A review of the history of an office system application is presented, highlighting the learning process that took place during its evolutionary development. This office system has served as the basis for a PRPQ (customized program) recently announced by IBM and known as the Professional Office System (PROFS). The general application architecture is discussed, with a specific focus on the use of virtual machines. Functional details of the various components are described, and the key distinction between office systems and office automation is addressed. The paper also discusses usage of the system and points up some of the benefits being realized by current users of a prototype system in IBM. The application function review details the electronic document distribution capabilities of the system.

A system for the automated office environment

by P. C. Gardner, Jr.

The office system developed in the Poughkeepsie laboratory of IBM and known within IBM simply as the Office System, or OFS, is a VM/370-based application intended to address the various text-processing requirements of technical professionals, secretaries, and managers. The text-processing needs of such a user group have been called "in-house publishing" and include all internal correspondence and documentation ranging from short informal notes to formal internally published reports. An office system should address all the various aspects of "document" creation, filing, storage, retrieval, and distribution. OFS is directed to this end.

History and background

The OFS application was developed as an internal tool within the IBM Poughkeepsie Information Systems (I/S) organization that provides data processing support to users within IBM at Poughkeepsie. In early 1970, I/S installed its first virtual machine

system, a System/360 Model 67 running the control program CP/67. That software environment was the predecessor of today's VM/370 (Virtual Machine Facility/370)[1] operating system. Users of the new computing system were presented with a rich set of tools to create and manipulate data. A text formatter, called SCRIPT, provided the programming staff with a view of the text-processing possibilities of the new facility. The interactive terminal support, known as the Cambridge Monitor System and later named the Conversational Monitor System (CMS), gave the user an individual system image with capabilities to personalize his or her own environment.[2] Background and general capabilities of the VM/CMS environment can be found in Reference 3.

The new machine was located in a building remote from the major I/S facility. The I/S people had no secretarial support and were in need of some form of administrative help. Early exposure to this virtual machine structure stimulated the thought that help could be found among the new system tools.

Looking to the new system for solutions, we undertook a small-scale effort to develop an initial application. Keeping track of business correspondence and controlling the local document files was a real problem. A primitive mail-logging facility was designed and implemented to address this requirement. All incoming documents were logged in a file called the mail log by entering index or filing records into an on-line display terminal. Each document was given a system-assigned index number. The document filing record contained this number, the author's name, recipient information, the subject, and various filing and status data. There was also some space for comments or keywords about the document and its content. The documents were then filed sequentially and were retrievable by the index number. This first tool helped the small programming groups cope with their document filing requirements and introduced the concept of office automation.

early prototype

The mail-log application was followed by the development of a document-creation routine to assist the programming staff with the task of typing the necessary business correspondence. The technique of using the text formatter and CMS editor to ease the burden of creating a business memo quickly caught on. The application, called MEMO, controlled the format of the document using previously created format control files. It prompted the user for the various fixed-field inputs such as subject and reference lines. The main body of text was entered free-form. The application handled all paragraph and margin formatting, thereby freeing the typist to focus on the words being keyed. The user needed only to respond to the few prompts and type the text of the memo with no concern for format. A nontypist could create a formatted memo with little or no effort.

As a result of the early user experience many incremental improvements that enhanced the general usability of the document creation tool were made. A technique was developed to capture all of the necessary filing information from the user-prompted responses during the text-entry process. The filing record was thus automatically created and logged as a by-product of the user input. This process freed the user from the burden and concern of filing the newly created document. Automatic filing was a big benefit to the early users of OFS.

Once the technique of creating memos was mastered, it became obvious that the next step was to transfer copies of these documents to other users of the system. The virtual machine system allowed users to move a document file from their private storage area to the common system storage, known as the spool files. The specific document transfer process was initiated by users issuing a "MAIL" command for a document that was stored on their local files. A copy of the document was moved to the system spool files. Recipients would read the document from the spool files and place the copy in their local files. The receiving function also updated the recipients' mail log with a copy of the document index record. The users' mail-log file would then contain index records for the various documents that they created or received. This process allowed copies of documents to be moved between users of the system. With this simple system technique the first attempt at electronic document distribution was set in motion.

Along with the new MAIL function, the document-filing application was extended to provide a search and retrieval capability. The basic capability allowed users to scan their local files by searching for author, distribution, time span, and/or keywords. The results of the search process presented the user with a list of filing records for documents that satisfied the search parameters. The user would then refer to the index number to retrieve the desired document from the file cabinet or locate it in the on-line storage files.

assessment of prototype experience

By mid-1972 the set of functions had been expanded to include memo creation, local filing, document distribution, and search retrieval. The user reaction had confirmed the value of the functions and the general usability of the tools. The activity was at a point where a total assessment of the prototype experience was called for. The developers gathered all the known information on requirements of an office system, listed the problems and limitations noted in the prototype, and considered what had been learned from the exposure to the virtual machine environment.

The requirements, taken individually, appeared to be rather straightforward. The capabilities of the host virtual machine

system were more than sufficient to implement all of the major requirements. The challenge appeared to be in structuring the application such that the various tools could be brought together as a total package. The initial assessment highlighted the development of an application architecture as being a key item. The intent would be to encompass the many elements of the application in a manner that added structure and control to the resulting environment.

The large number of individual user data bases was recognized as the main problem existing in the prototype. That implementation allowed multiple copies of documents to exist with each user managing copies on his or her own local files. This factor was consuming a lot of file storage and generating a definite records management concern. It was also clear that if an information source was to evolve from this collection of business documentation, a common file would seem appropriate. Consideration of VM/370 and its implications as an office systems support environment prompted the most thought.

Office systems

It became apparent that there was a vast difference in scope between the concepts of office systems[4] and office automation. The implementation of automated office tools was recognized as being a component application in a total office system solution. The real meaning of office systems was seen as the introduction of the full range of information systems capabilities into the office environment. It was believed that a professional decision maker should use a system with relatively open-ended capabilities as opposed to one limited to text automation functions. Under VM/370 it appeared that an office automation application could be structured into a total systems environment, thus producing an office systems solution.

In order to better describe this difference we consider office automation in three fundamental categories. First, one may note the pervasive use of word-processing equipment and related stand-alone devices supporting text-entry editing aspects of office automation. This activity consists of a relatively small set of applications that addresses the text-entry processing requirements of the office. The users of these tools are the typists and document support personnel whose responsibilities are generally limited to the typing and editing functions. They have no need for other system capabilities.

The second phase of office automation could be viewed as a structured systems solution. Here potential users are presented with a package of application functions that address the automa-

tion of various office procedures. Such a system is highlighted by its structured control and procedural operation nature. The users of this type of tool would most likely be part of a firm's operations support function in areas such as administrative control that are highly proceduralized. In addition to the basic automation tools, users may have a need for structured access to data bases or information query facilities. These added capabilities are available but usually are restricted to the specific support task at hand. There is little, if any, ability (generally no need) for individuals to expand the scope of their usage activities.

A third aspect of office automation would be the implementation of a functionally rich user environment. The intent would be to provide flexibility of use with a wide range of system capabilities. Such an office automation application, when coupled with other information systems facilities available on a host processor, introduces the broader application of office systems. This type of system would be applicable to the needs of professionals and decision makers, that is, those whose job responsibilities are best complemented by flexibility and richness in functions. VM/370, with its wide range of functions and capabilities, appeared to be an ideal operating base on which to begin structuring a total office systems interface. This underlying distinction between the potential scopes of each solution was and has continued to be a major distinction between the OFS application and other available tools.

Architecture considerations of an office system

After assessing experiences with the prototype, we began work to establish the architecture that we hoped would carry the application towards a future goal that itself was not completely understood. It was felt that the architecture should accommodate a few fundamental concepts. One important consideration was to have the end user perceive his office as being supported by a dedicated "minicomputer." This element was emphasized in most of the requirements statements that were reviewed.

The appropriateness of a central shared data base had become obvious. The need to control the number of document copies and the desire to more readily share a common document file across a user organization became key objectives. It was also clear that the application being undertaken would evolve over time, as functions were understood and included. This thought prompted a design that was sensitive to functional separation and allowed additions to be made easily.

The last element that guided the architectural decisions was a vision of where the computer hardware technology was headed. It appeared that the three basic office system functions of

information creation, storage, and distribution would eventually find their way into specialized processors such as intelligent user terminals, document storage "machines," and distribution control facilities. With these concepts in mind, it became apparent that the environment presented by the virtual machines was ideal for this application. It was also perceived that the VM/370 virtual machine control program facilities might be just what was needed to address the goal of an office system.

The individual users' CMS facility provided the desired personal computer image and, when used with the VM/370 communications facilities, allowed for interoffice communications. The need for a common data base could be solved by establishing a single data base virtual machine. This approach also created the possibility for a common searchable file, accessible by a large set of users. The same concept of dedicating a virtual machine applied to the distribution function. Separation of the application into these three main components would also provide a level of function separation that addressed the concern for ease of application change and extension. Finally, this configuration complemented the ideas on technology direction by separating the application along functional lines that would be adaptable to those possibilities.

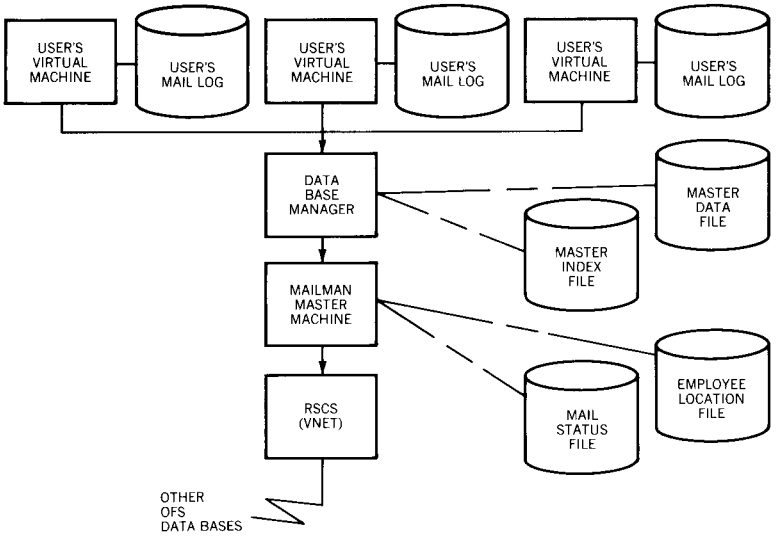
First system installation

A main element of the detailed system design was the extendability of the various functions. The early system was jestingly referred to as the "world processing" system to indicate the need for open-ended design. The general system perspective focused on the true office system as a link to the user's total information processing needs. The idea was to have an interface that should present the user with a window into the information resources of the user's enterprise and perhaps beyond.

As previously noted, the office automation tools were considered an important subset of the much broader office system solution. These tools were the means to get started and established the introduction point for what should logically follow. With this direction established, the process of extending and enhancing the application was begun.

The user requirements list was still very long. Some items only required application program implementation such as action file followup. Other requirements, such as multipage display, were dependent on technology that was not yet available. The intent was to implement a system from which the envisioned future system would later evolve. The individual requirements would be adaptable to the selected architecture in an evolutionary manner.

Figure 1 The OFS environment



If this process was to succeed, it was important that a good picture of the broad general objectives be understood, thus ensuring that each individual design consideration could be considered in that perspective. The process would prove useful during the many decisions selecting tradeoffs when apparently equal solutions were being considered. When other aspects were considered to be equal, the long-range direction concept would generally point to the selection of the option most consistent with that objective.

The significant elements of the initial production versions of the application were the common central data base and the controlled delivery of mail. The system controls were also adding structure to the user community, and the general interest level was picking up. The new system was first installed in Poughkeepsie in October 1974.

Three virtual machines

As noted, in order to accommodate the various design objectives and take advantage of the VM/370 environment, the way to divide the total application activities into three segments (Figure 1) was selected. In establishing the architecture, functions such as text entry, editing, and data manipulation were designed to run in the individual user's CMS virtual machine. The second separate virtual machine was set aside as the data base function. It was called the Data Base Master Machine and was intended to control

the common data files and manage the needed document security, authorization, and audit functions. The third component, the distribution manager, was referred to as the Mailman Master Machine. It was intended to handle the transfer of electronic correspondence and control receipts to guarantee electronic mail distribution. The latter two machines were to run continuously, in a disconnected mode, in order to provide service whenever needed by a user machine. With these basic architecture designs in mind, a more detailed review of the functions at the individual virtual machine level can be done.

User machine

The individual user's virtual machine is intended to provide a general-purpose personal computing environment. As noted above, all of the users' interactions with the OFS application facilities take place within this virtual machine facility. The document preparation routine is a major user application component that runs here. It provides for the control of all document entry and edit activities. Mail processing and the document disposition function are other key application elements that run in the user machine component. In addition to these primary features, all time management and personal service routines, such as suspended file management, meeting scheduling, remind functions, and message processing are executed in the user machine.

Several application-related files exist in the user's local storage area. Most visible of these files is the user's mail log that contains the filing records for all documentation owned by the user in question. The user can add comments and/or action dates to this file to assist in the management and control of the correspondence that is being processed through the office. The search and retrieval function also runs in the user machine and uses the mail-log file to find the requested documentation. Documents are located by using a variety of search terms such as author, addressee, subject keywords, or date.

In addition to the office system application, a full range of other user applications is available through the general CMS environment. A typical OFS user may use the CMS file management functions to work with some job-related data on his or her personal file storage. Various utility-processing capabilities are available, and a user may find the need to run some specialized program function. As the users increase their familiarity with the new tool, they are better able to take advantage of this rich personal computing environment. The IBM Thomas J. Watson Research Center has focused considerable effort on providing end-user tools for the CMS user. A comprehensive description of that environment is contained in Reference 5.

The CMS machine also provides users with a communications link to the rest of the information system resources of their enterprises. Using intervirtual machine communications facilities, a user may send files or messages to other users of the local system, or the user may communicate with the networking facilities to initiate messages or file traffic across the communications network.

These basic system communication capabilities can be structured into applications that facilitate user interaction or access with local or remote data bases and execution processing. The user is given the image of being "plugged" into the network and can access the facility or communicate across it as the need arises. The applications that support these activities have been termed DATASTAGING[6] tools. The specifics of their use and general capabilities are discussed in Reference 7.

Additional facilities, known as terminal passthrough, allow the user to access other interactive systems. Such access is attained by logically attaching the user's display to a virtual machine, serving as a terminal message-switching concentrator, that, in turn, is attached to remote interactive services. This facility rounds out the general system control environment available to the CMS user. With such a wide range of additional applications available, the user is able to execute a comprehensive set of local tools, transmit and receive files from other machines in a local communications network, and access remote machines for data base inquiry or retrieval.

Data base manager

The data base manager (DBM) is a true virtual machine that runs as its own task manager under the CMS control environment. It performs numerous application functions such as updating filing records, storing and retrieving documents, assigning filing numbers to new documents, and validating user requests. The various tasks performed by the DBM are designed to interrupt themselves after various time increments and return control to the task managers. The amount of time is determined by the relative priority of the task to be completed. The task manager always gives control to the highest-priority activity. This technique was considered the simplest that would be independent of the system control program and also provide sufficient task-dispatching flexibility to meet our anticipated performance requirements.

Most of the functions that would affect end-user performance execute in the user's CMS virtual machine. For this reason it was assumed that the data base performance considerations could be easily accommodated. Many of the time-consuming tasks done by the data base manager, such as storing a document or updating

index files, are asynchronous from the user interface functions. It was felt that a user would accept a few seconds delay when retrieving a document from the data base manager. Other tasks, such as assigning numbers or validating authorization, items which were in-line processes with various user functions, should be responsive in the order of a second or less.

The data base manager controls two types of data files: the master index file and the actual document files. The master index file contains index records for each document that is stored in the data base. The records contain all of the document filing information and necessary system status indicators, along with retrieval authorization flags. The other file elements in the data base manager are the actual documents. They are individually contained in the master data base and managed as CMS files. The document files are stored in compressed and compacted form to save storage space and may be optionally encrypted for added security.

The data base is physically allocated across "*n*" CMS mini-disks, which may reside on separate real-storage devices. The data base is logically structured into 256 sub-data-base elements. Users are given authorization to access documents in some subset number of these logical data bases depending upon their organizational position. The authorization concept was based on a pyramid structure, where senior management in a given organization may be given access to the entire data base consisting of all 256 sub-data-bases. In the case of a department clerk, this authorization may be restricted to his or her department's individual sub-data-base files.

Retrieval access is structured into four major categories. General information or announcement notices are usually placed in sub-data-base (SDB) 0. Documents stored in SDB 0 may be retrieved by any authorized user of the system. Other business correspondence is normally stored in the user's default departmental sub-data-base where it can be retrieved by any user having access to that area of the files. The author of a document may place further restrictions on it, such as controlled distribution which restricts retrieval of a document to only those users who have been given permission to do so by a previously authorized user. This type of authorization starts with the author and is extended to others as the document is mailed from user to user as in a chain mail process. The final retrieval authorization is the restricted distribution control. In this case an author, selecting restricted distribution, limits retrieval authorization to those users to whom he, and he alone, gives retrieval authorization. For this category of documents, chain mailing is not available.

The data base manager performs extensive activity journaling, maintaining records of all transactions on an audit trail file. This

file is available to installation support personnel for both accounting purposes and activity analysis of the user population. Data extracted from this file can be useful to an organization in determining value-added benefits. The time between completion of a document and its delivery or the number of times a document is retrieved from the files are examples of elements that can be evaluated. Further analysis may be useful in projecting items such as a need for additional hardware facilities or perhaps even identifying the need for end-user education.

Distribution virtual machine

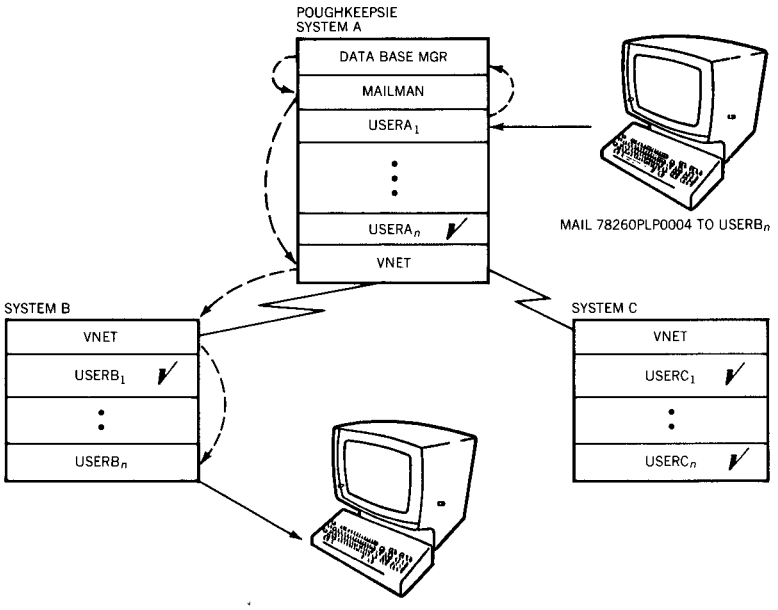
The distribution manager virtual machine controls the distribution of mail notices and manages the receipting of all such activity. The distribution process operates with two primary data files. The first file consists of distribution status records used to track mail in the process of being delivered. This file is controlled by the distribution manager. End users may query the distribution manager to determine delivery status. The system will provide the requester with a list of all as-yet-undelivered mail that they originated. The second file supporting the distribution process is the Employee Locator File (ELF) and is managed by the data base virtual machine.

The ELF contains user name and address information for all members of the local user community. Users are able to add their names to the file and make corrections to reflect changes in address information. The file can be queried to determine the electronic address of any member of the local user community or to look up phone numbers.

The ELF file can also be used to automatically generate notice records in response to a document mail request. This technique is currently being experimented with. The specific process involves retrieval of the detailed distribution list information from the mailed document by the distribution manager. This list is then compared against the employee locator file to resolve electronic addresses. Where matches occur, distribution notices are automatically sent. In the case of no match or multiple matches, the information is returned to the user, who is asked to supply additional mailing information or to authorize printing copies of the document on the local hard-copy output facilities. Effective use of the automatic mail distribution function is considered the key to promoting the concept of soft-copy distribution and having users gradually give up the habit of generating hard-copy output.

The distribution facility is capable of "talking" to the teleprocessing virtual machine, known informally as VNET (RSCS—Remote Spooling Communications System),[8,9] which in turn will communicate with other real virtual machine systems. There are

Figure 2 Local mail process at Poughkeepsie site



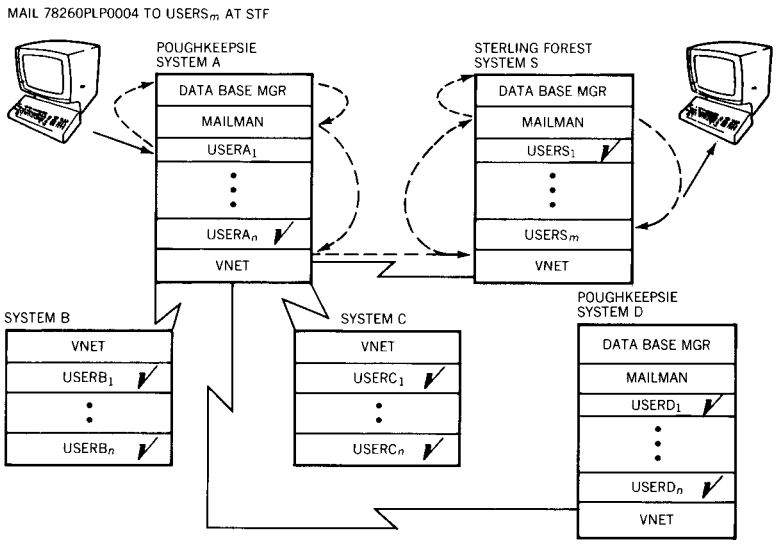
currently more than 370 CPUs that comprise the internal IBM network.[7] Its communications facilities connect all of the major development and manufacturing sites and extend to many IBM locations worldwide. The network is formally known as the IBM Corporate Job Network but is often referred to as VNET because of the extensive use of the VM/370 control program on the network CPU nodes. When one or more of these remote virtual machine systems supports the OFS application, an environment for receiving remote electronic documents exists.

Electronic document distribution

Using the transmission facilities of VNET as an intersystem file transfer medium, a comprehensive "electronic mail" function has been developed based on the OFS application. A document distribution facility has been implemented that spans multiple real systems and provides guaranteed delivery with data-base-to-data-base receipting. The VNET communications act as the carrier, and OFS provides the user procedures along with overall control of the distribution process.

The VNET communications links establish a network environment where a single application program can span multiple processors. The Poughkeepsie OFS application environment consists of a distributed front-end processing capability with a central shared data base. Figure 2 depicts the local mail process at the Poughkeepsie site. The VNET links form the various real machines in the

Figure 3 Process of sending mail to distant site



network into a total system environment. The single application distributes its component functions across the real machines in such a way as to create a single user interface encompassing this multiple-machine configuration. In this configuration, OFS presents a good example of the concepts of distributed processing and shared data bases and how they can be implemented across a system complex interconnected by VNET links.

In the local OFS environment, consisting of several VM/370 systems, one would find individual users working with the data entry tools from any one of the real VM/370 systems. Their individual virtual machines communicate with the single central data base virtual machine over the local VNET communication links. Any number of users operating off the local complement of virtual machine systems can store documents in and retrieve documents from the single shared data base virtual machine. The two VM/370 systems not running the data base virtual machine can be viewed as distributed processors. To consider the shared data base concept, we must look at the interaction of the distribution applications with other locations.

The distribution manager virtual machine controls the flow of correspondence across the total complex of local real machines. If mail is to be sent beyond the local application environment, the distribution function will communicate with a counterpart distribution function running at the destination location as shown in Figure 3. In this case, the application extends itself one level of distributed control. The local distribution facility communicates

across the VNET link to a remote copy of the application. The two distribution managers can exchange electronic documents, update each other's data bases, and maintain total control of the process with full transaction receipting back and forth.

This combination of a central data base and distributed control creates an application environment that will span any number of real systems. The individual user sees a single application interface, regardless of which real VM/370 system he is running on. The intervirtual machine communications capabilities of VM/370 and the intrasystem communications supported by VNET make this possible.

Growth of the internal IBM network and its extensive I/S resources have increased the potential usefulness of the system. The pervasiveness of the systems and attached terminals that have access to the Corporate Job Network[7] makes the concept of an electronic address for every interested employee a realistic possibility.

Enrichment of the functions

The period between 1975 and 1977 was devoted to extending the scope of the functions contained in the application. The implementors relied heavily on local user experience for validation of the functions and assessment of the usability. The biggest challenge presented to the developers was to temper their views as systems programmers with the perspective of the nonprogrammer users. It was clear, in both obvious and subtle ways, that a trained programmer and a novice administrator would not see the same system image when attempting to use the new tool. These differences had to be recognized and accommodations made for them in the user interface. An example of this difference was the realization that a programmer would rather remember that the short command "RETR" was needed to get a document out of storage, whereas a secretary would prefer to type "RETRIEVE."

The nature of the development task lent itself to the implementation of a command-oriented user interface. The objective was to add function and understand how these functions would be used in the office. It was understood that this type of interface would be somewhat challenging to learn and use. For this reason it was possible to develop a functionally rich application, but it was obviously faulted for its usability.

A major milestone occurred in late 1977 when IBM entered into a joint study with a customer. The customer was just starting to undertake some office automation experiments and felt that the IBM system would provide a good base for that work. The

customer brought some much-needed insight to the application, and both parties learned from each other. After a year of further development, the application evolved to what was felt to be a functionally complete system. The first formal source library of over 50K lines of code was established. This marked the completion of the logical first phase of development. The future direction was now coming into focus. A paper describing this version of the application was presented by IBM at the Boston SHARE Meeting in 1978.[10]

The system at that point addressed all major aspects of office automation (document creation, filing, retrieval, and distribution) and provides some limited elements of personal service (action files, buckslip forwarding, and message processing) but was somewhat limited in its usability. Its general acceptance was limited by human factors aspects. The command set was quite extensive and was difficult to use by anyone inexperienced with computer systems. There was general agreement among the developers that it was now time to address the ease-of-use and user-friendly aspects of the system.

full screen support

The need for full screen menus that would prompt the user and operate through program function key selection was apparent. A prototype office automation application,[11] conducted at IBM's Data Processing Division Headquarters, tested the use of full screen presentation techniques and clearly confirmed their value. This insight, along with feedback from the users of the OFS tool, helped to set the new direction. Development of the new user interface took about a year. The developers at the customer site provided most of the usability improvements. They were supporting a large user population and were getting more of the valuable end-user requirements. Many of their users were not experienced data processing professionals. This factor required a thorough assessment of the applications usability and presentation to novice users.

The full screen support made the application more friendly. Instead of requiring a user to remember the long list of commands and the associated parameters, the full screen version presented the command functions for selection. The required user parameters were made visible by a "fill-in-the-blanks" technique and use of the program function keys (PFKs) as shown in Figure 4. A user could step through many of the application functions by simply depressing specific keys. Most of the full screen support was implemented at the customer site.

The new support confirmed that the application could be more user-friendly. It was also discovered that this support made it easier to train new users and was more supportive of casual users. The application screens were able to present a broader

Figure 4 Fill-in-the-blanks technique

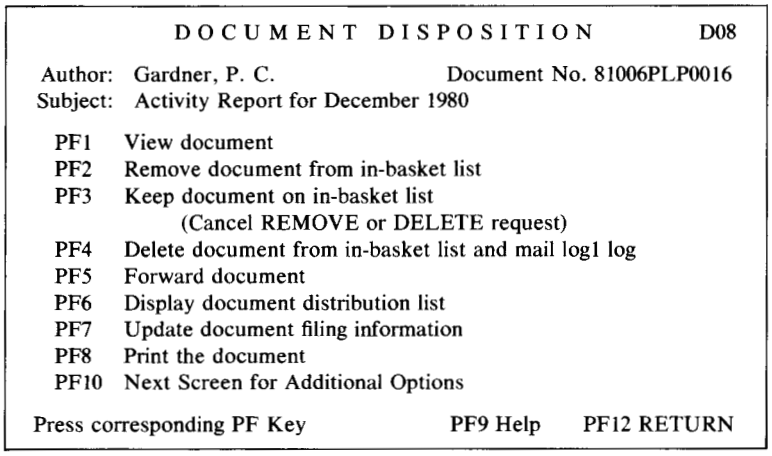
```
                    DOCUMENT SEARCH                    D01
Enter Desired SEARCH Parameters    Userid ___ (if not searching own)

  AUTHOR  gardner          /               /
AND
  TIME    jan81            /
AND
  KEYWORD ofs              /office         /
AND
  ADDRESSEE                /               /
AND
  IDENTIFIER     /    /    /
AND
  DESCRIPTION    /  /  /
AND
  ACTION         /          /

PF1  Search for items DUE
PF2  Find documents that contain ALL of the keywords
PF3  Replace (and) with (or) between parameters

Press ENTER to initiate                PF9 Help    PF12 RETURN
```

picture of the system's range. The novice or casual user saw more and thus learned more. The end result was that the first full-screen-oriented version clearly improved on the usability of the application, and additionally it provided insight to the next step.

An important observation was made when it was recognized that the screen should not only be easy to use but should convey information that would help the users to better understand the environment they were working in. This capability would become a key element when development progressed to the point of addressing end-user personalization. It was becoming increasingly clear that the format and content of the screen presentation were of primary importance. The ease with which an installation or individual user could customize or personalize the screen content would affect the acceptance and usefulness of the new tool. When the users understand the environment and how the various functions and system capabilities work, they are better able to adapt these facilities to their individual needs.

In addition to the various usability items, support for scheduling meetings was introduced for the first time. This facility allowed individuals to maintain their personal schedules on-line and view one another's calendars. Secretaries and principals were able to schedule meetings by using their office display to find an appropriate time slot on the various attendees' schedules. Authoriza-

Figure 5 Open-mail process



tion structures were in place to control who could see the schedules and who could update any given schedule. The facility was well received and generally considered a beneficial first step. Copies of the application were distributed to other IBM sites, where it was given further user exposure and acceptability testing.

process flow control

The first full screen version demonstrated the ease-of-use technique, but the application still was a collection of numerous functions that operated independently. Switching from typing a memo to searching for a reference document was a cumbersome task that required the user to keep track of various amounts of application data as they moved in and out of the desired functions. Adding a level of application process flow control was established as the main objective for the next release of the system.

Release 11, as the next one was designated, took approximately one more year to complete and was put into production at the customer site in November 1980. It was intended as the first product-quality-level release of the application. Emphasis had been directed at consistency throughout the system to ease user training. The addition of help facilities was introduced to further support the new or casual user. An explanation of the function being performed with a how-to example would be presented by depressing Program Function Key 9. The primary enhancement was the introduction of the structured process flow control. The flow control was a first attempt to provide for the structured connection between functions that would normally be used in some logical sequence.

The open-mail process is perhaps the best example of the improved flow control (Figure 5). Under the general control of

the open-mail function, a user is able to view all incoming messages and documents, take action on the messages such as replying back to the sender, forwarding to another user, optionally filing the message, or passing over it for a later session (Figure 6). Formal correspondence can then be viewed, routed with a buckslip for action, left suspended, or filed. All of the functions normally associated with the activity of going through one's in-basket are available in a logical sequence and are initiated with program function key control.

In order to present this structured flow to the user, considerable improvement was required within the basic application code. All user modules were rewritten to be reentrant and executable out of the commonly accessible, discontiguous saved segments of the control program,[12,13] thus allowing the application to more readily transfer control from one user function to another. Application performance was significantly improved, but what was more important, a new level of flexibility was introduced into the system. The application could move control from one function to another in a rapid yet structured manner. The individual could direct the application through the various elements in the order most adaptable to the task at hand. Switching from document preparation to search and retrieval for reference information, while sending off a few related messages, could be viewed as an example of the intended user activity.

personalization

Once the general system facilities were in place and the usability was at an acceptable level, the desirability of increasing the personalization of some of the new user tools came to the forefront. The first attempt at personalization was directed towards the initial application menu screen (Figure 7). With this feature the users were able to select, from a comprehensive list of application functions, those elements that were viewed as most significant to their office operation. These key functions could then be identified as part of the user's application profile, which describes the content of the initial application menu screen. In addition to OFS functions, users could select any valid CMS program to be included in the list. (Function and presentation syntax can be tailored to the end user's specific needs.) Users' initial screens would then be defined according to their unique office needs. This effort was the first attempt at structuring the individual user's view into the information system resource.

At a more general level, personalization has been implemented for the output-handling facilities. The users may describe to the system those printing facilities available to them and establish default output characteristics, such as special print parameters for variable-font devices. At any point when hard-copy output is to be created, the user can select from the list of printing devices defined for his or her needs.

Figure 6 Use of general control of open-mail function

```
MSG FROM: TDCSYS2 (bob)          TO: TDCSYS2 (paul)          03/09/81   4:56:47
Subject: Draft of OFS paper
I have reviewed the OFS paper. Looks ok. I would suggest that you include a figure to highlight the use of the
message facility.
. . . .
Bob Rogan
253 - 1234

                                  E N D   O F   N O T E

PF1 UP        PF2 Down        PF10 Next Screen        PF11 Previous Screen        PF9 Help
REPLY to: TDCSYS2 (bob)                                                           E02
Subject: Reply to Note 03/09/81 14:56      Draft of OFS paper.
Thank you for the help. I will plan to include some figures.

PF1 Top       PF2 Bottom      PF3 Delete Line      PF4 Add Line      PF5 Nulls On      PF6 Format
PF7 Send      PF10 Next Screen      PF11 Previous Screen      PF9 Help      PF12 Cancel
```

Figure 7 Initial application menu screen

```
                    P R O F E S S I O N A L   O F F I C E   S Y S T E M

PF1   Schedule Appointments (PCG)                    Time 2:58 PM
PF2   View In-Basket
PF3   Search and Retrieve                        1981  MARCH  1981
PF4   Document Preparation                       S  M  T  W  T  F  S
PF5   File Hardcopy                              1  2  3  4  5  6  7
PF6   OFS Directory Information                  8  9 10 11 12 13 14
PF7   Send Messages / Review Notes              15 16 17 18 19 20 21
PF8   Access Other System's (PVM)               22 23 24 25 26 27 28
PF10  Desk Calculator Facility                  29 30 31
PF11  Away from Office                              Day of year: 068

Press corresponding PF Key                  PF9 Help     PF12 EXIT
```

These few initial personalization techniques clearly demonstrated the desirability of this approach. It is obvious that additional work in this area will continue to improve the usability and usefulness of the system tools. Recognition of the dynamics and individualities of the business office clearly highlighted the need to further personalize system applications in order to adapt them to these environments. In the final analysis, personalization may well be the key to making office system solutions finally acceptable.

Current Poughkeepsie usage

Use of the OFS application has spread from a few of the early developers to a broad-based user community that spans several IBM sites. It has been identified as part of the internal office systems strategy and is being treated as a formal internal "product." A central control location has been established to distribute the application, control and promote its use, and coordinate user concerns and suggestions. A development function is in place to address the many new requirements that are being identified as system usage increases.

At the Poughkeepsie site there are currently in excess of five hundred users and the number is growing. Poughkeepsie I/S controls VM/370 systems that provide interactive CMS support to the site. The systems are used for text-processing activities, some program development, and general-purpose personal computing. Text processing has been the major thrust of local user support over the last several years. Many of the text-processing support tools complement the administrative automation application. The OFS users are being supported on the production VM/370 services and are thus sharing the OFS functions with the users of all of the other VM/370 end-user applications.

The emphasis of the office systems usage to date has been on the creation, storage, retrieval, and distribution of internal memorandum correspondence. There are currently over 400 programming and engineering professionals who use the tools on a regular basis to handle their documentation needs. In addition to the professionals, approximately 80 secretarial and clerical users at the Poughkeepsie site have access to the application. Of these, most are relatively active users, with more than 50 using the OFS tools on a full-time basis. The Administrative Service Centers in the manufacturing area are in the process of completing training for approximately 100 support personnel. When the training ends, the centers will be fully committed to the use of the OFS tools. The I/S management team, consisting of about 40 managers, has been using the system as a principal support tool for over two years. Some of the usage characteristics of these four distinct user groups are now highlighted.

secretarial and administrative users

The secretaries who use OFS find the mail-log facilities and document creation functions to be the most valuable. Mail logging is done by using the system tools to control all incoming hard-copy documentation. Some secretaries have switched to a totally sequential filing system, relying completely on the search retrieval functions of the system to find hard-copy correspondence. Others have used the on-line mail log in place of hard-copy logging, but continue to utilize previous filing procedures. The on-line mail-log filing record can accommodate this practice

through reference fields identifying the actual physical location of a given document and relating it to a project file or filing category.

The general consensus is that the technique of using the on-line mail log makes records more accurate and introduces some filing flexibility. The search and retrieve capabilities are of value to the secretaries, but the principals and professionals especially find it beneficial to be able to use the search functions independent of the secretarial staff. This facility is particularly helpful when administrative support personnel are not available.

The Administrative Service Centers have been able to implement procedures to better distribute the typing workload as a result of using the OFS tools. Documentation entered by one person can be temporarily stored in the common data base and worked on by other members of the department. Common departmental and organizational files are now possible, and the burden of record keeping has been greatly alleviated. Use of the document preparation facility, in conjunction with the on-line text formatters and editors, has greatly enhanced the productivity of handling the document creation workload. Initial assessments have indicated a productivity improvement of more than 25 percent in the document creation and revision process over the previous magnetic card typewriter techniques. The use of output printers, installed directly in the Administrative Service Centers, has improved the turnaround time and is now allowing distribution of final output from one center to another. Delivery of a document created in one building is facilitated by having it printed on the output station in another building, with resultant savings in distribution time.

professional users

Use of the system by professionals has caught on well despite some early skepticism. The intent was for all professionals to utilize the system in such a manner as to support all of their individual administrative requirements. Documentation productivity was emphasized by requesting that professionals do their own typing. It was anticipated that the professionals, most of whom were familiar with the basic terminal keyboard characteristics, would be able to type information into the system as easily as (potentially more easily than) recording the same information on paper. The experience of the past few years has demonstrated this assessment to be true.

Through the prompting, formatting, and automatic filing capabilities of the system, the burdens of document aesthetics and related administrative concerns are removed from the process. It has been found that with a minimal amount of exposure to the system (using it approximately a week or two), a professional can do his or her own typing with no increase in total work effort. In most cases, after an extended period of use (typically a few months),

the benefits of the on-line editor and full-screen text manipulation start to increase overall productivity, thus making the process more beneficial. Both in initial use and after training, the end results are still immediate availability of a final document, clearly resulting in a dramatic improvement in overall turnaround time. Both of these points have resulted in a general acceptance of the system by the professional staff.

Numerous examples exist of how general organizational effectiveness has been improved through the use of the system tools. Improvements have been noted in the quality of business memos and activity reports. A noteworthy benefit is the timely distribution of weekly status meeting information to the large number of technical support personnel requiring its availability. In this example, the various status meeting chairmen are able to update status documents immediately after the weekly session and distribute clean updated documents to all involved personnel. As a result, the information is more current and meaningful than it had been in the past. Many other benefits are difficult to measure specifically but are of obvious value to the overall performance of the organization. It has been recognized that the meetings are requiring less time and are considered more productive. The accuracy of status information is also contributing to a general improvement in control.

management users

The management staff has been using the system as a principal-support vehicle. The on-line calendar and scheduling facilities are proving to be especially beneficial. Users are able to enter meeting notations and move or copy appointments from one date to another. With proper authorization a user can look at the calendar of another user and schedule time. The more meetings to be scheduled, the more useful the tool. It provides a handy real-time way of looking at various individuals' calendars that in the past were unaccessible or required numerous phone calls to coordinate. In addition, consider the fact that calendars often reside physically in an office that is at some distance from the person needing the information. The task of arranging a meeting or locating someone can now be done directly through a principal's or secretary's terminal.

The use of the open-mail facilities and the ability to "manage" incoming correspondence through normal in-basket processing is also a benefit. The major improvements are seen as better control and follow-up and the vastly improved speed with which action items can be moved through the organization. A number of the managers, who are frequently out of town or at meetings in remote buildings, have found it useful to periodically check their "mailboxes." They communicate with the Poughkeepsie machine through the internal IBM communications network and gain access to their incoming messages and memos. Such tools are

thus providing a valuable means of staying in contact with the day-to-day information flow of one's office and allowing a principal to continue participating in the management of the office while attending to other business at offsite locations. More importantly, however, the on-line mailbox and complementary message-routing facilities have speeded up the flow of important business-planning and decision-making information among the management and professionals of the organization that have the facilities available.

The use of the informal message has caught on well, and it is used extensively by many of the principals having access to the system. Messages are used in place of telephone calls that often would have taken several attempts to complete. The tool is also providing an ideal means of communicating with off-shift personnel or a means of initiating activity when working late in the evening.

General improvement in information availability is just now becoming understood. Because of the ease with which documents are generated, the professional users are being encouraged to document their activities more completely. This focus is resulting in a rather comprehensive project and activity file. The amount of detailed project reference material is increasing and resulting in more timely and accurate communications. Management is now attempting to better understand the usage techniques that will apply this new source of documented information to their business decisions. The task of administrative follow-up and closure on business activities is being greatly enhanced by the readily available memos and reports. The exploitation of this new information source may potentially hold the biggest benefit for the management decision makers.

Summary

The OFS application has evolved over a period of ten years of use and experimentation. The broad-based objectives of office systems have guided the ongoing development. VM/CMS has proved to be ideal as the control program base for such an activity. Actual user experience and feedback have been an invaluable means of validating both implementation and presentation techniques. We have found that with every new insight several new questions surface. The current OFS application is the product of this evolutionary process.

Considering the computer network and user tools that exist, we may conclude that the concept of electronic mail is truly operational. It remains for us to learn the procedural and administrative techniques of exploiting these tools and to fine-tune the whole

process. Any actual or precise assessment of the value is still to be determined. We are finding that as use of the application grows, we continue to uncover new areas of potential. A lot has been accomplished, and more is being done, but much more is still to be learned.

I have found the task of bringing information systems capabilities into the office of the user who is not familiar with data processing to be very challenging. That process has often required an implementation approach different from traditional applications. I feel that we have taken the first step and are excited by the possibilities ahead.

ACKNOWLEDGMENTS

The author wishes to recognize the many years of development and personal effort by Bob Rogan. Significant contributions were made by Jim Stewart and Henry Young and their organization at the Amoco Production Research facility in Tulsa, Oklahoma. Also, the dedicated support of Marjorie Percy and the Poughkeepsie Advanced I/S Application staff is greatly appreciated.

CITED REFERENCES AND NOTE

1. *IBM Virtual Machine Facility/370 Introduction*, IBM System Library, GC20-1800, IBM Corporation; available through IBM branch offices.
2. *EXEC2 Reference*, IBM System Library, SC24-5219, IBM Corporation; available through IBM branch offices.
3. L. H. Seawright and R. A. MacKinnon, "VM/370—a study of multiplicity and usefulness," *IBM Systems Journal* **18,** No. 1, 4–17 (1979).
4. The term office systems is used to represent the full range of information systems capabilities that may be appropriately used in a professional office. They include data base query, personal computing, special services such as business graphics, etc., along with office automation tools.
5. A. M. Gruhn and A. C. Hohl, "A research perspective on computer-assisted office work," *IBM Systems Journal* **18,** No. 3, 432–456 (1979).
6. M. E. Petersen and R. A. Pletzer, *Network Datastaging: A User's Guide*, Technical Report TR00.2839, IBM Corporation, Poughkeepsie, NY 12602 (March 1977).
7. P. C. Gardner, Jr. and T. C. Hartmann, "VNET: How it developed and how we use it," *Proceedings of the Online Data Network Conference* (London), Online Publications, Ltd., Northwood Hills, United Kingdom (June 1980).
8. *IBM Virtual Machine Facility/370: Remote Spooling Communications Subsystems (RSCS) User's Guide*, IBM System Library, GC20-1816, IBM Corporation; available through IBM branch offices.
9. E. C. Hendricks and T. C. Hartmann, "Evolution of a virtual machine subsystem," *IBM Systems Journal* **18,** No. 1, 111–142 (1979).
10. P. C. Gardner, Jr., "An administrative automation application," *Proceedings of SHARE 51* **1,** 96–123 (1978).
11. G. H. Engel, J. Groppuso, R. A. Lowenstein, and W. G. Traub, "An office communications system," *IBM Systems Journal* **18,** No. 1, 402–431 (1979).
12. *IBM Virtual Machine Facility/370: Planning and System Generation Guide*, IBM System Library, GC20-1801, IBM Corporation; available through IBM branch offices.
13. *IBM Virtual Machine Facility/370: Systems Programmer's Guide*, IBM System Library, GC20-1807, IBM Corporation; available through IBM branch offices.

The author is located at the IBM Data Systems Division laboratory, P.O. Box 390, Poughkeepsie, NY 12602.

Reprint Order No. G321-5152.