How to Edit a Journal by E-mail

Andrew W. Appel
Department of Computer Science
Princeton University
`appel@princeton.edu`

CS-TR-494-95

July 10, 1995

Abstract

The editor of a refereed scientific journal must somehow get unpaid referees and unpaid associate editors to provide prompt evaluations of technical papers. The use of electronic mail, with a well organized management system—and the software and staff to support it—make this possible, without unduly burdening either the editor, associate editors, or referees.

Some refereed scientific journals have long delays from submission to publication. Others achieve fast turnaround at the cost of burdening their editors and associate editors with a heavy workload.

I have had an opportunity to observe the operation of several journals, some from the vantage point of author, others as associate editor, and one as editor in chief. I have also talked to the editors and production managers of journals in other disciplines. Since my appointment as editor in chief in December 1992, I have run TOPLAS (*ACM Transactions on Programming Languages and Systems*, published by the Association for Computing Machinery) according to a well organized system that I put together by borrowing the best ideas from as many places as possible. The result is that submissions are refereed quickly without taking too much of my time or the associate editors' time. I can use the time saved to work on editorial policy matters, negotiate page budgets with the publisher, handle special cases, discuss specific editorial decisions with associate editors, and (sometimes) even do my own research.

The fundamental principles of this system are:

- Associate editors and referees are happy to provide their scientific judgement and expertise; but they do not want to spend their time on clerical or organizational tasks. Associate editors should *not* be responsible for remembering who owes referee reports by what date, reminding referees, formatting e-mail messages, copying and printing papers. They should just respond to each e-mail message sent them, and not have other worries.
- There should be one single person (a *traffic manager*) whose job is to ensure that the participants adhere to schedule. This task should *not* be fragmented among all the Associate Editors' secretaries, for none of whom is it a high priority.
- Electronic mail can replace all paper communications, and electronic databases can replace all paper files.

The participants in the editorial process are:

EIC Editor in chief.

TM Traffic Manager: the EIC's editorial assistant. This system requires a very competent traffic manager who can handle routine correspondence independently and will refer any unusual situations to the EIC.

To use the software system I have devised, the TM must be reasonably computer literate and have some familiarity with Unix. Facility with the *Xmh* mail system[2] (a window-oriented version of the Unix MH mail reader) is necessary, but can be learned easily on the job. Knowledge of common Unix tools such as *compress, uncompress, ghostview, tar, gzip* is useful. If the software were to be implemented on another platform, such as Microsoft Windows, then Unix knowledge would not be necessary.

The TM should have diplomatic personality, to remind referees repeatedly without arousing resentment. Familiarity with the journal's scientific subject matter is not necessary, but familiarity with journal protocol is useful.

For TOPLAS, which receives about 140 submissions per year, the TM is a half-time technical secretary.

AE Associate Editors (TOPLAS has eight to ten). AE's must be knowledgeable in their research fields so that they can judge papers and choose referees, but (and this is very important) *they need not be well organized*—the TM reminds them what needs to be done next, and all they need to do is respond to the TM's messages.

Referee The referees should be accessible by e-mail; fortunately, in the field served by TOPLAS, over 99% of them are on the Internet. The others can be accommodated (through U.S. mail) with some extra work.

Author Most TOPLAS authors are on the Internet and submit papers (in Postscript® format) by e-mail.

ME Managing Editor: supervises the production (typesetting, copy editing, printing) of accepted papers. For TOPLAS the ME is on the staff in the Journals department at ACM headquarters.

E-mail tracking database

The Traffic Manager uses a tracking system integrated with the electronic mail sender/reader to keep track of where each paper stands and what needs to be done to it. Each outgoing e-mail message is annotated with a special action keyword, such as: `Action: query`.

The Internet mail system ignores the Action header line; but a blind carbon copy of each outgoing message, with its Action header, is kept in the database.

Incoming messages will not have "Action" headers, because outside authors, referees, and associate editors won't put them there. Instead, the TM will manually annotate each incoming message to add the appropriate Action header.

By examining the Action line of each saved message, the software can tell what needs to be done next. For example, if one (outgoing) message is annotated:

```
Action: query
To: dobchinksy@cs.isu.edu
 . . .
```

and there is no later message of the form

```
Action: answer
From: dobchinksy@...
 . . .
```

then the tracking software can alert the TM to prod Dobchinsky for a reply. In this case, the TM has written "Action: query" into the message sent to Dobchinksy; Dobchinksy will reply without using an Action header, and the TM will insert the "Action: answer" line into Dobchinksy's response using the e-mail system's text editor.

The Xmh mail reader is used as a message database. Xmh is a window-oriented interface to the MH e-mail filing system, and in fact any interface to MH could be used. Xmh (or MH) organizes mail messages by "folders"; each folder contains mail messages numbered $1, 2, 3, \ldots$ and may also contain subfolders. Our folders are:

inbox in which incoming messages land;

drafts in which outgoing messages are kept before they're sent;

forms containing form letters that can be used for routine correspondence; there are subfolders:

- **author** form letters sent to authors,
- **editor** form letters sent to associate editors,
- **referee** form letters sent to referees,
- **acm** form letters sent to the journal production department at ACM.

papers This folder contains one subfolder for each submitted paper. The name of the subfolder for paper #1234 is `p1234` (this its full name is `papers/p1234`). Thus, `papers` contains hundreds or thousands of subfolders, only a few dozen of which are in active use at once.

p169, p170, p171 Xmh can accommodate a folder (such as **papers**) with hundreds of subfolders (such as **p1001**, **p1002**, ...), but the menu-oriented interface has trouble displaying a menu with hundreds of items. To work around this problem, we create special "virtual" folders containing ten papers each. The virtual folder `p169` holds papers numbered 1690 through 1699. We keep only those virtual folders necessary to access the active papers; at any time we might have twenty or thirty virtual folders of ten papers each.

acm Correspondence with the ACM journals department.

solicit Sometimes the journal solicits specific papers from potential authors. We keep track of the relevant correspondence in this folder.

eds Correspondence (not related to any specific paper) with associate editors.

misc Miscellaneous.

Each outgoing e-mail message must have headers labelled Action, To, Subject, and Fcc (folder carbon copy). Here's an example:

```
Action:  query
To:  John.Mitchell@cs.stanford.edu
Subject:  TOPLAS 1335
Fcc:  papers/p1335
```

Dear John: Enclosed is a new paper,
NUMBER 1335
TITLE Circular Reflection
AUTHOR Rebecca L. Weir
Please tell me who should referee it.

```
-----------------cut here-------------
%!PS-Adobe
```

... this is the Postscript file ...

The Subject line contains the paper number so that when the reply is received, its subject line will (very likely) also contain the paper number. This saves TM the step of looking it up.

The Fcc line keeps a blind carbon copy in the folder `papers/p1234`.

The Action line doesn't do anything when the mail is sent, but the presence of the Action line in the folder carbon copy allows the tracking system to see what's going on.

One manuscript's history

I will explain the system by following the processing of a single TOPLAS paper from submission to publication. Each step is labelled by its Action keyword. The database tracking software users these keywords to follow the interactions between TM and editors, referees, and authors, according to the state diagrams of Figures 1–3.

- The paper is submitted: the Author sends it in Postscript form, by electronic mail, to the journal's e-mail address (which should be distinct from EIC's personal e-mail address, so that journal-related e-mail can be processed by TM). This incoming message is not labelled with an Action keyword, since the author isn't expected to know about actions.

receipt TM sends e-mail to the author acknowledging receipt; this message includes information about copyright assignment, submission number for future reference, and so on.

receipt-q When a submission is on paper, TM sends e-mail to the author requesting an electronic copy, at the same time acknowledging receipt and sending information about copyright assignment. This message is really just the combination of a **receipt** message and a **query** message.

If the author cannot send Postscript, then the submission is handled by old-fashioned methods (U.S. mail); or, TM feeds it through a scanner, producing a Postscript file that can be sent through e-mail.

Some authors have difficulty including large Postscript files into their e-mail messages. They may instead Postscript by FTP (Internet File Transfer Protocol), by transferring their Postscript into the EIC's "incoming" FTP directory.

query TM sends e-mail to EIC. Before assigning the paper to an associate editor, the EIC may want to know which editors are overloaded and which editors don't have much to do. The *editors* command generates a listing of the recent papers (by month and year) assigned to each editor:

```
steffen    2/95  2/95  1/95  1/95 12/94 12/94
soffa      4/95 12/94 10/94  8/94  5/94  5/94
wing       3/95  2/95  2/95  1/95  3/95  3/95
mitchell   3/95  3/95  2/95  3/95  1/95 12/94
debray     3/95  3/95  2/95  1/95  1/95  1/95
davidson   4/95  2/95  2/95  1/95  1/95 11/94
kamin      3/95  3/95  1/95  1/95  1/95  1/95
gunter     4/95  3/95 12/93
ellis      4/95  3/95  3/95  2/95  1/95  1/95
```

Here we see that Debray has handled six papers since January '95, while Soffa has handled only four papers since August '94 (and Gunter is newly appointed to the position of AE).

answer EIC replies with a selection of AE to handle this paper.

query TM sends paper by e-mail to AE, requesting names of three referees plus an alternate.

remind If AE doesn't respond within four days, the software automatically tells the TM, and the TM sends a reminder to the AE.

abandon If the recipient of a query repeatedly fails to respond to reminders, the TM must abandon the query. Waiting forever will just delay refereeing of the journal submission. So, for example, if an AE must be "abandoned," TM notifies EIC, who chooses an alternate AE for the paper. Even the most responsible of AE's

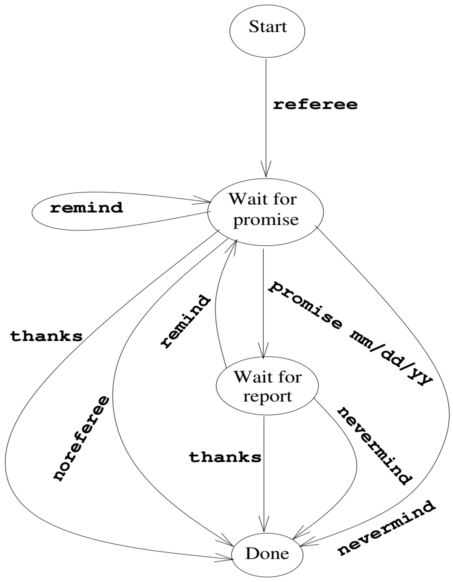

Figure 1: Interaction with referees

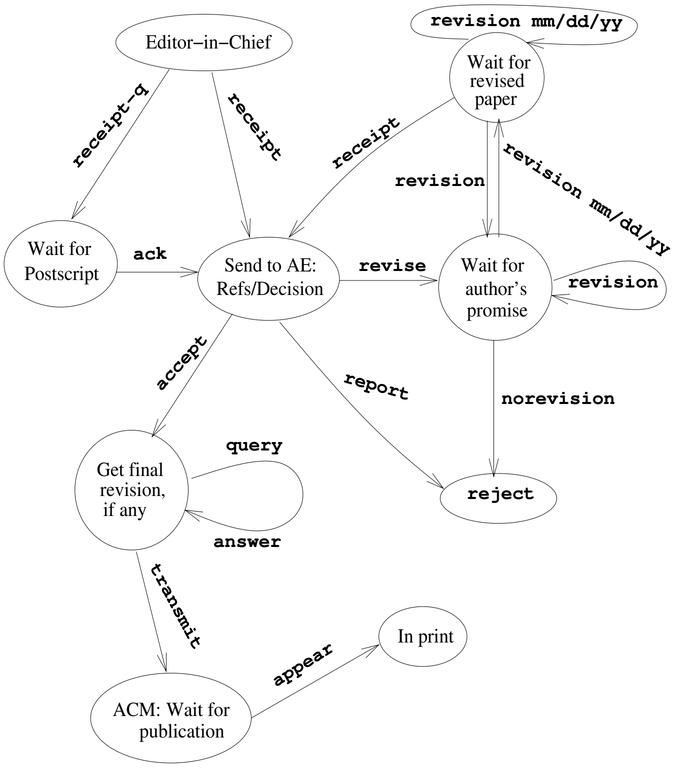

Figure 2: Interaction with authors

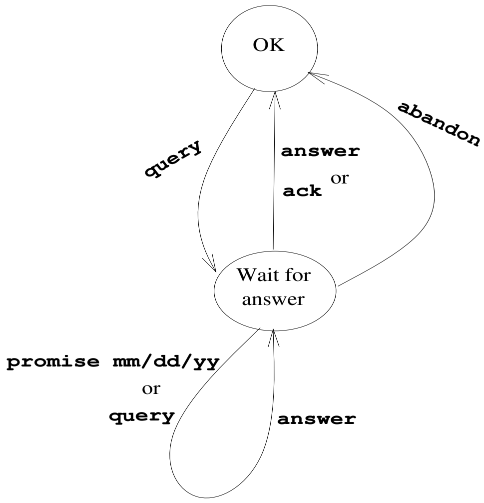

Figure 3: Interaction with associate editors and others

sometimes have busy times or fall ill. In this case the message from TM to AE is

```
Action: abandon Ima.Lout
To: Ima.Lout@cs.iru.edu
fcc: papers/p1225
Subject: TOPLAS 1225
```

Dear Dr. Lout:
I assume you must be busy, since you haven't responded to my query about paper 1225. The EIC has chosen another editor to handle this paper; you don't have to worry about it.

In general, any query may be abandoned this way. Sometimes one wants to abandon a query without sending mail to anybody in particular; this is easy; just send a message to "nobody" with a blind carbon copy for the tracking system:

```
Action: abandon Ima.Lout
To: nobody
fcc: papers/p1225
Subject: TOPLAS 1225
```

Giving up on Dr. Lout.

answer (assuming abandonment wasn't necessary) AE responds with names of referees.

TM checks referee database to see if these referees have recently done more than their share of refereeing; or are currently refereeing papers at the request of other AE's. If so, TM alerts AE to choose alternates.

referee TM sends paper by e-mail, with standard "referee request form letter," to three referees.

Letter says: "[Name of AE], Associate Editor of [Journal], has suggested that you would be a good referee for the enclosed paper; please respond within 1 week to say whether you can referee the paper within 3 or 4 weeks."

- Referee 1 responds saying "can't do it in 4 weeks, but I could in 6."

promise mm/dd/yy TM sends mail back saying six weeks is fine and thanks the referee for agreeing to referee; sets `mm/dd/yy` to the date six weeks hence in the Action field of the outgoing message.

- Referee 2 responds saying "too busy right now, but Nikto would be a good referee."

noreferee TM thanks Referee 2 for being prompt in responding.

query TM forwards Referee 2's suggestion to AE for consideration.

answer AE agrees, choosing Nikto as the new referee 2.

referee TM sends paper to Nikto, who agrees to referee by a certain date (TM then sends **promise** message acknowledging).

remind Referee 3 does not respond. TM sends a courteous reminder after a week.

nevermind Referee 3 still does not respond. TM sends message saying "never mind," and chooses an alternate referee 3 from AE's original list; the new referee 3 agrees to send a report within four weeks.

Four weeks pass.

remind Referee 3's four-week deadline arrives, with no report. TM sends mail reminding. Referee 3 responds with apologies, promises report in another week.

promise mm/dd/yy
TM acknowledges with a **promise** message with a date one week hence.

- Referee 1 sends report.

thanks TM sends letter of thanks.

remind Referee 2's deadline passes; TM sends reminder.

- Referee 3 sends report.

thanks TM sends letter of thanks.

Another week passes.

query Referee 2 is still not heard from. TM sends reports of referees 1 and 3 to AE, asks whether this is enough for a decision. The format for this letter is:

1. Paper number, title, and author; snatched (with mouse) from "status" command.
2. Names of referees, numerical ratings given by referees, other messages from referees for editors but not authors.
3. Anonymous referee reports.

It's important for the TM to make sure that part 3 does not contain any material that would compromise the anonymity of referees, because this part will be included in the subsequent decision letter to the author.

In the message to AE, TM also includes the output of the **editor** command, which lists all the papers that editor has handled and which of them were accepted. This gives the AE a context for ranking the quality of this paper relative to others he or she has seen.

answer AE response with a letter of decision ("reject") along with one or two paragraphs summarizing the referee reports and explaining the decision.

report TM prepares a letter to the author rejecting the paper, with referee reports and AE's explanation. The keyword **report** is a euphemism for "reject;" keywords appear in the outgoing mail, and we want to be polite.

On the other hand, perhaps the paper is a good one:

answer AE responds with letter of decision ("accept with revisions"), along with a one or two paragraph summary of reports and instructions for authors, explaining which of the revisions demanded by referees are important to the AE and which are not.

nevermind TM sends "never mind" letter to referee 2.

- TM prepares a draft of the letter to Author (by e-mail, as usual), incorporating referee reports and paragraph from AE. TM sends e-mail to EIC, who reviews and edits the draft. (EIC also checks to make sure that the backlog is not getting too large, in which case standards might have to be raised and the paper rejected. In this case EIC discusses the paper with AE by e-mail.)

Asking for revisions of a paper is a major decision, since it implies that as long as the authors make the requested changes then the paper will presumably be accepted for publication. Therefore, the EIC reviews every such decision.

revise TM sends letter to Author. Letter includes sentence "respond immediately to say when you will send your revised paper." Also included are instructions on obtaining a LaTeX style file mimicking the TOPLAS format, for optional use by the author in preparing camera-ready copy.

A week passes.

revision TM sends reminder to Author, saying "we need to know when you will send your revised paper."

revision mm/dd/yy
Author responds, saying "two months." TM annotates the incoming message with **revision mm/dd/yy** where the date is two months hence.

Two months pass.

revision TM sends reminder to Author, saying "You promised to send a revised paper."

revision mm/dd/yy
Author responds, saying "two weeks." TM annotates the incoming message with **revision mm/dd/yy** where the date is two weeks hence.

A week passes.

receipt Author sends revised paper. TM sends acknowledgment of receipt.

query TM sends paper to AE, asks which referees should re-review it.

answer AE responds, "Author has addressed concerns of referee 3, but send it back to referee 1."

referee TM sends paper to referee 1, saying "respond within a week to tell me how many weeks you need."

promise mm/dd/yy Referee 1 responds saying "three weeks."

- Referee 1 sends report.

thanks TM sends letter of thanks.

query TM sends report to AE.

answer AE responds with "accept" decision, along with paragraph of explanation.

accept TM sends letter to author, incorporating referees' reports and AE's paragraph.

Here the EIC is usually not involved, since the important decision (to ask the author to revise the paper) was made much earlier.

- Author prepares final revisions of accepted paper, sends it back.

receipt TM acknowledges.

query TM sends paper to AE for final once-over;

answer AE responds "OK."

transmit TM sends accepted paper to ME, with a cover letter listing names, addresses, e-mail addresses, dates of submission and acceptance, and other supporting information. Now the paper has to go through the production process.

If it's done camera-ready, ME will correspond by e-mail with author (with cc: to EIC/TM); if it's done the old-fashioned way, ME will correspond by physical mail with author.

The production process (copy editing, typesetting, proofreading, scheduling, compilation of table-of-contents, page proofs, etc.) is beyond the scope of my "referee management system."

appear mm/dd/yy
Finally, ME notifies EIC/TM when an issue dated mm/dd/yy is released to the printer; TM annotates the message and copies it to the subfolder of each paper appearing in the issue.

Figure 1 shows the interaction with referees as a finite-state diagram; Figure 2 shows the interaction with authors; and Figure 3 shows the interaction with associate editors.

Throughout this process, either the TM has e-mail waiting in the *inbox* to be processed, or the TM is awaiting a response from an Editor, Referee, or Author, or both. The electronic processing system is set up so that whenever the TM is awaiting a response, the system knows about it; and when the response is late, the system alerts the TM to send a reminder or "abandon" the respondent. Thus, papers don't get delayed because of a single perpetually late referee or AE.

Software

I have implemented a software system to handle this process. It must be regarded as a prototype, and I am not sure that the prototype system is sufficiently robust and well documented that others could use it. However, it has been extremely useful and effective for me and my TM.

Since almost every "transaction" is associated with an e-mail message, and most of the e-mail messages cause state changes in the systems, it makes sense to have the e-mail file and the tracking database be one and the same.

The Xmh mail system stores all the e-mail (and papers in Postscript format, which arrive in e-mail messages). Xmh (and its cousins MH and Emacs-MH, which store data identically) has the virtue of using an extremely simple and program-friendly format for storing messages and folders. Each folder is just a Unix directory, and each message is just a file. Thus, it is easy to implement the tracking software that reads all the mail message and interprets the mail headers.

So, I make a directory (folder) **inbox**; and a diretory (folder) **papers** with subdirectories (subfolders) **p1000, p1001, p1002, ...** for each of the submissions numbered 1000, 1001, etc. Each papers's folder has one dummy message (message number 1) listing the Author(s), Author-Address, Title, Associate Editor, Length (in pages). The remainder of the entries in the folder are just the incoming and outgoing e-mail messages themselves.

The database software scans all the folders and messages, looking at the *Date, To, From,* and *Action* fields of each message. From these it can calculate who is late in responding to queries; it makes a transaction report of which messages in which folders need action (reminder, or abandonment of correspondent, etc.).

The TM's primary job, then, is to clear the **inbox** and take appropriate action for each line of the transaction report.

Other software scans all the folders and messages to compile other reports:

transaction report
Who owes us what.

```
p1370:88     8 days since query to ellis
p1375:72     7 days since referee request to jones
p1375:73     7 days since referee request to nikto
p1375:71     7 days since referee request to wilde
p1575:86     1 days since promise from weir
p1624:25     7 days since referee reminder to gonzales
p1671:56     8 days since referee request to peter
p1690:33     7 days since referee reminder to ruslan
p1699:20    15 days since referee reminder to pch
p1699:24     8 days since referee reminder to lavalle
p1704:24    11 days since query to kamin
p1707:16     7 days since referee request to hector
p1712:39     8 days since referee reminder to stein
p1715:7      7 days since query to ellis
misc:249    73 days since query to fischer
misc:262    53 days since query to nhora
acm:556     24 days since query to criscione
solicit:49  10 days since promise from krause
```

The notation **p1690:33** indicates that message 33 in folder **p1690** has not been responded to.

summary
Status (one-line summary) of every paper; estimated total pages of editorial queue and publication queue; etc.

quarterly report
A report suitable for e-mail to ACM. This includes statistics about volume of submissions, size of editorial queue, and (important!) a list of papers that EIC believes are in production at ACM. This avoids the problem of papers "falling through the cracks" of the production process.

referee *jones*
Shows the history of papers that Jones has been asked to referee, and shows whether he refereed them and how quickly he did so. It is helpful in achieving an equitable distribution of referee requests. If an AE chooses a referee who has just finished refereeing a paper for one of the other AE's, the TM can suggest that the AE choose someone else.

status 1234
Summarizes the history of paper #1234: title, authors, length, date of submission (and resubmission), names and e-mail addresses of referees (and dates they sent referee reports), date of editorial decisions, and a list of what responses are owed by correspondents to the TM.

look *word*
Searches the database of all papers, and find every occurrence of *word* in any paper title, author name, editor name, etc.; and show the paper number, authors, and title for each occurrence.

editor *Mitchell*
List all the papers (author, title, number) that associate editor Mitchell has ever handled, categorized by accepted, rejected, or still-being-refereed.

editors
Show how many papers each editor has recently been assigned (the output of this command was illustrated earlier).

I have also implemented report software to tell me which associate editors respond the fastest, the average number of pages per paper, and how papers are refereed (on the average), and so on.

There are several other commands and software utilities used to run this system: a command to bundle up Postscript file for transmission to the production department, a facility to send U.S. mail letters but record them in the e-mail database, and so on.

All this software is written in AWK (a Unix string-processing language), and is pretty horrible to look at. However, since it only reads—never writes—the e-mail database, bugs in the software can't corrupt the data. All modifications of the database are done by the Xmh software, which is (in principle) reliable.

This system takes some skill and training to use, particularly in the choice of Action keywords and the mechanics of using Xmh to refile messages. Bad choice of Action keywords can lead to papers falling through the cracks. (In that case, what usually happens is that after a few months the author inquires, and the problem is fixed. However, this is not a good way to run a journal, and fortunately such events are rare with this system.) An industrial-strength reimplementation of this software would fix these problems.

Production

After papers are accepted, they go to the publisher (in this case, the ACM Publications department) for production. The "old style" of production took several months, with several weeks for each of the following tasks, performed or assigned by the Managing Editor:

- copy editing
- composition, galley proofs
- in-house proofreading
- author proofreading
- to editor for table-of-contents ordering
- compositor again (corrections), whole-issue page proofs
- checking page proofs, release by managing editor
- printing, binding, mailing

At every stage of this process, U.S. mail was involved (the managing editor is in a different city from the compositor), and at every stage the papers would wait in a work queue for processing. In the new world of electronic composition, global networking, and electronic dissemination [1], things should be much faster. I will not address these issues here.

Whether in "old" or "new" style of editing and production, there is a certain comfort in having a long queue of accepted articles awaiting production. Then, if the flow of new submissions (and newly accepted articles) comes in bursts, there is a cushion: articles can continue to appear in the journal anyway. Similarly, if ME or EIC is replaced or goes on vacation, with a hiatus of a month or two, no harm is done.

But a several-month cushion implies a several-month delay; articles appearing the journal are that much older and that much more obsolete; and the journal can't respond quickly to a changing world. To be relevant to its readers and appreciated by its authors, a journal should use "just-in-time" scheduling of papers, and operate with as small a cushion as possible. Since "bursts" of submissions and acceptances cannot be avoided, that may mean that regularly scheduled issues may vary in size; this is not a big problem, and in the future, with electronic dissemination, it may not be even a small problem.

Just-in-time scheduling requires close cooperation between EIC and ME, in identifying problems that need to be resolved, scheduling and rescheduling papers in issues of the journal, and so on. E-mail helps enormously. With sufficient communication between AE, TM, and ME, we can coordinate the ME's copy editing of papers even before they are officially accepted for publication, when the AE feels that the paper is virtually certain of acceptance. This speeds up publication by a few weeks.

In order to run a "just-in-time" journal, EIC needs continuous feedback on the flow of papers through the editorial process. If the number of papers accepted exceeds the page budget for printing them, standards must be tightened accordingly. If there are not enough papers to print, AE's can be encouraged to go out in the community (by e-mail, of course) and solicit new papers.

But these controls must be applied early in the editorial process: it's unpleasant to reject a decent paper on the grounds of limited space, but if it must be done at all it should be done with the first set of referee reports. To ask an author to revise a paper, to referee the revised version, then to reject the paper wastes the time of the author, the referees, and the editors.

By keeping an electronic database with information on every paper in the editorial and production queues, including the size of the paper and its likelihood of acceptance (based on whether the paper is an original submission or solicited revision), the

EIC can attempt to control the queue and keep the journal running smoothly.

Conclusion

The conventional wisdom in the computer science community is that physicists and biologists recognize the importance of refereeing and do it promptly, but computer scientists are much less responsible. I have found that with the right approach (get referees to commit by e-mail to a specific schedule, politely remind them if they're late), computer scientists are willing to provide thorough and conscientious referee reports in a timely fashion.

The Associate Editors love this style of interaction with TM: they *never* have to spend their time reminding referees to send reports, and they don't have to be organized in any way. Their entire job is to receive technical and scientific queries (choose referees, make an editorial decision) from the TM, and respond to those queries.

Before the use of e-mail and electronic filing, TOPLAS had a half-inch-thick file folder on each submitted paper; every correspondence between EIC, AE, Author, Referee, and ME went slowly through U.S. mail; AE's were responsible for reminding referees to send their reports (and hated this task); EIC couldn't easily find out the status of a paper being handled by an AE; and it took years for papers to go through two rounds of refereeing.

Now, only 5% of the submissions are from authors who cannot submit electronically, so we need much less file space. This is mail that we don't need to sort in a mail room, copy on a photocopier, and wait for the U.S. mail to deliver.

The results speak for themselves: Of all papers submitted to TOPLAS in February through June of 1993, for example, the average time to an editorial decision was 83 days—not too bad, for a fully refereed scientific journal. The average time from submission to acceptance (including author revisions and re-refereeing) was 212 days.

Acknowledgments

Rebecca Davies has done an excellent job as traffic manager for TOPLAS. Jan Zubkoff was so effective as managing editor and traffic manager of *Journal of Lisp and Symbolic Computation* (which I observed only as a contributing author) that I was inspired to organize TOPLAS in this way. Carole Appel, journals manager of the University of Illinois Press, shared much of her knowledge of the publishing industry with me. Charles Fischer demonstrated that electronic submission of Postscript files really works. George Criscione has been a great managing editor, always willing to experiment with production innovations.

References

[1] Peter J. Denning and Bernard Rous. The ACM electronic publishing plan. *Communications of the ACM*, 38(4):97–103, 1995.

[2] Jerry Peek. *MH & xmh: E-mail for Users & Programmers.* O'Reilly & Associates, Inc., Sebastopol, CA, 1992.

Appendix

This appendix describes other software and procedures used in the TOPLAS operation.

bigscan This program should be run about once per day; it scans the entire database to produce the report and index files that some of the other commands use. These files are:

- **treport.out** The *transaction report* of which external correspondents are late in responding.
- **tsummary.out** A summary listing of the status of every active paper.
- **tsummary.all** Summary of the status of all active *and* inactive papers. This file is also used as input by the `editor` and `editors` commands.
- **transmit_report** A summary report of the editorial queue in the format used for quarterly reports to the ACM Journals department.
- **referee.out** Index file used by the `referee` command.
- **bigscan.out** Index file used by the `look` command.

mkpapers Example: `mkpapers p120` will create the "virtual folder" to access papers numbered 1200 through 1209.

packup Useful in bundling up the Postscript files for an accepted paper, for transmittal to the ACM's journal production department. For example, given a Postscript file (sent by the author) called `1200.ps`, **packup 1200.ps** will compress and "uuencode" the file and create **1200.ps.uu** with appropriate mail headers to be emailed to ACM. (ACM has requested that Postscript be emailed to them in this form.) Once the file is converted, then the next step is to refile it back to xmh folder:

```
refile -file /u/toplas/tmp/1200.ps.uu +papers/p1200
```

bundle-it We maintain a set of LaTeX style files for use by authors producing camera-ready copy. The files are available to authors on the Internet (the World-Wide Web). From time to time we modify or improve the style files. The `bundle-it` command, when run from the source directory containing these files, installs the latest version in the appropriate FTP and WWW directories for authors to retrieve.

Authors often have trouble using the special style files. The TM, or someone else knowledgeable in LaTeX, must answer their queries. Also, the style file often needs maintenance or improvement to better comply with the ACM format; the TM (or someone else) must collect "bug reports" from authors and from the ACM's copy editors (or managing editor), and fix the style file.

mlatex Once in a while it's necessary to send out letters through physical mail, but we'd like to keep a copy in the e-mail system. This can be done using the special LaTeX letter form in the forms folder, which is simultaneously a LaTeX source file and a valid e-mail message. It looks something like this:

```
\newcommand{\headr}{:
Action: ???
To: ???
Date: ???
------
}

\documentstyle{letter} \nofiles
\begin{document}
\signature{Andrew Appel}
\begin{letter}{
% Address here
}
\opening{Dear :}

\closing{Sincerely,} \end{letter}
\end{document}
```

To use it, first use an Xmh command to create a new draft letter based on this form, and move the draft from the `drafts` folder to the folder for the appropriate paper number (e.g., 1234). Suppose, in doing so, it becomes message 3 in folder `papers/p1234`. At this point it might look like:

```
\newcommand{\headr}{:
Action: receipt
To: Jones
Date: 4/18/95
------
}

\documentstyle{letter} \nofiles
\begin{document}
\signature{Andrew Appel}
\begin{letter}{
Ivan B. Jones\\Department of ....
}
\opening{Dear Professor Jones:}
We have received your paper entitled
``Killing trees'' (TOPLAS 1234).
```

```
\closing{Sincerely,} \end{letter}
\end{document}
```

Now you can run this through LaTeX using

```
mlatex +papers/p1234 3
```

where (in this case) 1234 is the paper number, and 3 is the message number. The result is a file `temp.dvi` that can be printed and mailed.

clean Checks each file in every folder and removes everything *in outgoing messages* below the line

```
---cut here------
```

This is to save on disk storage. A message containing a large Postscript file arrives from the author; then the same Postscript is sent out to AE's and referees. We now have a copy of the file in each outgoing blind carbon copy as well as in the incoming message. It's safe to delete the outgoing copies of the Postscript file. But we don't want to delete anything above the line; this stuff is our record of correspondence with AE's and referees.

Warning: Don't put anything you really need to save below the `---cut here---` of an outgoing message!

refile Suppose you have a Postscript file that didn't arrive in a mail message (perhaps it arrived by FTP), and you want to get it into the mail database. To move a Postscript file `foo.ps` into an xmh message in in the folder for paper number "1234", the procedure is

```
refile -file foo.ps +papers/p1234
```

Camera Ready Copy: Authors who prepare camera-ready copy (using the LaTeX style file package) are expected to be able to provide 600-dpi hardcopy to ACM.

For those authors who cannot, the TM can produce 600dpi laser-printer output with computer modern fonts on special Hammermill Laser Plus (Featuring Wax Holdout) paper, only if author's paper is in LaTeX with *no* system-dependent figures. The ACM's ME will determine who may need this service.

In general, authors should *not* send Postscript files to the TM for 600-dpi output, they should send LaTeX source or dvi files. Most dvi→ps translators use raster fonts which are specialized to a particular resolution (typically 300 dpi), so they'll look bad at 600 dpi. The dvi→ps translation should be done locally with 600-dpi fonts.

Transmit An accepted paper is transmitted to ACM using the e-mail address of a machine that can print Postscript files. It's also useful to send a separate message to the ME, so he'll know to look for it.

Appendices Some papers are published with "electronic appendices" that are not printed in the hardcopy version of the journal. Appendices are in Postscript and will be available by anonymous FTP at Princeton and at ACM. A copy is also available from Article Express (a commercial technical-report distribution company) for a small fee.

TM's responsibility is to make sure the author provides a correct appendix format, place it in the two FTP sites and, send a hard copy to Article Express.

Early Copy Editing: In order to cut the time from acceptance of a paper to publication, we sometimes send papers to ACM for copy-editing even before they are accepted. In principle this could mean wasted work (copy editing papers that will not appear), but ACM sufficiently values fast publication that we have tried this as an experiment.

When editors make a "send back for revision" recommendation on a first-round paper, the TM should ask them which of the following applies:

a. Revised paper will very likely be close to finally accepted paper, *and* the paper is almost certain to be accepted [Minor Revision].

b. Revised paper may not be ready for the copy editor [Major Revision].

Look for keywords like "minor" or "major" in the recommendation for guidance, if this issue has been accidentally omitted.

When in doubt, TM should query the editor. This is only an experiment. **Be sure to explicitly state in the cover letter to the ME that the paper is only for early copy editing.**

Authors' Addresses: TM must notify ACM of any change of author's address.

Postscript Postscript papers are submitted in several ways: normal Postscript, uuencoded and compressed, and some times in tar format.

Xmosaic (or any Wide-World Web browser) is a wonderful tool for the TM to find or confirm email addresses, postal addresses, telephone numbers; or for the Editors to learn about the areas of expertise of a potential referee.

Accessing old papers: When a "virtual folder" contains only inactive papers, it is deleted (the underlying "real folders" still continue to exist in the (large) `papers` folder). But sometimes one needs to revisit an inactive paper to look up some information. Here is the procedure (for paper #1464):

```
ln -s ~/Mail/papers/p1464 ~/Mail/p14x
```

Exit xmh and re-enter xmh and folder p1464 will re-appear in p14x.

Inactive folders Virtual folders are not deleted automatically. When all the papers in the range (for example) 1460-1469 have disappeared from `tsummary.out`, then the virtual folder may be deleted by

```
rm ~/Mail/p146/*
rmdir ~/Mail/p146
```

Exit and re-enter to see the change.

todo mm/dd/yy Sometimes the TM or EIC wants to create a reminder message that will show up in the *report* on a certain date. This can be done with the **todo** action:

```
   Action: todo 3/25/95
   To: nobody
   Fcc: papers/p1630
   Subject: whatever

  whatever message body you like
```

Then, in the treport on 3/30/95 (for example), you'll see:

p1630:53   5 days ago, do: whatever

Also, when you do "status 1630" even before the "due date," you'll see an indication:

p1630:53   *20 days from now, do: take out the trash

After the job is done, edit "Action: none (was todo 3/25/95)" and it will disappear in the treport.out when bigscan is run and it is a record that the job is complete.