MEMORANDUM

To: Frank Heart (c.c. WRC, AAM, BPC, JBL)

From: Dave Walden

Subject: Division 6 Use of TENEX -- 2

Date: March 2, 1973

I don't think Strollo and Burchfiel's answers to my previous memo are entirely sufficient.

1. The problem with listings being broken into 5-minute chunks is that pieces are frequently lost (apparently in the spooler on some occasions). I suggest that print jobs up to 15 or 20 minutes long should be done in their entirity with them being queued for printing strictly by length (shorter first). (Incidentally, the 5-minute splits are not inconsequential as not everyone bursts their listings every time.)

2. Our need for a paper tape punch (and reader) is almost only for export and import to mini's. But we do move a lot of paper tape back and forth to mini's. The TENEX reader and punch are probably fast enough. (Such things as threading convenience become important if the reader and punch are used a lot.) Since my previous memo, I have come to understand that the main characteristic the reader and punch should have is that they must be in the IMP room. A reader and punch would have to be remoted from TENEX; and, perhaps, this could be done over the network, for instance to a reader and punch which would be installed on the NCC 316 computer (I don't know that the TIP can support a high speed reader and punch).

3. (From the user's point of view, some documentation and a little bit of (grudging) consultation is abailable from the computer center staff.) I think Jerry and Ted miss the point of my comment: for a period after any switchover to TENEX we would probably need exceptional amounts of consultation to quickly effect the switchover. We might also discover that some (small?) system changes would be needed to support our use of TENEX. Basically, we would want the TENEX group to support our changeover with energy,

enthusiasm, and acceptance of the fact that they don't necessarily know what's best for us.

4. I hate a continual hassel to garbage collect the discs. We would be pretty good about putting unused files on demountable discs, tapes, etc. But sometimes we might have a very large number of active files. This can, of course, ultimately be solved by adding extra drives -- it might not be a bad idea to help the TENEX group pay for a couple of disc drives so they are mostly ours. At least we should pay a flat rate for disc storage.

5. We are not interested in a more flexible, general, and powerful interface between user processes and the network. We want to send and receive network messages -- subject to no constraints. For instance, we may be doing an experiment with the performance achieved when subnet error control or flow control are violated. We might want to send non-protocol messages within the protocol space to certain (perhaps non-protocol) Hosts. We are willing to promise not to be malicious and to debug carefully. Jerry and Ted point out that they are now designing an interface which permits network messages is an interesting one - the fact is that I have been asking to be able to send network messages from TENEX 6 months and at one point long ago I was promised I would have it in 14 days. We will not always be able to wait while elogent packages are designed and built -- sometimes ad hoc kludges will have to be provided for us for the interim.

6. Great.

7. I can't accept that hacks will necessarily be non-interfering or that they should be subject to scrutiny. Again I suggest that we try to get a flat yearly rate which would subsume both our hacks and our work. I don't want to be any more inhibited about hacking than I now am or than Tomlinson is.

8. Ted and Jerry completely miss the point of my comment. While autostartup is nice and we'll love it a lot, what we want is auto-continuation. If I'm getting a listing on my teletype and TENEX goes down for a day of maintenance the next character in the listing should be printed on my teletype when TENEX comes back up. If I'm computing something that takes a week, the same computation should proceed for a logical week over physical stops. If I'm writing a disc file, and the system is stopped to clean the fans, when the system comes back up, my next disc read should be done. I hate it when TENEX says "SYSTEM RESTARTING..." and loses my work. Of course, there will always be occassional unrecoverable crashes, but in general only one user should have to be killed to recover and significant efforts should be made to save that one. A time sharing system should never require all users to logout.

9. Good -- we can get a minimum fraction of the CPU allocated to a given process -- we will probably want this for the process that reloads TIPs, etc. But can we always login? We want a guarantee that 6 or 8 jobs can always login.

10. It will be fine if we can arrange that our minimum operational requirements (reloading TIPs, receiving network status reports, etc.) can be serviced by system B when system A is down. If we have warning, our program development can probably be carried out on other network TENEXs when system A is down -- another reason to have our own remote punch. By the way, it would be best if we could also be able to send and receive network messages from other TENEXs for when the local ones are down. The TENEX operators would probably have to switch our minimal stuff to system B when A is down -- are they willing to do this?

11. If four or fives lines are permanently patched to TENEX and we get an operational TIP in addition to the developmental one, we will be glad to access TENEX via the network if we are always guaranteed network ports for the 10 additional terminals and the TENEX network handling code is made more efficient so it is comparable to direct connections.

12. There must not be any breakpoints in the TENEX we use!!! TENEX should be modified and debugged on system B. (By the way, TENEX now does have breakpoints in it and we get blamed ("lost allocate", etc.) when TENEX hits a breakpoint -- TENEX should at least send a "TENEX BREAKPOINT" message to network users before halting for the breakpoint.)

13. Of course, debugging of TENEX should be done under the general control of Division 5. However, we don't work the same hours as the Division 5 guys and they don't have the same concerns. For instance, I have been trying to get them to help us find the "lost allocate" bug (wherever it might be) for months. We want to be able to dig in and patch the NCP to write bad messages to the disc, etc. if this is necessary. Yes, wheel status is separable from debugging status but we want wheel status too -- we want to, or change our passwords, or give our new employee NETWIZ privileges. Perhaps round the clock TENEX operators would be a substitute for wheel status.

14. As I remember, the day after Jerry showed me a draft of their memo, the TENEX password system was cracked by someone from out of the network.

15. Neat.

16. O.K.

17. Prior notice so we can move to another network TENEX and availability of our minimum functions on system B should be sufficient.

18. I don't think we could stand for both systems to be unavailable several nights a week for two to four weeks.

I have thought of some more points since my last note.

1. Can TENEX stand another 10 to 15 users, 8 to 10 of them very heavy users?
2. Our operators should be trained to restart TENEX and do other normal TENEX operator functions such as putting paper in the

line printer so they can keep it running nights, weekends, and Christmas.

3. I have noticed an interesting difference between the TENEX operators and the NCC operators. The NCC operators do what I tell them ("get a listing quick", "punch a paper tape quick", "put wide paper in the printer"). The TENEX operators are very cavalier about our requests for their help (I'll get your listing after coffee", "I'll put wide paper in the printer tonight").

4. Our priorities and needs would have to be taken into consideration when scheduling software development, system maintenance, operator coverage, etc. For instance, we release systems early Tuesday mornings and would perhaps require a TENEX operator standing by during this period.

5. TENEX doesn't have STRCOMP (TELCOMP is no substitute - nothing is).

Basically, I'm very favorable about moving to TENEX. There are problems but if the TENEX guys are willing to help try to solve them (rather than saying they don't exist), we should be able to solve them. I think there is nothing technically infeasible about using TENEX except perhaps system capacity. In general, I think all non-economic problems can be nicely resolved if the TENEX and IMP guys attack each problem trying to find a pragmatic, technically sound solution.

DW/ph