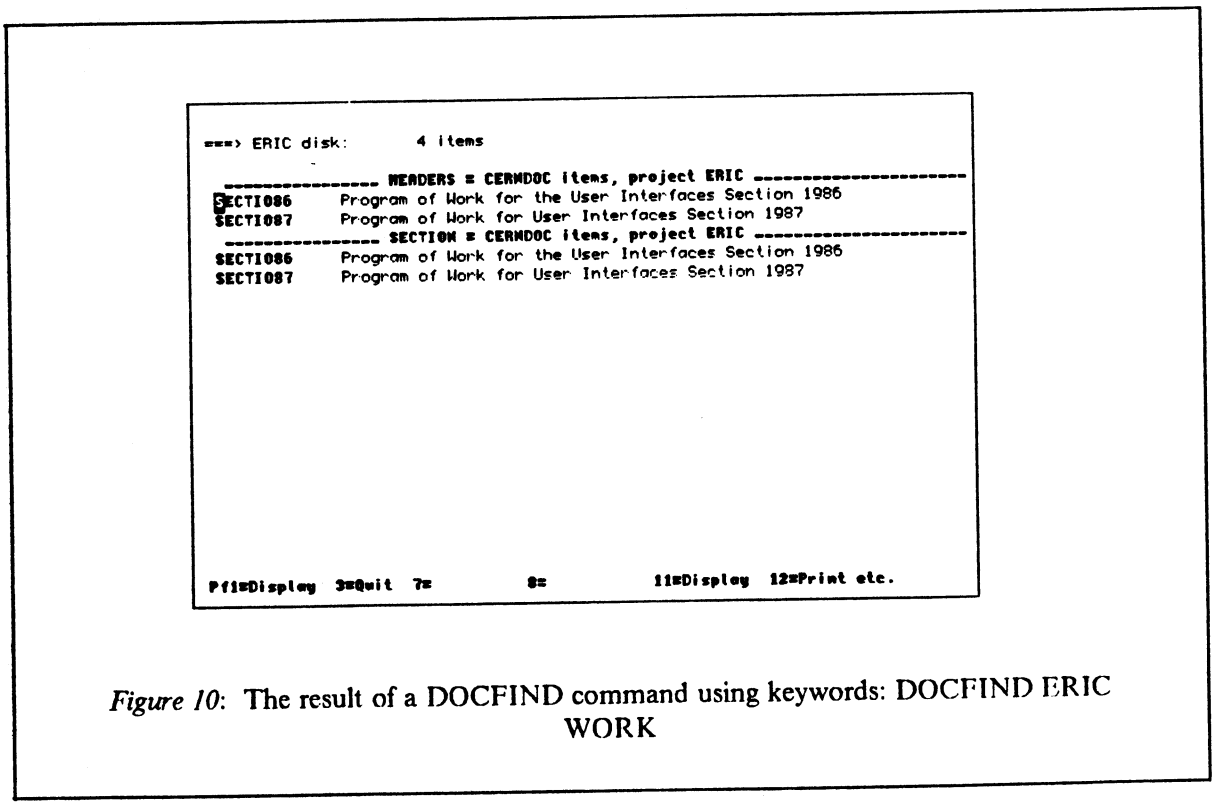


Figure 10: The result of a DOCFIND command using keywords: DOCFIND ERIC WORK

The name of each file containing the keyword is displayed, together with one line containing a short summary of the file. This has an interesting by-product. As all registration headers of a project are kept in a special category called **headers**, the command

```
DOCFIND project-name HEADERS
```

displays a list of all files in a project (confidential and public) with a one line description of their contents. This provides a useful addition to the CMS filing system allowing a display of all files on a minidisk together with a one line comment.

By giving CERNDOC the name of the project, the name of the category or the name of the document, the HELP system will be entered. It is assumed that in that case the user knows what is wanted and the file (or menu) is shown immediately. An easy way of searching through the menus is also provided.

Some options are available to automatically access the index of a category or the last document entered into a category.

4.4.2 Confidential documents

To retrieve confidential documents DOCFIND cannot be used, as this makes a LINK to a minidisk. Breaking out of the program would leave this LINK behind, allowing all confidential files to be accessed. For this reason a separate command exists, **DOCACCES** which sends files to the virtual reader. The system checks if the user has the correct access privileges, i.e. if the corresponding userid belongs to a list or by prompting for a password. Options exist to allow the file to be received onto a minidisk, to format it, to have it printed automatically on a printer of the user's choice, or to distribute it to users that have registered to the project. By default a copy of the file (formatted, source or registration information) is sent to the user's virtual reader.

Using the spool is consistent with general VM practices but it does imply that there could be a proliferation of copies of confidential documents. Moreover, the spool is a relatively unprotected area (spool files owned by different users may be transferred with certain system privileges). Probably encryption should be considered to improve the security.

4.4.3 Printing public files

To print public files the command **DOCPRINT** may be used. For printing confidential files, **DOCACCES** should be used. The source, formatted file or registration information may be printed on any printer supported by the computer centre; if this is not adequate, the user may request the file to be sent to a userid via MAIL for further processing. Within DOCPRINT, the project and category menus may easily be searched so that it is not necessary to remember the filenames.

5. Structure of the programs

In this chapter a short description is given of the main CERNDOC programs.

5.1 DOCRUN - the main program running inside CERNDOC

The CERNDOC service machine runs a program called DOCRUN (see figure 11).

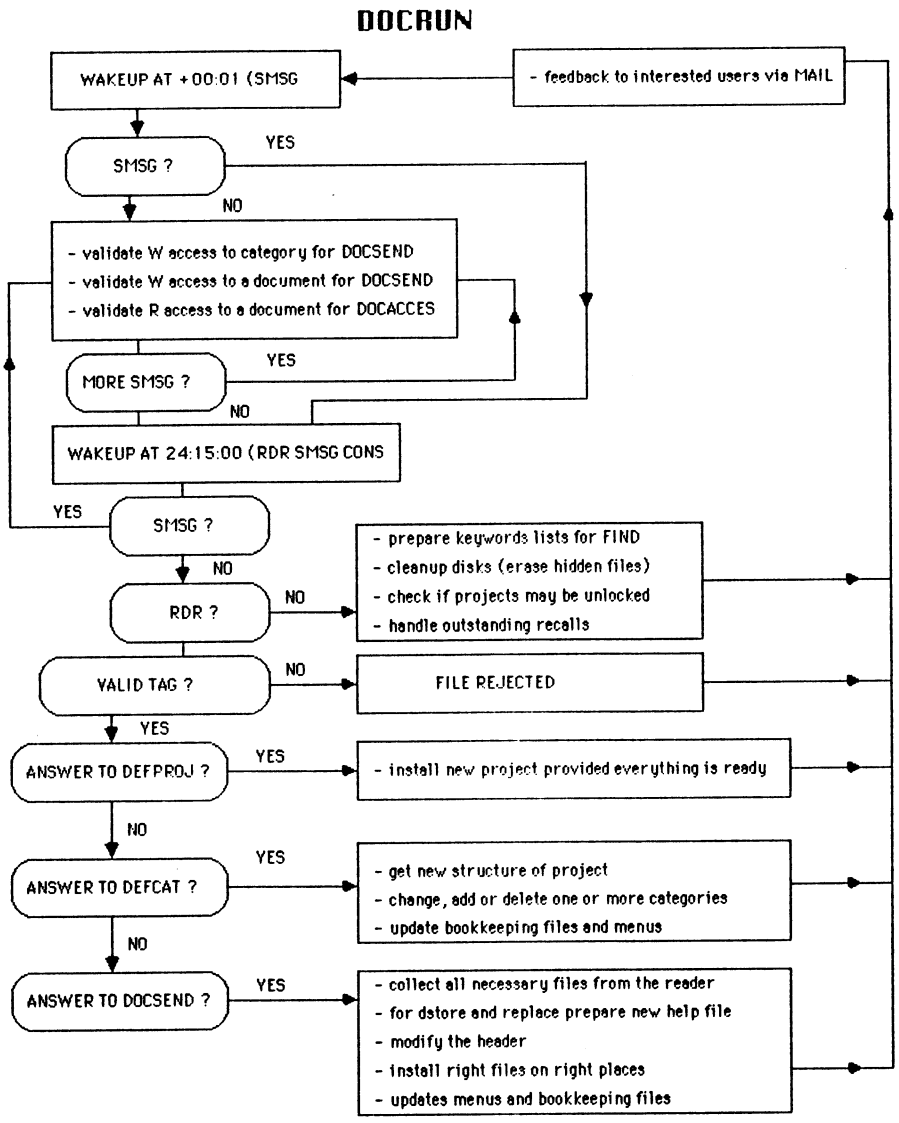


Figure 11: Structure of the DOCRUN program

The DOCRUN program places CERNDOC in a dormant state, from which it may be aroused by any SMSG (special message) or file that is sent to it. All SMSGs are treated first until there are none left. When an SMSG arrives, it will normally be sent by a user executing the DOCSEND or DOCACCES command. These utilities use this mechanism to make enquiries about read/write access authorisation. The information resides on a disk to which only CERNDOC has access. Once the authorisation has been granted, an SMSG is sent to the user (whose machine in turn will have been placed in a dormant state by DOCSEND or DOCACCES) with the necessary files. The user's machine is activated and continues the command.

Any SMSG which has not been treated after one minute is ignored.

When all the SMSGs have been processed, CERNDOC re-enters a dormant state. Again, it may be aroused by SMSGs sent to it, by files arriving in the CERNDOC reader or by the fact that it is midnight. If a file arrives in the reader, a label accompanying it called the **tag** is analysed; files with invalid tags (e.g. random mail files) are sent to the CERNDOC administrator. This tag indicates which project, category and document are concerned by the request and if it was issued after a DEFCAT or a DOCSEND command.

If a DEFCAT command was given, CERNDOC will act on the categories of the given projects. It will add, modify or delete them and update all menus when necessary.

A DOCSEND command will either mean that a new document has to be stored or that a user wants to act on an existing document. In the case where a new document is being stored, DOCRUN will prepare a terminal displayable version of the document, as well as a header where all the registration information will be kept. The header will be stored on the public minidisk whereas the formatted version and the source will be stored on the secret or public minidisk depending on whether the document is confidential or not. DOCRUN will take special care to avoid the corruption of the directory of those users which have links to the disk at the same time. It will also update the menu of the category, the keyword dictionary for this project, the index of the category and various other bookkeeping files. DOCRUN always ends the session (period of activity between two dormant states) by informing the user either that his request was successfully completed or that some problems were encountered during its process. This is done whenever possible via special messages otherwise by MAIL.

When CERNDOC wakes up at midnight, a number of various tasks are carried out:

- The list of keywords related to documents registered the day before is added to the different keywords dictionaries. All dictionaries (one per project) are updated.
- DOCRUN wil re-install the archived documents for which a RECALL request has been issued.
- All files which were left behind to avoid the corruption of disk directories are removed.
- Finally CERNDOC will unlock, whenever possible, all projects which were locked due to lack of disk space.

5.2 DEFPROJ - the command to start a new project

A new project may be registered by using the DEFPROJ command. DEFPROJ first checks if the syntax of the project name is correct (uniqueness, validity of characters and length). It continues to check if the allocated minidisks exist. If these conditions have been met, prompts are given to supply the name of the manager, the chosen assistant(s), the level of confidentiality and the name of the project password. Finally all information is sent to CERNDOC in a file; CERNDOC checks the correctness of the project password, the uniqueness of the project and so on.

5.3 DEFCAT - the command to define the attributes of a new category

The attributes of a category may be defined, changed or inspected using the DEFCAT command (see figure 12). DEFCAT first checks the status of the user. If it is an ordinary user, only displaying of category attributes will be allowed. If it is an assistant, a prompt for the project password will be issued. In addition, changing of attributes and adding categories will be permitted. A file of changes and additions is prepared and sent to CERNDOC. If it is a manager, the deletion of categories will also be allowed. Again, a file is prepared and sent to CERNDOC.

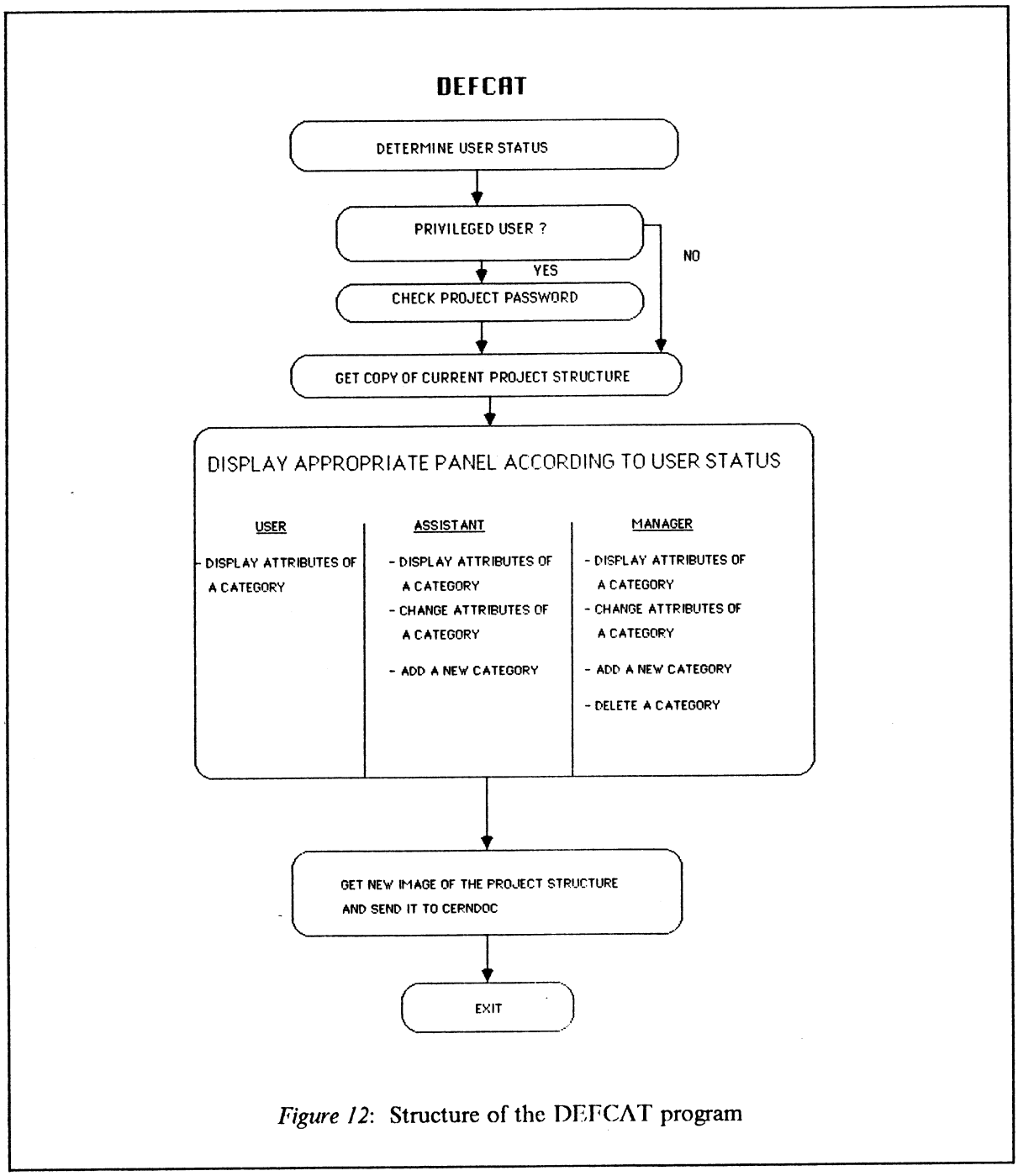


Figure 12: Structure of the DEFCAT program

5.4 DOCSEND - the command to send documents to CERNDOC

Documents may be sent to CERNDOC using the command DOCSEND (see figure 13). DOCSEND first determines the subtask to be carried out.

In the case of the STORE subcommand write access to the category is verified. The filename is checked. Prompts are given for any registration information that is required but cannot be found in the

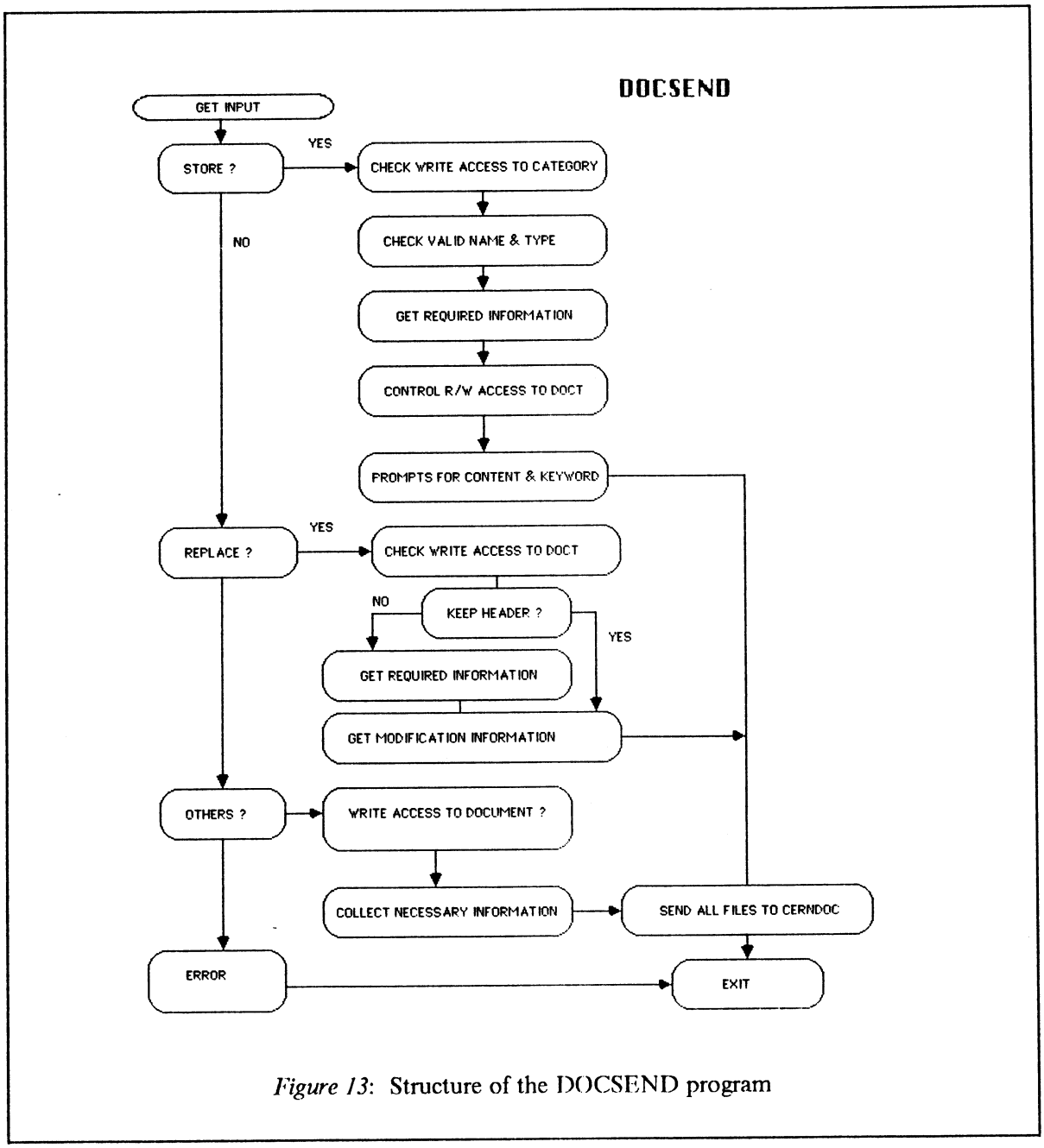


Figure 13: Structure of the DOCSEND program

document. Read/write access control of the document is defined. Prompts are issued for the contents and keywords (the filename, date, author name, category name, project name and all reference numbers associated with the document are automatically made into keywords). This information is collected in several files and these are sent to CERNDOC.

In the case of the REPLACE subcommand, write access to the document is verified. Prompts are given for modification and other information. This is collected in several files which are sent to CERNDOC.

In the case of the other subcommands, except REGISTER and UNREGISTER, write access to the documents is checked. Information is extracted from the user and sent to CERNDOC.

5.5 DOCFIND - the command to retrieve public documents

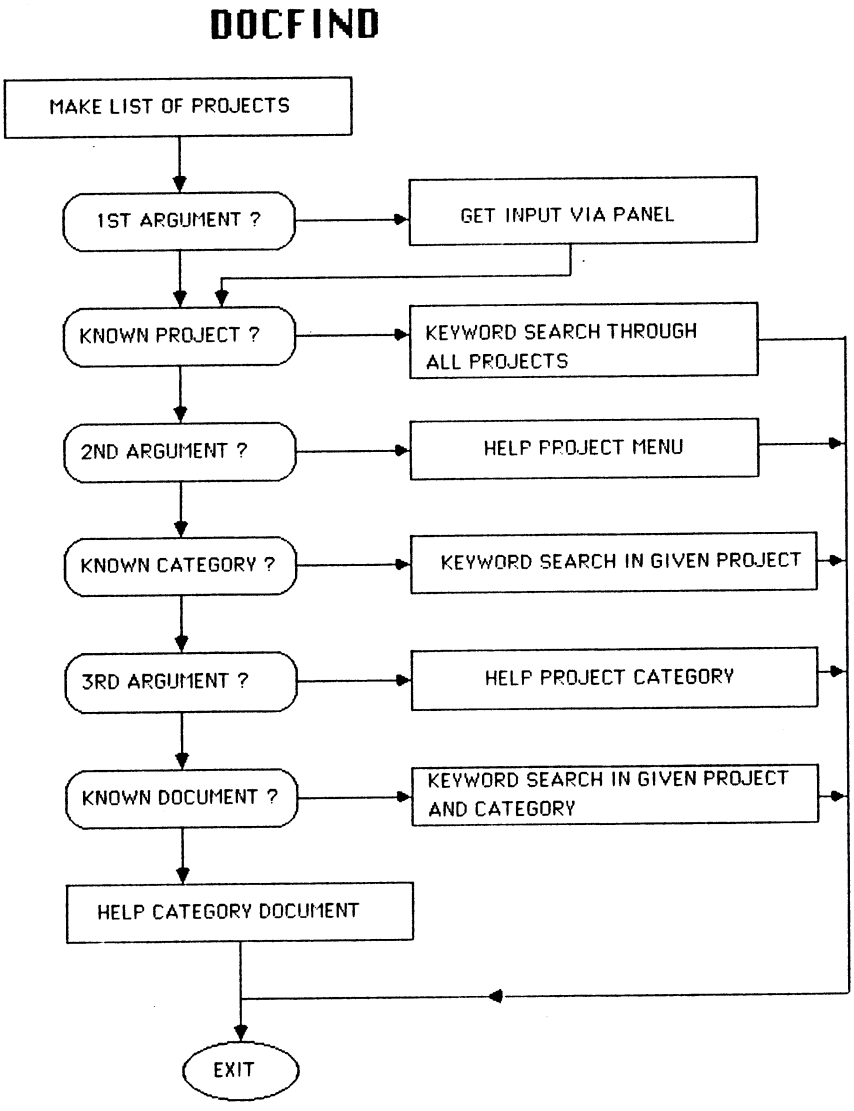


Figure 14: Structure of the DOCFIND program

Public documents may be retrieved from CERNDOC by using the DOCFIND command (see figure 14). DOCFIND first makes a list of all public projects added with any private ones owned by the

userid issuing the command. If no argument is given, prompts are issued for keywords and/or project, category and file names.

If the first argument does not equal the name of a project appearing in the list, the arguments are taken as keywords, a horizontal search through all projects for these keywords is made and the results are presented to the user.

If the first argument equals the name of a project and there are no further arguments, the menu of all categories in that project is displayed. If there are further arguments, and the second one is not equal to the name of a category, these are taken as keywords and a search through the given project for these is made.

If the second argument is the name of a category, and there are no further arguments, the menu of all files in that category is displayed. If there is a third argument and a document exists in the category with this as name, it is presented immediately. If there is no document with the third argument as filename, it is taken as a keyword and all documents in the category are searched for that keyword. Any additional arguments are always assumed to be keywords.

5.6 DOCPRINT and DOCACCESS - the commands to retrieve and print documents

These commands allow retrieval and printing of documents (see figure 15 and figure 16).
The structure of these programs is simple. Prompts are given for project name, category name and file names. The user may at any time interrogate the list of the available items, and in the case of DOCPRINT, look at a document before deciding to print it. Finally, for DOCACCES, access is checked and if validated, the document is printed, sent or displayed.

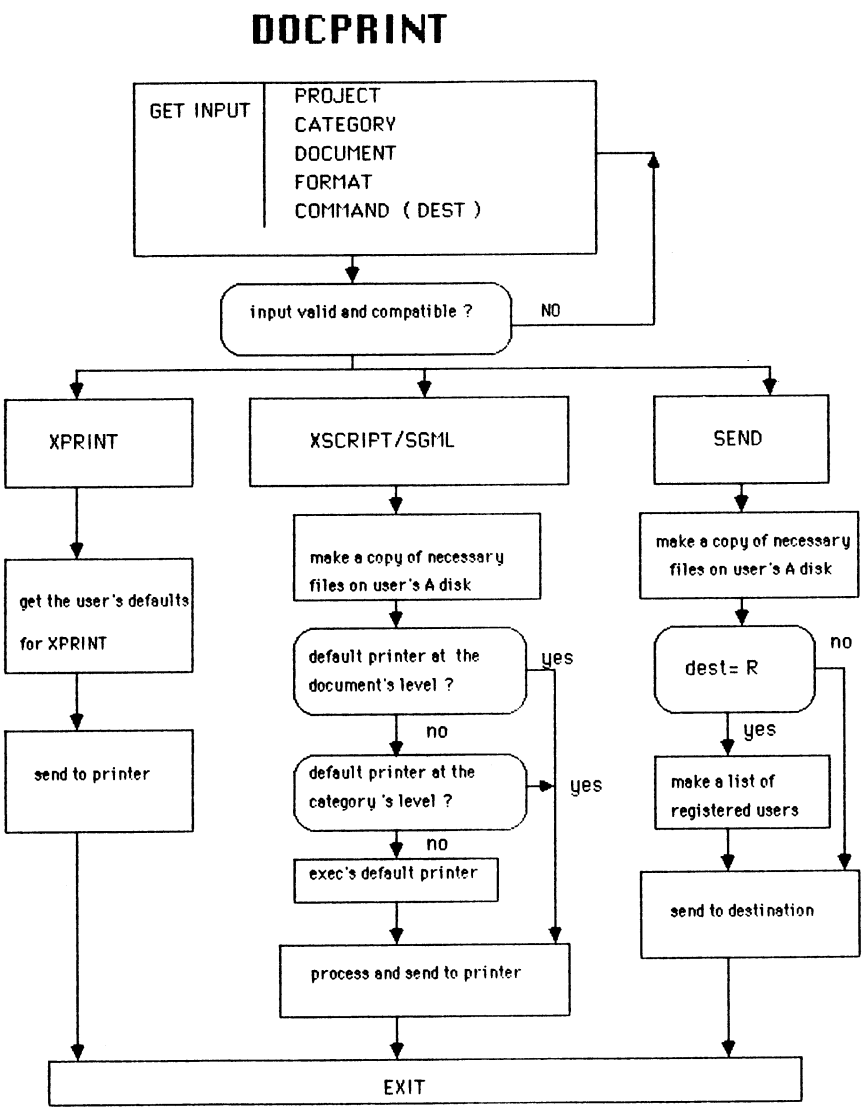


Figure 15: Structure of the DOCPRINT program

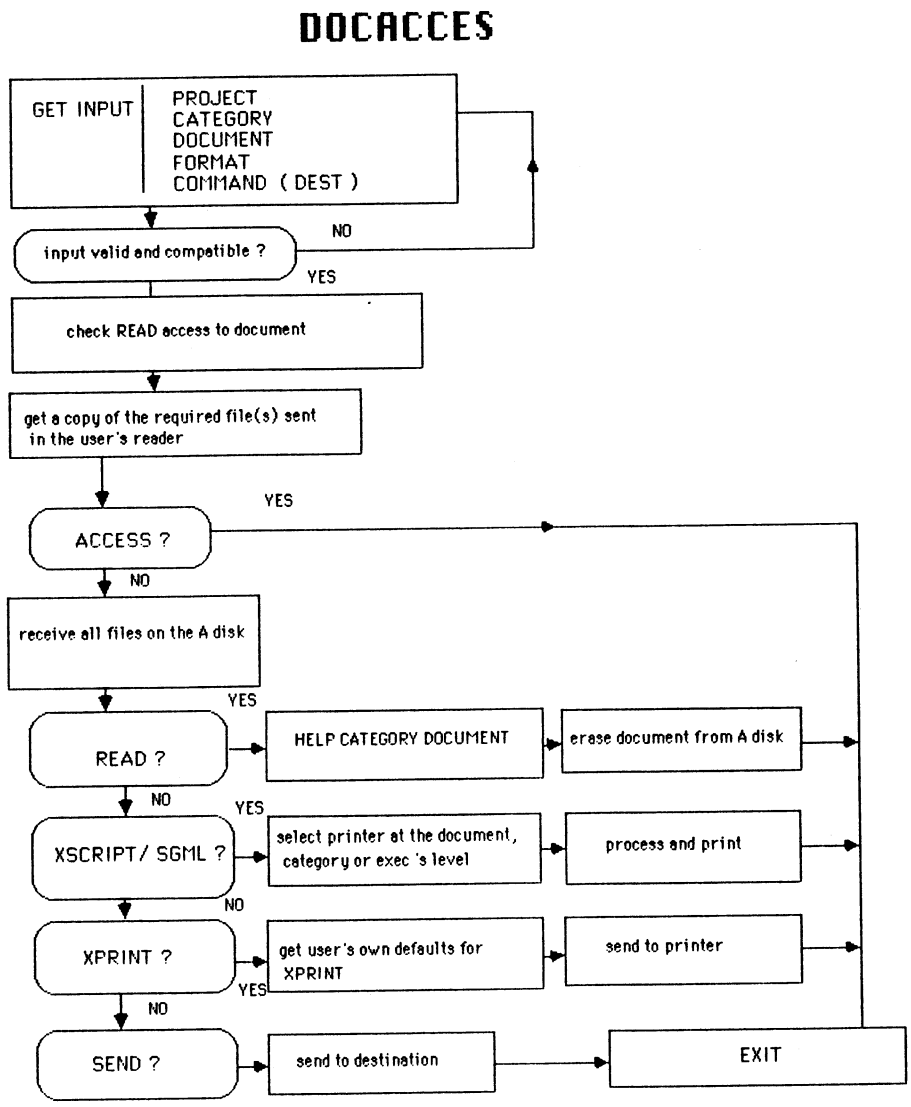


Figure 16: Structure of the DOCACCES program

6. Network access

In the first implementation, a facility was supplied that allowed users to obtain documents via EARN/BITNET. Task-to-task communication [14] could make interactive network access a reality in the future, but for the moment interactive facilities are out of the question. However, being informed of changes to a project by MAIL, together with this facility, proved sufficient for users at least to obtain documents remotely.

To achieve this, CERNDOC was programmed to analyse and reply to files sent to it via MAIL, NOTE or SENDFILE. The MAIL, NOTE or file sent to CERNDOC should contain only a one-line DOCFIND or DOCACCES command, i.e.:

```
DOCFIND project-name category-name document-name ( options
```

for public documents, or

```
DOCACCES project-name category-name document-name ( options
```

for confidential documents. Supplying passwords via MAIL is generally considered not to be prudent and access to certain documents may therefore be refused.

In this way, a minimal document retrieval facility exists for remote users. Plans exist for DCF command files to store documents from the central CERN VAX.

7. Experience and conclusions

From the implementors point of view, many problems caused by the CMS filing structure had to be overcome.

Dedicating minidisks to CERNDOC is a messy procedure which immediately exposes many user-unfriendly aspects of VM (e.g. obtaining minidisks, use of DIRMAINT, understanding LINKS) to (in many cases) infrequent VM users. A special command may be created at CERN to facilitate this procedure.

Updating shared minidisks is inconvenient and must be done with great care. Book keeping files need to be kept and cleaned up by the service machine to assure no one ends up with corrupted disk directories.

Use of the spool files is regarded as difficult by most inexperienced users. The DOCACCESS command had to be written to hide, as much as possible, the transfer of confidential files via the SPOOL.

The integration into HELP and the local FIND system did not pose any technical difficulties. However, the necessity to know the filenames for editing and replacing documents proved to be rather awkward for those used to other operating systems. Being able to modify documents simply by giving, for example, the reference number was a frequently received request.

The default security mechanism in VM is too rigid: either there is none, or there is too much. Lists of authorised people need to be maintained. To avoid updating problems CERNDOC keeps pointers to these lists on the user's minidisks. Had the IBM Help system been incorporated into a relational database most of these issues would not have arisen.

From the users point of view, once these problems had been solved, the possibility to use CERNDOC as a private disk manager (for all sorts of files, not only documents) and catalog has proved a particularly useful enhancement to the cumbersome CMS file structure. Most users found retrieving files stored by CERNDOC easy.

The majority of user criticisms reflect the rigidity of CMS as outlined above. The use of a service machine to share files makes updating a rather indirect process. Much flexibility would be gained if, for example, one could Xedit the headers directly. Such a simplification is being studied.

Strong advantages were seen by users using the news facilities and for those storing SGML documents. Most of the registration information required for these documents is presented on the front page, and a common appearance is guaranteed for documents in the same category. Some SGML developments were necessary to allow a better representation of for example reference numbers.

Users from the VAX community have expressed a wish for a similar system running on VMS. Whilst on the one hand the file structure of VMS would facilitate the design, there are some parts of the REXX implementation that would probably be difficult to transport. The ease of debugging a dynamic prototype in REXX was a considerable advantage. Rewriting CERNDOC in a high level language once would be an interesting exercise in the portability of large software systems from VM to VMS, something that could be of benefit to the CERN users who frequently encounter these problems.

However, commercial communications products are becoming available which will enable the inclusion of heterogeneous server nodes into a consistent network environment. For example the task-to-task communication provided by Interlink would enable CERNDOC (running on VM) to be used interactively from VMS on the central VAX at CERN.

Acknowledgments

We wish to thank all our colleagues in the DD Division at CERN who have helped us with discussions, suggestions and proofreading as well as a considerable amount of patience during the development phase of CERNDOC.

References

1. R. R. Ragan, 'CONTEXT: An On-line Documentation System', *Software – Practice and Experience*, **16(3)**, 217-224 (1986).

2. I. Sommerville, R. Welland, I. Bennett and R. Thomson, 'SOFTLIB-A documentation Management System', *Software – Practice and Experience*, **16(2)**, 131-143 (1986).

3. T. Rosato,'The CMS File System: A Historical Perspective', *SHARE 67 Session O705*, August 1986.

4. DEC, 'Guide for VAX/VMS System Security', *VAX/VMS version 4.2 manual, order number AA-Y510A-TE*, July 1985.

5. Ph. Charpentier, I. Dahl-Jensen, M. Metcalf, C.M. Story, 'Proposal for the DELPHI Software Documentation', *Internal DELPHI Note 85-90 PROG 32*, November 1985.

6. C. Piney, 'How to use D O C x: The Document Organisation and Control Procedure', V9, *CERN Internal Document DOCP/G/G7*, Aug. 1985.

7. S. Robinson, 'The Design and Operation of the SYSDOC System Documentation Facility', *Rutherford Appleton Laboratory Internal Note CCTN.46*, December 1982.

8. Details about the HEPVM collaboration and the HEPVM tape may be obtained from S. Jarp, DD Division, CERN, Geneva, Switzerland.

9. Display Input/Output Facility Version 2. IBM Program Offering SB11-5820-0.

10. R. Schafer, 'MAIL/MAILBOOK version 86.359', Rice University, 1986. At CERN, the MAIL EXEC is used with the MIT MAILER.

11. E. van Herwijnen, 'CERNPAPER Users Guide', *CERN Internal US Note DD/US/50*, May 1985.

12. A. Berglund, 'CERN SGML Users Guide', *CERN Internal Note*, October 1986.

13. L. Esteveny, E. van Herwijnen, 'CERNDOC - The DD Documentation Scheme', *CERN Internal US Note DD/US/103*, May 1987.

14. Interlink Computer Sciences Inc., 'IBMvm/DECnet Gateway Task-to-task Communication Programmer's Guide', *Publication No. UG03TSK2.1*, May 1987.