Development and Implementation of Peer Review Plus: A Computer-Based Tracking System for Editorial Offices†

JEFFREY D. SPRING,* JOHN T. KEYS, and LORRIN R. GARSON

Research and Development Department, Publications Division, American Chemical Society, 1155 16th Street NW, Washington, D.C. 20036

Received February 21, 1989

Keeping track of manuscripts and reviewers in editorial offices can be a complex and time-consuming task, especially for offices that handle thousands of manuscripts annually. The American Chemical Society (ACS) has developed a computer-based system, called Peer Review Plus, which is used by editorial staff for tracking manuscripts, assigning reviewers, and generating statistical reports. The current software is a second generation system that required approximately 4 man-years to develop. The software centers around reviewer and author/manuscript files with a capacity of approximately 25 000 author and reviewer records providing rapid record access via several file keys. The system permits office staff to quickly select appropriate reviewers for manuscripts in specific subdisciplines, check the status of manuscripts, identify overdue reviews, list manuscripts for a particular author, list manuscript processing time, identify lack-of-action bottlenecks for manuscripts, etc. Word-processing functions have been integrated with the Peer Review Plus software, and automated telecommunication facilities have been developed to permit unattended transfer of data in a networked environment.

INTRODUCTION

Prior to acceptance of manuscripts containing original research for publication, editors send papers out for review by two or more reviewers judged by the editors to be knowledgeable in the subject area of the manuscript. The papers are reviewed for scientific rigor, originality, accuracy, and completeness. Before the availability of relatively inexpensive computer technology, editorial offices had no choice but to manage information concerning the status of manuscripts, and selection and use of reviewers, by traditional manual filing systems. Generally, these systems employed card files and a variety of "tickler" files. Normally, editorial offices had several such files, each of which was independent of the other (not cross-referenced), and accessed (a) alphabetically by corresponding author's name, (b) alphabetically by other authors' names, (c) by manuscript number, (d) alphabetically by reviewers' names, and sometimes (e) by areas of expertise of reviewers. Some offices used cards or paper clips in a variety of colors to indicate various classifications. In the late 1950s and early 1960s, some editorial offices used "edge-notched" cards for this purpose, especially for classifying reviewers' areas of expertise.[1] While these methods are suitable for offices managing a few hundred manuscripts a year and an equal number of reviewers, such methods can be very time consuming for larger operations. Moreover, gathering statistical information from manual files, even for a small operation, is labor intensive and tedious. Also, there can be logistical problems in keeping accurate data when there is more than one editor, and the editors work at different locations.

In 1979, the American Chemical Society (ACS) began installing computers in editorial offices for their journals to assist editorial staffs in managing information. The first journal to receive computer support was *Biochemistry* at the University of Washington in Seattle. The design, development, and implementation of the original software has been previously described.[2] Over the next several years, similar, but not identical, software was installed for the *Journal of the American Chemical Society*, *Inorganic Chemistry*, *The Journal of Physical Chemistry*, *Analytical Chemistry*, and the *Industrial & Engineering Chemistry* journals. While *Biochemistry* has one editorial office with all of its editorial staff in one location, many of these other journals have two or more offices with associate editors at distant geographical sites.

By 1982, it had became clear the software needed to be revised and standardized because each editorial office had somewhat different requirements and methods of operation. Also, the original software was not well suited for the telecommunication of information between offices at different sites. During its implementation, no allowance was made for telecommunications because the design requirements were for an editorial office in a single location. Moreover, the capability of the microcomputers in the late 1970s and early 1980s would have made such processing difficult, if not impossible.

When the software was installed in editorial offices that required remote access by staff located at other sites, modifications were made to existing programs to permit their use by dumb terminals with access via 300/1200 baud modems and voice grade telephone lines. From the user's viewpoint, this never was particularly satisfactory for numerous reasons including slow screen displays, "noisy" telephone lines, and no printing capabilities. Also, to meet each office's individual needs, we had to make modifications in numerous programs, resulting in nonidentical code which became increasingly difficult to maintain with our limited manpower. Thus, the revision described in this paper provided an opportunity to (a) take advantage of advances in hardware design, (b) incorporate software enhancements desired by editorial offices including telecommunication facilities, and (c) standardize the software at all locations to improve efficiency in software maintenance. This current version of the software (revision 2.0), called Peer Review Plus, is the subject of this paper.

OTHER SOFTWARE FOR EDITORIAL OFFICE USE

Two software packages similar to Peer Review Plus are used by the *New England Journal of Medicine* and the American Physical Society. The NEJM package, called the Editorial Management System (EMS), allows the editorial staff to handle more than 3500 manuscripts submitted annually. EMS was written in 1984 in COBOL by use of an SQL relational database and is available to NEJM editors at eight remote

† Presented as two papers at the Division of Chemical Information, 191st National Meeting of the American Chemical Society, April 16, 1986, New York City: "Development and Implementation of Peer Review Plus: A Computerized Tracking System for Editorial Offices", Jeffrey D. Spring, John T. Keys, and Lorrin R. Garson; "Automated Telecommunication Software for Editorial Offices", Jeffrey D. Spring, John T. Keys, and Lorrin R. Garson.

0095-2338/89/1629-0261$01.50/0

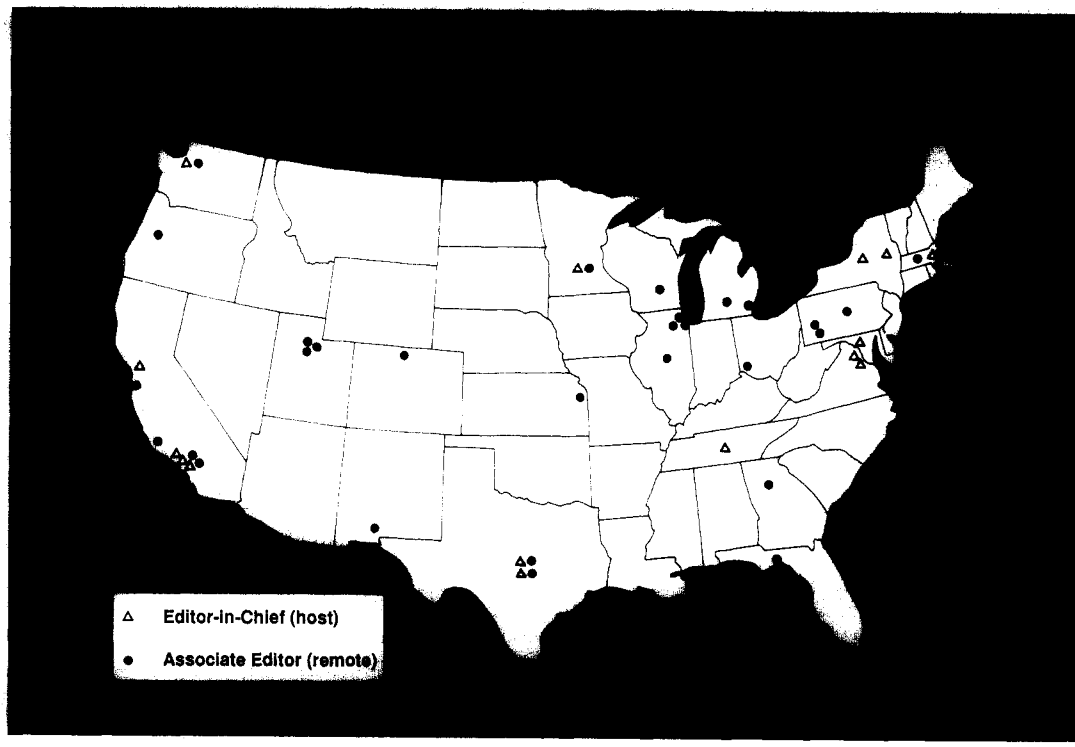

Figure 1. ACS editorial offices with Peer Review Plus systems.

sites. Editorial staff members are assisted by the EMS software in three areas of office operation. First, it maintains a database of over 7000 reviewers for the journal, along with searchable fields of expertise to determine which reviewers would be appropriate for a given manuscript. The EMS also assists in keeping track of manuscripts as they move through the review process, allowing the staff to efficiently answer questions on the status of a manuscript under review. Finally, it keeps track of copyright permissions.[3]

The American Physical Society, which currently publishes 55 000 pages of physics-related material annually, developed a UNIX-based system in the late 1970s. The software for the system is centered around a referee database and a manuscript database with the information residing in 1 gigabyte of storage on a Digital VAX 780. The referee database, with approximately 15 000 entries, includes areas of expertise and manuscript assignments along with additional information. The manuscript file contains the name and address of the corresponding author, when the manuscript was received, and a log of communications with the author and referee(s). Over 13 000 new manuscripts are processed annually. When the system was initially developed, all the editors were located at the American Physical Society. That has now changed, with 7 of the 22 current editors being "off-site" and having the capability of accessing the data files via long distance.[4]

PEER REVIEW PLUS

Development of the second revision of peer review software (Peer Review Plus) began in late 1982, and the first installation was for *Biochemistry* in May 1984. It currently is being used by 16 ACS journals at a total of 41 editorial offices (see Figure 1). As with the previous version, this one was developed for use on Alpha Micro computers (Irvine, CA). Most of the code was written in BASIC, with some of the subroutines in Assembly language.

A number of features were added to Peer Review Plus to make it easier to use. These included single-keystroke entry where possible (no need to press RETURN), both online and hardcopy "help" files for all programs, easily understood prompts that were consistent throughout all the programs, and incorporation into the software of editing features and commands employed in the word-processing packages used in the editorial offices. All programs are run from a "Main Menu". This includes access to the author/manuscript and reviewer files and the generation of various forms, reports, and labels, along with system diagnostics and utilities.

The software was designed to maximize efficiency and convenience for editorial office staff. It allows up to three separate editorial offices to share a single computer. Each office can manage a maximum of 25 different journals. Currently, ACS has two sites where two editorial offices share a single computer. In one of these instances, one office (the Manuscript Office in Washington, DC) has handled up to six journals on the system. Each editorial office can keep on file approximately 25 000 manuscript and reviewer files (this limitation is imposed by the hardware, specifically disk space and configuration, rather than the software).

The system provides rapid access to specific records. For example, in a reviewer file of 6300 records, a search for all reviewers with last names beginning with "Sm" afforded 48 hits in less than 1 s and for all reviewers whose last name begins with "J" 124 hits in approximately 1.5 s. Similar searches in a 6600 record reviewer file provided 48 names beginning with "Sm" in less than 3 s and 321 starting with "J" in 6 s. Once the desired record is found in the list, a single keystroke provides immediate access to that record.

File Structure. Files for Peer Review Plus include system control files, "pointer" files containing keys to free data records, main data files, and index files. The system control files contain operating parameters for a specific system. This includes the names of users, journals, editors, printers, terminals, and modems for each office in the system. System control files also contain flags that set specific user privileges and determine how certain programs can be run in each office. (This last feature was necessitated by the fact that several offices had different criteria for determining the steps in the manuscript process.)

The main data files are the heart of the file system. They are divided into author, reviewer, and common files. The

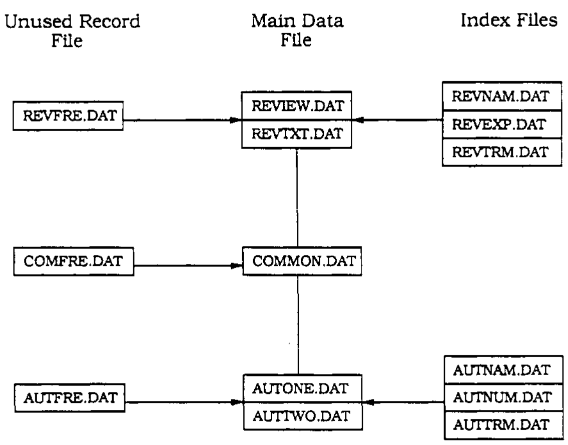

Figure 2. Schematic relationship of Peer Review Plus files.

common file (COMMON.DAT) contains data needed in both author and reviewer records. Storing the "common data" in a single file with pointers to the corresponding reviewer and author files saves disk space. A given author or reviewer record consists of two 512K blocks.[5] Each author/manuscript record consists of one block each in two files (AUTONE.DAT and AUTTWO.DAT). Reviewer records are also stored in two files (REVIEW.DAT and REVTXT.DAT). A common record is 64 bytes in size. The pointer files (REVFRE.DAT, COMFRE.DAT, and AUTFRE.DAT) keep track of which records in a particular file are available for use. The relationships between the unused record (pointer) files, main data files, and index files are illustrated in Figure 2.

Records are also indexed to provide rapid access to data on the basis of particular criteria. Reviewer files can be searched by reviewer name (one per record), expertise codes (up to 15 per record), index terms (up to 4 per record), or the unique reviewer number assigned to each reviewer. Expertise codes correspond with the reviewer's self-designated areas of expertise within the general field handled by the journal(s). These can be accessed by using the Boolean AND operator among areas of expertise and qualified with degrees of expertise to obtain increasingly narrow areas of specialization. Areas of expertise can be ranked from 1 to 9 depending on how proficient (expert) the reviewer is in a given field ("1" being the highest proficiency). The four reviewer "index" terms can be used to store any field an office might want to use for searching, such as the reviewer's employer, ZIP code, or key terms used to identify the reviewer. Similarly, author/manuscript search keys include author names (up to five authors per manuscript record), the manuscript number, and index terms (a maximum of six).

Automatic Telecommunications. Telecommunications are one of the added features provided with Peer Review Plus. This provides for automatic updating of data and electronic mail at all offices of a multioffice journal. The office of the editor in chief is considered the "host office". The associate editorial offices are referred to as "remotes". Each night, the host site calls all the remotes and polls them for changes in their author and reviewer data files. Once all sites have been polled and data from each site has been processed, the host machine again calls each remote site and transmits all the data collected. Consequently, each day all the editorial offices for a journal have on their local computers all the data changes performed in *any* office (host or remote) for that journal from the previous day.

If telecommunications are unsuccessful to one site, the changes are stored and transmitted during the next successful telephone call. All this occurs automatically late at night when telephone rates are at their lowest. Transmission times depend on the amount of data being transmitted; using 1200-baud

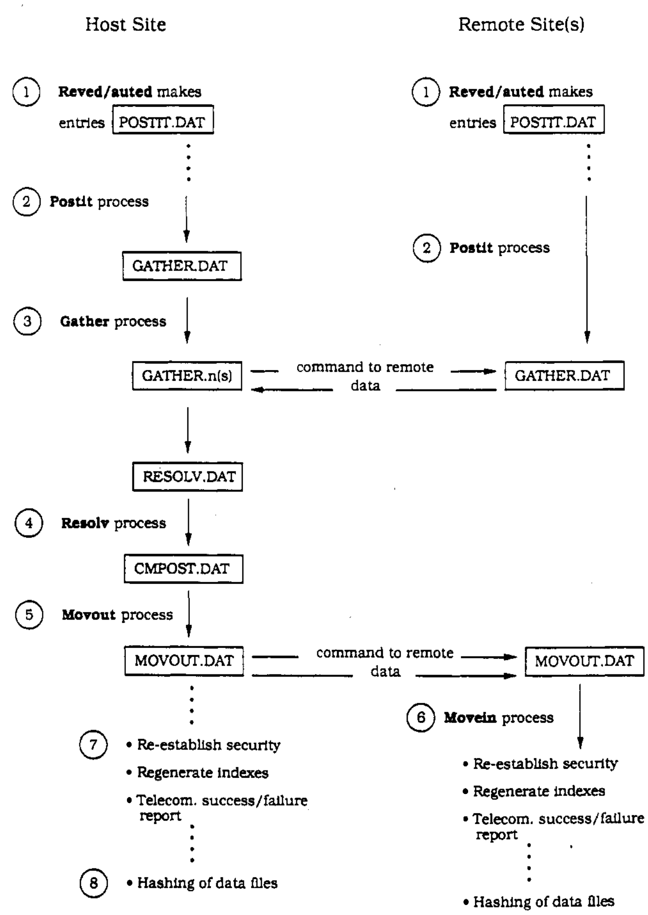

Figure 3. Automatic telecommunications flow diagram.

modems, approximately 10 512K blocks can be transmitted per minute.[6] Calls to systems that have no data to transmit can pass through security, determine there are no data, and reinvoke security in less than 1 min. When the data are accumulated at the host prior to being transmitted back to the remotes, they are algorithmically checked to see if any specific data field has been changed by two or more offices during that day. Such a change is designated a "deadly embrace", and the host office is notified of these occurrences via an electronic mail message conveying the particulars to a designated individual for manual resolution.

If the reviewer or author/manuscript files could be transmitted as complete files, the tracking and transmission of information changes would be relatively simple. However, these files are typically 5–15 megabytes in size and each would require 16–48 h per site to transmit. This is clearly impractical and would be prohibitively expensive even at transmission rates faster than the 1200 baud currently used. Consequently, we have developed a method of transaction processing in which only those data fields that are changed in the files are transmitted. This requires quite extensive monitoring and processing. Moreover, security is an important consideration, and this factor must be taken into account. Since the automatic telecommunications capabilities are one of Peer Review Plus's more interesting features, they are described here in some detail. The following eight steps are shown schematically in Figure 3.

1. Whenever a data element is added to or changed (including the addition of completely new records) in the reviewer or author/manuscript files, several elements must be tracked and monitored: the file, record, and specific data element in which the data are stored along with the date the change or addition was made. These transactions are saved on the system where the change is made in a transaction log file called POSTIT.DAT.[7] POSTIT.DAT itself does not contain the value for the data elements, but rather a pointer to the reviewer or author/manuscript file in which the data are written and stored.

2. At an established time, a program (POSTIT.RUN) is run automatically in background and creates a file called GATHER.DAT which contains the information in POSTIT.DAT plus the actual data from the reviewer, author/manuscript, or common file(s). Extraneous data, such as multiple changes in the same data field, are eliminated, and error checks are made to validate reasonable values for the data. This activity occurs at remote and host sites.

3. The host system then calls each of the remotes by executing the program GATHER.RUN in background, clears security (a double-password system), and "pulls over" the file GATHER.DAT from each remote system. The data file is renamed on the host to GATHER.*n*, where *n* is the number of the system (0 for host, 1 for first remote, 2 for second remote, etc.). If the phone link cannot be established or the file cannot be downloaded error free (noisy phone lines, etc.), the error condition is noted and three attempts are made to make contact and transmit the file. All transmissions of files are done with hash checking to assure error-free transmission. After GATHER.RUN finishes retrieving the GATHER.DAT file, security is reestablished and the phone connection is broken. After all the sites have been polled, the GATHER.*n* files are appended by GATHER.RUN into the file RESOLV.DAT. RESOLV.DAT is sorted on two keys: (a) the record identification (which includes identification of the file, the record number, and the field number) and (b) the date/time.

4. RESOLV.DAT is then scanned by RESOLV.RUN for redundancies and deadly embraces. Redundant data, multiple transactions on the same data element that have data of identical values, are eliminated. Notification of the deadly embrace(s) is sent to a designated individual on the host system to determine the correct value for the element(s) in question. The most recent value of the data element in question is passed through to all the systems.

After the processing of RESOLV.DAT has been completed, the data (values for specific data elements, such as reviewer's last name and data manuscript accepted) are written to the appropriate reviewer and author/manuscript files on the host. Entries are also written to a file called CMPOST.DAT which contains pointers to the reviewer and author/manuscript files and a bitmap variable indicating to which system these changes have been successfully transmitted.

5. At this point, a program called MOVOUT.RUN is executed which scans the entries in CMPOST.DAT and creates a data file called MOVOUT.DAT. MOVOUT.DAT, a unique data set for each site, contains the actual data values and pointers to where the data belong in the main data files. MOVOUT.RUN then calls the appropriate remote site(s), clears security, transmits the file MOVOUT.DAT to the remote(s) where it is called MOVEIN.DAT), and invokes a program called MOVEIN.RUN on the remote(s).

It is possible for a command, generated on the host to execute a process on a remote, to get garbled in transmission, resulting in that process not being executed on the remote. To guard against this possibility, data such as MOVEIN.DAT are not erased on the remote site until MOVEIN.RUN has successfully executed on the remote system. Subsequent MOVOUT.DAT files transmitted from the host are appended to the existing MOVEIN.DAT files on the remote system. Error-free transmission of *files*, however, is either a success or a failure, and this condition is immediately known to the appropriate program on the host.

6. MOVEIN.RUN loads the data into the main data files on the remote site(s) and causes the indexes to be regenerated on the remote(s). After the host system (via MOVOUT.RUN) invokes MOVEIN.RUN on the remote(s), security is reestablished and the phone connection is broken. MOVEIN.RUN on the remote(s) is done as a background job. If all three attempts to a given site fail, the data are saved for the next telecommunication session. Data successfully transmitted to a particular site are marked as "sent" in the bitmap variable in CMPOST.DAT. When data have been sent to all sites, the references are deleted from CMPOST.DAT.

7. Once the final telephone call is completed, security is reestablished on the host and the indexes are regenerated. The success or failure for automatic transmission of data is reported via electronic mail to a designated individual on both host and remote systems. On the host system, a detailed log (called UPDATE.LOG) of the entire process is maintained and examined daily by R&D staff in Washington. UPDATE.LOG contains such information as (a) when (date and time) each step occurred, (b) how long the process took, (c) error messages generated by the programs, (d) number of "information packets"[8] that required retransmission, and (e) number of attempts required to make phone connections and/or send or receive files. By monitoring this file, R&D staff can focus on sites not communicating properly and determine what is causing the problem.

8. As part of our quality control to assure that the main data files on each system in the network are identical, a multilevel hash for the author/manuscript, reviewer, and common files is generated for each record in these files on every system. These processes occur daily in background. R&D staff routinely compare the hash files to be sure that data are correctly being transmitted. Occasionally, editorial staff will question whether a particular piece of data is valid or has been transmitted. The hash files permit easy validation and/or retransmission of questionable data. (This rigorous examination of data integrity has, on occasion, permitted R&D staff to diagnose hardware problems prior to the editorial staff being aware of any difficulty.)

Electronic Mail. Messages up to 60 lines in length can be sent to users on the system. A given message can be sent to a specific person, several people, or all users for any particular office. Messages can be edited while being composed in a manner similar to a word-processing file. If the message is for someone who does not reside on the local system (for example, a message from the host to a remote), that message will be automatically transmitted overnight. Word-processing files can also be sent as a mail message. Consequently, copies of letters can be sent electronically from one site to another without the need to rekey the document.

Each user on a system (either host or remote) has a unique name (logon identification). Associated with that name is a flag which identifies that user as being either a local user or a remote user and the identification of the specific site if a remote. If the recipient is local, the message is immediately posted to that user; if the recipient is remote, the message is stored for forwarding. All mail, except that for a local recipient, is routed through the host system; that is, mail from a user on a remote site to a user on a different remote site is routed through the host computer.

Prior to the automatic telecommunications for database information, a program (called SRMAIL.RUN) on the host machine is executed in background in much the same manner as previously described for GATHER.RUN. SRMAIL.RUN calls each of the remote systems and pulls over the mail file (called MAIL.DAT) which contains all messages generated by each site for nonlocal recipients. The MAIL.DAT file received is "unpacked", and messages destined for local recipients are immediately posted while messages destined for other remote systems are stored in a file called NODE1.DAT, NODE2.DAT, and NODE*n*.DAT, where *n* refers to a particular remote node. After all sites have been polled by the host to retrieve mail, each of the sites is telephoned again by the host and the appropriate NODE*n*.DAT file is transmitted to that node. When the remote node receives mail, a process on the remote machine is run in background to unpack the mail file received and send each user his or her messages. Incorrectly addressed mail (unidentifiable) is sent to a "dead letter" file that is routed to a specified person on the machine which is unable to identify the recipient. At the present time, electronic mail is transmitted once a day, generally about 9:00 p.m. local time (host). The capability exists to transmit mail more frequently if justified.

Reports Generated. One of the advantages of the Peer Review Plus software is the wide variety of reports, forms, and labels that can be produced from the main menu by using the database. Programs used to generate information concerning the manuscript/author file include the following.

EDLIST compiles a list of new manuscripts received, journal by journal or editor by editor, within a specified time period. This program would generally be run to determine how many new manuscripts are flowing into the office and which editors are handling them.

MSSTAT gathers statistical information regarding acceptance, rejection, and other status changes within a selected time period. This is often used on a monthly basis to tally up manuscripts under the following categories: received, resubmitted or reactivated, inactivated or withdrawn, accepted, rejected, manuscripts in process, and manuscripts sent to the publication office. Information is listed for the specified manuscript type(s) for the selected period, year-to-date, and the previous year.

EDSTAT gathers monthly statistics for various manuscript types (regular papers, communications, notes, etc.) for each editor. The program allows for up to 50 editors and 26 types of manuscripts. Output options include year-to-date and current tallies for manuscript status (accepted, rejected, withdrawn, etc.) for all editors or one editor and any or all manuscript types.

PROTIM compiles the average amount of time taken to handle manuscripts by a given editor.

MSTIME compiles the length of time required for each manuscript to complete various steps in the review process.

MEMOED automatically generates memos to editors containing a listing of the manuscripts enclosed in a package.

MSPROC determines the bottleneck or holdup for manuscripts currently in process within a specific office.

LORDER generates a list of manuscripts that have been accepted but have not been published. Such identification provides a "safety net" to catch manuscripts that staff might overlook.

ABSURD analyzes inconsistent data within the data files. For example, manuscripts that have been sent out to reviewers before being received or manuscripts accepted for publication before being returned by reviewers. In these examples, implausible dates have been entered. More sophisticated checks are made to identify "absurd data".

SCREEN produces a paper copy of a manuscript (or reviewer) record as it appears on the screen. The user has an option to print the entire record or just a portion of the information.

Reviewer reports include the following.

OVERDU lists manuscripts that have been out for review longer than a time period specified by the user. The lists can be generated in three formats: a size that will fit on labels that can then be affixed to post cards and sent as a reminder, a single-page printout that can be mailed to the tardy reviewer, or a list with names and phone numbers if the editorial staff prefers prodding the reviewers by telephone.

REVUSE lists all reviewers on the system who have been used (or not used) during a time period determined by the user. If there is more than one journal for the editorial office, the user can select which journal(s) the output should include.

REVEXP provides an opportunity for the user to display, edit, add, delete, or print out the areas of expertise currently on file for the editorial office.

SCREEN prints paper copies of reviewer (or manuscript) records. Either the entire record or just the first two (of six) pages can be printed.

PRNEXP provides hardcopy printed outputs listing (a) the first (and presumably most important) expertise code for every reviewer on the system or (b) a screen dump of all the reviewers found under a given search by expertise codes. The first format is a means of printing a list of all the reviewers in the database. The second option provides an output that can be useful in determining which reviewers are qualified to review a certain manuscript under consideration.

Various forms and labels can be printed including the following.

COPYRT retrieves data for preprinted copyright transfer or receipt of manuscript forms. Information for the copyright transfer can be edited prior to printing the form to allow the addition of any authors or part of the title that was not included in the database due to space restrictions. As well as the form option, acknowledgment of receipt of a manuscript may be printed on label stock to affix to preprinted forms or post cards, or it may be printed on paper and mailed with window envelopes. The acknowledgment notice includes the date the manuscript was received, the name of the journal, the name(s) of corresponding author(s), the manuscript number, and the title.

AUTFRM generates numerous label outputs including mailing labels for reviewers and authors, manuscript folder labels, and labels to notify the author that the manuscript has been accepted. A manuscript transmittal form can also be generated to send information to the publisher regarding an newly accepted manuscript.

REVLAB generates mailing labels for one, several, or all reviewers in the database. If labels are desired for several reviewers, they can be done sequentially (for example, reviewers 1234–1240) or randomly as a number of individual reviewers (1233, 2011, 1428, 2177). The labels can be sorted alphabetically, by ZIP code, or by reviewer number. A TEST option is available to make sure that the label stock is aligned properly before large quantities of labels are printed.

Maintenance Utilities. Routine maintenance required of the editorial office staff can be initiated from the Main Menu. This includes setting the system time and date, creating and certifying system backup tapes, unmounting the disks prior to shutting off the power, and calling another computer (exclusive of automatic telecommunications). Since there is no simple method to expanding the capacity of the author or reviewer files, when they approach capacity, old, inactive records need to be deleted to make room for new ones. Prior to deleting old records, office staff have the option to archive the records to paper. More technical diagnostics, such as recreating the system indexes, displaying the index counters, and checking the integrity of the data files, can also be handled from the Main Menu.

FUTURE OF PEER REVIEW PLUS

The ACS recently converted the Peer Review Plus system, which has been dependent upon Alpha Micro hardware, to run on UNIX-based systems. We selected the NCR Tower series of computers, which operate on the UNIX System V operating system, as the target machine. Since the converted source code is in the C programming language, it is expected that it would be relatively easy to port the software to a wide variety of systems. There are several reasons why this conversion was done:

• To attain hardware independence. Code written in ALPHA BASIC for the Alpha Micro cannot be used on other hardware.

• To permit the use of much more sophisticated word-processing software. The current Alpha Micro systems use word-processing packages that are approximately 10 years old in design, and output is limited to the ASCII character set. It is desirable to permit such document preparation capabilities as generation of mathematical expressions, chemical structures, and tables.

• To improve security. Alpha Micro systems are particularly weak in this area. Although we have had no problems in this regard, improvements would be a wise precaution. The first UNIX-based system was installed for *Analytical Chemistry* in August 1988, followed by a UNIX installation for *Biochemistry* in November 1988. Replacement of all Alpha Micro systems is expected to require 3–4 years.

We have received numerous inquiries about making Peer Review Plus available on personal computers, in particular for IBM PCs and clones and on Apple's Macintosh systems. Because Peer Review Plus software requires a multiuser (the system can be used by more than one user at a time) and multitasking (multiple processes can be simultaneously executed) operating environment—as well as the capability to execute programs in background at predetermined dates and times—these personal computers are not suitable. Moreover, the Peer Review Plus software is relatively complex in operation, in particular the telecommunications aspects, which requires that it be routinely monitored by professional computer staff. Also, we have found that in excess of 100 megabytes of disk capacity is needed to store data for an average size journal. Such storage capacity has only recently become available on personal computers. At such time when personal computers have these capabilities, which may happen with machines using CPUs based on the Intel 80386 or Motorola 680x0 series processors, we will carefully examine the possibility of making our software available on a wider selection of computers using the UNIX operating system.

REFERENCES AND NOTES

(1) "Edge-notched" cards have sequentially numbered positions marked around the perimeter of the card which can be punched out to indicate a particularly defined characteristic. One card is used for each person (record)—reviewer, author, manuscript, etc. Each hole position is assigned a particular attribute (e.g., position 1 = nucleic acid, position 2 = NMR). If a card is to be classified with a particular attribute, the appropriate hole is notched at the top of the card. To retrieve cards with a specific attribute, the cards are carefully aligned and a "knitting needle" is inserted through the appropriate hole. Cards notched in that position fall out of the stack and are thus retrieved.

(2) Garson, L. R. Computer-aided Selection of Reviewers and Manuscript Control. *Scholarly Publishing*, **1980**(Oct), 65–74.

(3) Neufell, J. L. Personal communication. Systems Analyst, *New England Journal of Medicine*, Waltham, MA, March 17, 1988.

(4) Rice, C. Personal communication. Director of Editorial Office Services, American Physical Society, Ridge, NY, Aug 30, 1989.

(5) The AMOS (Alpha Micro Operating System) operating system imposes a maximum of 512 bytes per record. Consequently, two physical records had to be designated for each logical record.

(6) A 'K' is a unit of 1024 characters, which is 2^{10}. If telecommunications were completely error free, that is, some portions of data did not have to be retransmitted due to errors, one could expect slightly more than 14 512K blocks to be transmitted each minute.

(7) File names on Alpha Micro systems can have a maximum of six characters, followed by a period, followed by an extension of three more characters. By convention, executable files written in BASIC have an extension of RUN (POSTIT.RUN, for example). BASIC source code files have the extension BAS (for example, POSTIT.BAS). Files containing data have the extension DAT (GATHER.DAT).

(8) Information packets consist of 256 bytes of data along with a hash value which are transmitted from one site to the other. The receiving site recalculates the hash value for the data received and compares it to the tranmitted hash value. If the two have identical values, the receiving system transmits a signal to the sending system to send the next packet. If the hash values are not identical, a signal is sent to retransmit the previous packet. The specific communication routines used in our system, written in Assembly language, were developed by Michael Knolls of Nome, AK, under contract to the American Chemical Society. For more details on telecommunication protocols, see reference 9.

(9) Campbell, J. *C Programmer's Guide to Serial Communications*; Howard W. Sams: Indianapolis, IN, 1987.

Using Chemical Bonds To Analyze Data Retrieved from the Inorganic Crystal Structure Database

I. D. BROWN

Institute for Materials Research, McMaster University, Hamilton, Ontario, Canada L8S 4M1

Received August 9, 1989

The programs SINDBAD (batch) and STRUMO (interactive) use a chemical bond model to examine structures retrieved from the Inorganic Crystal Structure Database. The bonding topology at any desired level of bond strength is determined from the bonds (and their valences) calculated directly from the atomic coordinates stored in the database. Hydrogen bonding, disorder, internal stress, defects, and diffusion paths can be examined and modeled by using related techniques.

(1) INTRODUCTION

The four crystal structure databases[1] covering the fields of metals and inorganic, organic, and macromolecular crystals contain data for the 100 000 crystals whose structures are currently known. The data are stored in the form of atomic coordinates, lattice dimensions, and symmetry operators, information that allows one to find the location of all atoms in the crystal. Chemists are more interested in the way atoms bond together, and an array of atoms, whether described by a list of coordinates or displayed on a screen, does not directly provide this information. To appreciate the chemical properties of the structure, one must know something about the sizes of atoms and the locations of the chemical bonds.

Chemical descriptions of structure are usually based on empirical or highly simplified physical models. Several such models, each with its own advantages and disadvantages, are currently used by solid-state chemists to describe the structures of inorganic compounds. There are models based on chemical bonds and others on the close packing of anions, models that emphasize symmetry relations or that look for structural similarities, and models that use interatomic potentials. Any program to transform the coordinates in the database into

0095-2338/89/1629-0266$01.50/0