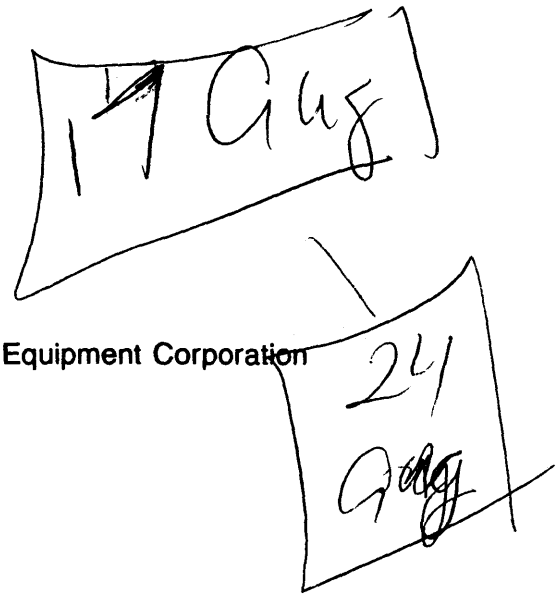
Going On Line and On CD-ROM with Software Manuals at DIGITAL

This paper describes projects under development at DIGITAL to fully automate the documentation production process (the VAX DOCUMENT product) and to extend VAX DOCUMENT's capabilities to support on line documentation; that is, to provide user manuals that are intended to be displayed and *read* on the workstation screen, not on paper.

Mary H. Utt
Documentation Specialist
Corporate User Publications Engineering

Digital Equipment Corporation
110 Spit Brook Road ZK01-2/C21
Nashua, NH 03062-2698

7-June-1988



Digital Equipment Corporation

1 INTRODUCTION

Most of the software documentation at DIGITAL (and some of the hardware documentation) is produced under the Corporate User Publications umbrella. Over 550 technical writers, editors, and production personnel (text formatting specialists, proofreaders, graphic artists, etc.) in three documentation cost centers in Massachusetts, New Hampshire, and England report up to one manager.

An engineering group of 16 people, charged with providing tools to support the documentation groups, also reports to this manager. This group has almost fully automated the documentation process with the development of VAX DOCUMENT, used internally for the past five years to produce manuals for many DIGITAL products (including the VMS operating system) and released as an external product in August 1987.

Corporate User Publications and VMS Engineering are currently working on a joint project to extend VAX DOCUMENT and to put user manuals on line; that is, to provide user documentation that is intended to be accessed, displayed, and *read* on the workstation screen, rather than printed on paper.

This paper describes the VAX DOCUMENT production system in general and its use in producing on line documentation in particular.

2 THE VAX DOCUMENT SYSTEM

VAX DOCUMENT is a batch-oriented document formatting and production system that is designed for large writing and production groups, where documents:

- are produced and maintained by multiple writers
- are part of a set of documents and contain many inter-document relationships
- have a long life cycle
- are frequently revised or updated
- are long (100+ pages) and complex

The following sections provide an overview of VAX DOCUMENT features.

Generic Markup Language

One of the primary features of VAX DOCUMENT is a generic mark-up language. That is, source files are format-free and device-independent — writers are freed to concentrate on the *content* of their documents, rather than the format.

Writers "tag" the structural elements of their text, rather than specifying formatting commands. For example, <P> tags a paragraph, <CHAPTER> tags a chapter, <TABLE> tags a table, <HEAD1> tags a first-level heading, and so forth. Tags are also used to construct complex tables, to set mathematical formulas, to merge computer-generated graphics with text, and to specify index entries.

A feature unique to VAX DOCUMENT is the ability to create reference templates, using the template tags. This ensures consistency in documenting reference information, even in very large reference manuals to which several writers contribute.

There are also conditional tags which allow maintaining text variants in the same source file. By processing the file for one condition or the other, different outputs can be produced to meet different documentation purposes.

Tags are expanded into appropriate formatting commands — instructions that control type sizes, margins and indentions, tabular material, pagination, and so forth — and then output to a printing device.

Multiple and Customizable Designs

Formatting information is provided in files separate from the documentation source file. These files are called in when the source file is processed. Therefore, the same source file can be used to produce different styles of output. VAX DOCUMENT supports format designs for manuals (reference and user), memos, reports, articles (one and two columns), and overheads

In addition, many of the formatting parameters for these designs are user customizable, so new designs can be developed to meet new requirements. Formatting parameters include page dimensions, font specifications for text elements, vertical spacing, and positioning of text elements.

Sophisticated Page Composition Software

VAX DOCUMENT's page composition features include automatic pagination (avoiding widows and orphans) with page numbers and running heads and feet; hyphenation and justification; automatic numbering of logical text elements (chapters, sections, figures, ec.); multipage tables and figures (with continuation heads and captions); and footnotes.

Complete Bookbuilding Tools

VAX DOCUMENT provides automatic indexing, table of contents generation, and cross referencing. The indexer collects, sorts, formats, and outputs a finished index from the index entries tagged throughout the source file; and the table of contents is generated automatically from the headers in the source file. The <REFERENCE> tag, which takes a symbolic name as an argument, is used to automatically resolve cross references from one section to another in text and to figures, tables, and examples. Thus, writers need not keep track of changes to numeric sequencing of sections and to page numbering as a book is written or revised.

VAX DOCUMENT makes it possible to build an entire book, including table of contents, index, and accurate cross references and numeric sequencing of headers, figures, tables, etc. The index entries for individual manuals can also be amassed into a master index for an entire documentation set.

High Quality Output

VAX DOCUMENT produces typeset output on laser printers, including Postscript® devices. It also supports output on line printers and ASCII terminals.

With on line documentation, VAX DOCUMENT will add VAXstation monitors to the list of output devices.

® Postscript® is a registered trademark of Adobe Systems Incorporated.

2.1 WHY GO ON LINE?

With all of VAX DOCUMENT's capabilities for fully automated, high-quality print output, why do we want to put documents on line? Some statistics about the VMS V5.0 documentation set (over 20,000 pages) answer this question:

- The vinyl used for the binders would cover 102 football fields (4,590,000 square feet).
- Laid end to end, the binders would extend 185 miles.
- The binders alone would fill 170 trailers; the paper — 6.6 million pounds — would fll 83 trailers. The trailers for binders and paper would stretch 4.8 miles.
- The 6.6 million pounds of paper requires 6,600 cords of wood. This is enough wood to heat the average home for 825 years or a small town (population 2,500) for one year.

In short, we are leveling forests and depleting fossil fuels to print manuals that are out of date in six to nine months.

2.2 Drawbacks of Paper

In addition to these statistics, there are many reasons to deliver documentation on line:

- Time and cost of paper production — as products proliferate and become more complex (usually requiring more pages of documentation) and as release cycles shorten to meet customer demand, the print production process takes a disproportionate amount of time. Even highly automated hardcopy systems still must go through a lengthy and costly printing process — books must go to print before software kits are built in order to make both the software and documentation available on the same date.
- Accessibility of information — many shops buy only one or a few manual sets and keep them in a central repository (sometimes under lock and key). Thus, the information — even if it is complete, accurate, up-to-date, and elegantly typeset — often does not get used because it is difficult to access.
- Paper takes up a lot of space — the VMS operating system manual set is about twelve linear feet. A complete set of documents can take up more space than a CPU, bit-mapped monitor, and CD reader.
- Waste — most users do not need an entire documentation set and may discard out-of-date manuals still in their protective shrinkwrap. For example, a user of a VAXstation in a low-end cluster probably has little need for system management documentation; conversely, the system manager of that cluster probably has little need for run-time library documentation.
- Monitors are catching up with paper as a display medium — while the resolution of bit-mapped monitors (75-100 dots per inch) has a way to go to match laser printers (300 dots per inch) and typesetters (1200 dots per inch), they do permit the use of proportionally-spaced type fonts, the display of complex graphic images, and they do not limit the display to 24 or so lines on the screen.

2.3 Advantages of On line

Delivering documentation on line can directly address most of the drawbacks of paper (although it will be much harder to take a manual home to peruse in the bath). The savings in space (both in the warehouse and in users' offices) , paper, vinyl, shrinkwrap, and shipping will be enormous. Users' will no longer have to unpack boxes in boxes of documents, or insert update pages into their manual sets. Customers with only one or a few documents will not have to worry about manuals or pages "walking" away with users.

Although documentation production procedures must be dramatically altered to do so, it should be no more difficult and time consuming to build a documentation set than it is to build the software kit. This lengthens the time writers have to write and correct manuals (release notes may become obsolete!).

Information is available to all users and can be accessed faster than it can be using a book if all you need do is point and click on an index entry with a mouse to display a page, rather than flipping through pages in a book. It should also be possible to partition the information so that, for example, nonprivileged users get only the information that is useful to them and system managers get privileged system information. This not only provides protection of the system, it means that users need not deal with the "clutter" of irrelevant (to them) documentation.

At DIGITAL, we already have a "database" of generically encoded documents. We do not have to retype information to put it in electronic form, nor do we have to add information or processing steps to capture structural information — the generic encoding using the structural tag set provides this.

2.4 Why Go CD-ROM?

The scale on which we produce documentation dictates CD-ROM. Compact discs are literally the only medium that can contain the entire VMS operating system documentation set. (The DOCUMENT engineering group uses the VMS operating system documentation as its laboratory: if we can do it for VMS, we can do it for anyone.) While the VMS documentation set is not our initial goal for a first release, it does represent a worst case that must be planned for.

In any case, compact disc is an extremely attractive distribution medium. It can hold vast quantities of information (even the VMS operating system documentation would only take up one-third of one disc). It is cheap to manufacture and once production is set up, it will be faster than the printing process. It is durable, cheap, easy to distribute, and takes up little space. Compact discs and CD players are already familiar consumer devices so CD-ROM drives should be readily accepted; indeed, the PC world has already made CD-ROM familiar to all computer users.

Compact discs hold all kinds of data: not only can text and graphics be distributed on compact discs, but software as well. One disc can hold many application programs and their associated documentation, accessible with the purchase of a license that entitles the user to access the information. Subscription services could provide regular updates to both the software and the documentation.

In addition, instead of issuing update pages to documents (which frequently never find their way into the binders), documentation providers can issue update compact discs, which contain the entire manuals including the updated information.

3 USING VAX DOCUMENT TO AUTHOR ON LINE DOCUMENTATION

The use of a generic markup language allows us to capture high-level information about document structure to structure the database. Because we will initially produce both on line and printed documentation from the same source text files, we must work within the existing tag set used in the documents. However, the technology of a completely automated text processing system has already driven changes in documentation writing that lend themselves to on line presentation.

3.1 Modularity

If modularity is good for software, it's also good for documentation. Succinct modules of text can make needed information easier and faster to find, and can help eliminate redundancy in manuals. It's also axiomatic that well-defined text modules, or chunks of information, will form the basis of non linear, hypertext documents.

One of the important features of VAX DOCUMENT is the ease with which writing groups can develop templates for highly structured information. For example, the VMS documentation group has developed a strictly defined template for DCL command descriptions which defines the way in which the commands are described: command name, syntax, parameters, qualifiers, textual description, examples. Each of these segments of the template is tagged, for example, <SYNTAX>. Coupled with writing guidelines, a large number of writers can contribute to the *VMS DCL Dictionary* with assurance that their descriptions are complete and consistent. Templated information now makes up about 80% of the VMS operating system documentation set, and a fair amount of top-down planning goes into a documentation set at the outset of each project to manage this modularity.

Modular information is the easiest to put on line. A module of templated information becomes *de facto* a on line topic. The entry points from the table of contents and index are clear: a command name, a routine name, and so forth. If the length of the templates is unweildy for on line display, the tagged segment heads can be captured to generate pull-down menus: readers can select only examples to view if they don't need to wade through the qualifier descriptions.

Because an on line "page" (window) is typically smaller than an 8½ by 11-inch printed page, we would have problems fitting some tables, examples, and figures inline with the text on the screen. However, all of these text elements are tagged and can be extracted to be presented as separate modules on the screen. Each table, for example, is called out in the source file using a <REFERENCE> tag. This tag will be translated as a "hotspot" on the screen; clicking on the hotspot with the mouse will bring up the table in a separate window, sized for the table.

Thus, some modularity of the database is achieved in the writing process; more modularity can be achieved where necessary by capturing the tagged, structural information and translating it appropriately for the on line presentation.

3.2 Indexing

Keyword search capabilities are important; however, the indexing information that the writers provide for the manuals can also be used for keyword searches. This information is also tagged and maintained inline in the source text. For example:

```
...
<P>
Here is a paragraph about indexing templated information...
<X>(Indexing<SUBENTRY>templates)
<Y>(Indexing<SUBENTRY>see also Master indexing)
...
```

The <X> tag generates index entries with page numbers; the <Y> tag generates cross references (synonyms) in the index. We ignore the page numbers for on line display but use them to generate the links (hotspots) from the on line index entry to the page.

Obviously one important requirement is that writers produce complete, accurate, meaningful, and useful index entries: if they don't index their material well, they may as well not write it because it won't be accessible. We have provided good tools for doing this and there are a number of workshops underway to help the writers use the tools well; however, one of the most effective ways of improving indexes is to put the manual on line and let the writers themselves try to access the information, sometimes coming to the conclusion, "This is a terrible index entry!"

3.3 Cross References

The <REFERENCE> tag mentioned previously as the mechanism for displaying tables, figures, and examples, is also used for text cross references. For printed documentation, this tag (whose argument is a symbolic name for a section, figure, table, or example) is translated into the appropriate text, for example, "See Section 3.4" or "See Chapter 2" or "See Figure 6-7." For on line documentation, the tagged information is translated to a hotspot; clicking on it produces a window containing the referenced page. (Handling figures, tables, examples, and text cross references in the same way not only makes conceptual sense but it greatly simplifies both the implementation and the user interface.)

Cross referencing is a very powerful way of traversing information outside of the heirarchical structure of the documents and is the first step toward a true hypertext system. Again, we are relying on writers to do a good job of providing cross references in the source files.

3.4 Graphics

In the past, on line documentation applications have been severely restricted because only the most simple forms of graphics could be displayed on character cell terminal. With workstations and bitmapped terminals, much more complex and detailed graphics can be displayed on the screen, along with proportionally spaced fonts and other graphic devices, such as color.

One of the new, in-house tools that the VAX DOCUMENT group is providing is a graphics editor, which will completely automate the documentation production process for both hardcopy and on line books. The editor is a tool designed for both writers and artists. A writer initially "drafts" a piece of artwork, incorporating it in early drafts of his or her document. When the draft artwork has stabilized after reviews and revisions, the art file goes to

a graphics artist who produces the finished, production-quality artwork that appears in the final document.

Again, the same source files will be used to generate artwork for typeset output and for display on the workstation monitor: writers and artists process a graphics "meta-file" for the resolution and characteristics of the different output devices.

For on line documentation, graphics — figures, tables, and examples — will not appear inline, as they would in manuals. The references to the graphic elements will be *hot* — that is, when a user points and clicks on a reference, the graphic will *pop up* in a separate window. Thus, the graphic can be kept on the screen for as long as the user reads about it or wants it visible for referral. The pop-up window can also be sized for the graphic; vertical and horizontal scroll bars can be provided for navigating very large graphics (larger than the screen).

4 PRODUCING AND DISTRIBUTING ON LINE DOCUMENTATION

Probably the biggest initial change in the documentation process will be in producing and distributing information on compact disc. As documentation people, we are very familiar with the printing process (generating "repro," checking "saltprints," dealing with "author's alterations," dealing with printers' lead times, and so forth). We are less familiar with the kit-building process that leads to putting the product on magnetic media; we are not familiar at all with the process of putting things on compact disc.

4.1 Automating Documentation

The first requirement for compact disc distribution is that the documentation production process is completely automated; there should be no manual intervention from source file to premaster tape to compact disc. One of the big requirements for document producers, therefore, is the production of compound documents, documents that contain graphics files processed along with the text files: there is no paste-up on compact disc!

While we have these capabilities today, putting it all together into a smooth process will be a challenge. Currently, manuals are built one at a time, and there are points at which it is possible to manually intervene and make last-minute corrections and changes, and although this practice is discouraged, it happens (it's not a good idea to alter an interim TEX[1] file that's in final production because the alteration may never make it back into the source file, but knowledgable users do it regularly). Writers are also adept at finding ways to circumvent VAX DOCUMENT's standardized formatting and forcing some special formatting. Where the same source file is used to generate both printed and on line manuals, which have very different format characteristics, this practice can produce very ugly formatting in one or the other case. So a great deal more rigor will be required of the production process to produce both on line and printed documents.

We are not only going to need to manage two processing streams, we must address the need for "docset builds," akin to system builds. If we are to resolve intrabook cross references, the entire set must be built at once. This means that everything must be ready and correct at the same time, and in time to build the set. TEX is highly compute intensive, and we've built a number of capabilities on top of it. It takes over 2 hours to build the *VMS DCL Dictionary*, which contains no graphics, on a VAX 785 with no other processes running. Bigger processors are available but docset builds will clearly be a great deal to manage. We must also bear in

[1] VAX DOCUMENT uses TEX to format text.

mind that what we pre-master is what gets put on the compact disc: there will be no author's alterations past the point of the build.

4.2 Shipping Discs *vs.* Shipping Paper

However, shipping discs as opposed to shipping paper offers a number of important advantages in three primary areas, all of which are obvious when one considers the difference between a disc and a documentation set (**any** documentation set, but particularly a 22,000-page set):

1. Media production — compact discs can be stamped by the hundreds, each one representing thousands of pages of documentation. Thus, media production for compact disc distribution is much faster than printing paper.
2. Packaging — a great deal of time and effort currently goes into planning how to package a docset: the number and size of ring binders needed, what color they are, what is printed on the paper inserts; tabs to separate manuals and sections within binders, the number and sizes of boxes to ship the manuals ("cardboard engineering").

   All of this must be planned from the outset of the project and involves numerous people, including documentation managers, product managers, Software Distribution Center planners, writers, and editors. Woe to the writer who exceeds his projected page count (because the software functionality changed) and needs a larger binder at the last minute! Packaging also includes the labor required to pack books into binders and binders into boxes.

   Clearly a lot of these costly, complex efforts become moot if documents are shipped on compact discs.
3. Warehousing — After packaging, docsets must be warehoused until they are shipped. Warehouse space is expensive, and the size of a compact disc is miniscule compared to thousands of pages of books, their binders, and their boxes. If too many docsets have been printed, they not only must be warehoused until obsolete, they must be disposed of. Storing discs will cost a small fraction of storing manuals.
4. Shipping — Docsets are very expensive to ship and can be damaged in shipping (we hear complaints that ring binders are damaged and don't work properly after shipping). Compact discs are small and durable and can be sent much more easily, more cheaply, and faster to customers.

Compact disc distribution of documents thus represents not only a cost savings for suppliers like DIGITAL, but better quality products and faster service for the customer. Discs also take up less space in customer's offices and are easier to handle than manuals, so there are benefits all around.

5 DISPLAYING THE INFORMATION ON LINE

As already mentioned, the resolution and display characteristics of bitmapped monitors allow much more complex and sophisticated presentation of text and graphics than character-cell terminals do. But reading text from a monitor is different from reading text from a page, and we were certain that we could not format text for the screen the same way we could for paper. So we undertook readability research to investigate the type fonts available for the VAXstation and determine which was best for reading text on the monitor.

Working with DIGITAL's Software Human Engineering group and a professional video font designer, we ran a series of studies designed to narrow the range of type fonts available on the 75- and 100-dot-per-inch VAXstation monitors. Briefly, the typeface designer looked at the full set of fonts and narrowed them to 4 that had characteristics suited for screen legibility. We then asked study participants to read an article on the screen. The article was divided into four parts, each part set in one of the four fonts. Their reading speeds were measured and they were interviewed as to their reactions to each of the fonts. Reading speeds did not vary significantly, but reactions did: users have very strong feelings about fonts! As a result of these studies, we are using 14-point New Century Schoolbook as the basic text font.

Early in the project we decided to base our on line documentation on the book model. This was partly a practical decision: we already have generically encoded book files for over 22,000 pages of operating system manuals produced by over 40 writers and editors; over 50,000 pages and tens more writers and editors if layered products are included in the count. If nothing else, the scale on which we produce documents dictates evolution not revolution!

In addition, an early prototype based on the book model was enormously successful in a series of human factors tests with both DIGITAL employees and customers. We are finding that users presented with a book model — with a graphical, mouse-driven (point and click) interface — can use the software with little or no instruction. In addition, there is a usability advantage in presenting users with books because they are already very familiar with them, and many users are already familiar with our documentation.

In a major departure from the book model, we have equated pages with topics. That is, "pages" are variable size depending on the length of the topic they represent (the only new tag invented for on line documentation allows writers to specify the topic granularity of their documents, based on header levels). There are no page numbers, and so far no one has complained: as we suspected, the interface and the medium make page numbers and fixed-size pages unnecessary

The interface for the on line books is a 2-window, mouse-driven (point-and-click) interface. One window (the selection window) contains the table of contents or index (radio buttons allow switching back and forth between the two). To access a page of text, the user merely points the mouse cursor at a section or keyword and the associated text appears in the text window. From the text window, the user can go to the next page or previous topic.

In addition to facilitating ease of access, windowing capabilities on the workstation mean that the on line documentation can be displayed in a window alongside the window that is running the application the user needs information about.

6 FUTURE DIRECTIONS: HYPERTEXT

Hypertext, or "hypermedia" if video and audio capabilities are added to text and graphics capabilities, will take us beyond providing static documents whose structure and contents are fixed at each release. Instead of relying solely on the author's notions of how the information should be sequenced and linked, users will be able to provide their one paths and links through the documentation. In addition, they will be able to annotate the documentation, becoming co-authors themselves of the information, and they should be able to make their annotations publicly accessible to other users (on the other hand, some annotations they may wish to keep privately and not share with other users). These capabilities will bring some challenges to both database management and compact disc distribution.

Maintaining such information clearly must occur in memory since CD-ROM is non-writable (and it may always be desirable to distribute documentation in read-only form to protect our copyrights). This will require not only facilities to maintain links and annotations through one version of the documentation set, but to maintain the links and annotations over new versions as well. What happens to a link or an annotation if the documentation changes and a linked or annotated section is deleted?

These are the issues that will challenge hypertext systems, particularly hypertext databases that change on a regular basis, in the future. The hypertext vision is decades old, but only recently has technology begun to catch up with it. At DIGITAL, we are exploring ways of incorporating hypertext capabilities into on line documentation.

7 CONCLUSION

Given the evident data storage, durability, size, cost advantages of compact discs, DIGITAL is looking at on line documentation and compact disc distribution as a means of streamlining production processes and providing better service and higher quality documents to users. With graphical interfaces on workstation products and compact discs, applications such as on line documentation make a great deal more sense than they made in the past. Indeed, the users are demanding such applications.