```
Interesting elements within NSF <.d:et>
   Directorate for Mathematical and Physical Sciences, and Engineering
   [MPE]
   <:ebt>
      Division of Mathematical and Computer Sciences [MCS]
      <:ebt>
         Computer Science Section
            Kent K. Curtis, Section head
            Theoretical Computer Science Program
               Bruce H. Barnes, Program director
            Sotware Systems Science Program
               Thomas A. Keenan, Program director
            Computer Systems Design Program
               John R. Lehmann, Program director
            Intelligent Systems Program
               Sally Yeates Sedelow, Program director
            Software Engineering Program
               J. Richard Phillips, Program director
            Special Projects Program
               Fred Weingarten, Program director
   Directorate for Research Applications [RA]
   <:ebt>
      Division of Advanced Productivity Research and Technology [APRT]
      <:ebt>
         James D. Cowhig, Division Director
      Division of Exploratory Research and Systems Analysis [ERSA]
      <:ebt>
         Joshua Menkes, Director
   Directorate for Scientific, Technological, and International Affairs
   [STIA]
   <:ebt>
      Division of Policy Research Analysis [PRA]
      <:ebt>
         John V. Granger, Division Director
         Processes of Innovation and their Management
            Alden S. Bean, Group Leader
      Division of Science Information [DSI]
      <:ebt>
         Lee G. Burchinal, Head
         Access Improvement Program
             Harold E. Bamford, Jr., Program Director
         Research Program
            Edward C. Weiss, Program Director
         User Requirements Program
            Joel D. Goldhar, Program Director
   Directorate for Administration [A]
   <:ebt>
      Division of Information Systems [DIS]
      <:ebt>
         George Pilarinos, Division Director
         Computer Facility Management Branch
         Plans and Technical Standards Branch
         Softare Systems Management Branch
         Systems Support Services Branch
```

```
USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-
USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-
USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-ISIC   USC-

MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD
MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD
MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD   MAYNARD

(DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DS
(DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DS
(DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DSM)37147   (DS

FRIDAY, NOVEMBER 12, 1976 01:18:04-PST   FRIDAY, NOVEMBER 12, 1976 01:18:04-PST
FRIDAY, NOVEMBER 12, 1976 01:18:04-PST   FRIDAY, NOVEMBER 12, 1976 01:18:04-PST
FRIDAY, NOVEMBER 12, 1976 01:18:04-PST   FRIDAY, NOVEMBER 12, 1976 01:18:04-PST
```

The INCLUDE Sequence Generator 1

A new sequence generator called INCLUDE now exists as <xprograms,include.sq,>. This sequence generator scans each statement to see if it begins with the keyword "INCLUDE". If so the sequence generator replaces the statement by the Structure pointed to by the link(s) in the statement. 1a

To use this sequence generator issue the following command: 1b

Execute Programs Load Program (xprograms,include.sq,) 1b1

After executing this command setting viewspec O turns the sequence generator on, viewspec P turns it off. This sequence generator may be used in conjunction with the output processor and with the compilers, as well as with all other commands which employ viewspecs. The legal structure types to include are BRANCH, GROUP, PLEX, and STATEMENT, with BRANCH being the default. 1c

EXAMPLES: 1d

INCLUDE BRANCH (directoryname,filename,statementname) 1d1

INCLUDE GROUP (link) (.s:e) 1d2

This would include the group from the satement named link in the current file through its successor with viewspec e , relative level clipping, on. 1d2a

INCLUDE PLEX (maynard,dsm,journal.d:wli;SINCE (14-OCT-76 00:00);) 1d3

This would include all journal mail received by Maynard since October 14, 1976. 1d3a

Notes: 1e

Any comments in the INCLUDE statement should appear after the link, or between the two links in the case of INCLUDE GROUP. IF statement numbers are on, they are generated from the pointed to file. 1e1

INCLUDES may be nested up to a depth of three structures. Viewspec O in an INCLUDE link will cause the INCLUDES, in the structure pointed at by the link, to be expanded. Viewspec P in an INCLUDE link will turn off the expansion process. The limitation on the number of nested includes will be increased from 2 to 8 in the near future. 1e2

Due to the way NLS does line clipping, line clipping viewspecs (s,t, and r) will be ignored in INCLUDE links. The line clipping in effect for the top level file will stay in force over all included structures. All other viewspecs in an INCLUDE link are merged with the current viewspecs and are applied to the included structure. Thus level clipping, content filters, statement numbers, statement signatures, etc., may be controlled for each structure included. 1e3

Warning: 1f

A this is an experimental sequence generator. Although there are no known bugs, we can't rule out unknown bugs. Please report any problems to FEEDBACK. 1f1

Acknowledgments: 1g

The Include sequence generator was originally written by Ken Victor (KEV) for his program support LIBRARY subsystem. David Maynard (DSM) modified this and extended it to allow the inclusion of structures other than branches, and Dave Hopper (JDH) rearranged the address space of NLS to allow this sequence generator to work in conjunction with the Output Processor, and the L10 and CML compilers. 1g1

```
< MAYNARD, INCLUDE.NLS;5, >, 18-OCT-76 17:11 DSM ;;;;
FILE cinclude %(110.) to (maynard,include.sq,)%
   DECLARE EXTERNAL swtable = (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0);
   % This table stores pairs of addresses of sequence work areas.
   It is used to insure proper closing of all open sequences for both
   TNLS and DNLS%
   DECLARE swtl = 15; %length of swtable%
   DECLARE  bstruc=1, pstruc=2, gstruc=3, sstruc=4; %structure type codes%
   (include) PROCEDURE (sw, which); % Compile File with INCLUDES %
      LOCAL soffset, poffset, cm1tmp;
      LOCAL vpc, civp1, civp2, cica, cius, struc;
      LOCAL gotone, stid, i, svcm1;
      LOCAL TEXT POINTER tp, tps, tpe, tp2;
      LOCAL lpds[40], lswa[10];
      LOCAL STRING locstr[200];
      REF cm1tmp, sw;
      CASE which OF
         =sqopn: % called at open %
            BEGIN
            %Nothing to do here. Wait until we are called in sqgnxt mode.  This
            is because any local state information created here will go away
            when we return. In sqgnxt mose however we don't return until the
            sequence is finishedd thus state information is saved. State can't
            be saved in simple globals because this sequence generator is
            recurrsive%
            END;
         =sqgnxt: % called for next in seq -- fall through %
            BEGIN
            &cm1tmp _ 0;
            CASE 1 OF % Start with case 1%
               = 1: %processing the next statement, either from the passed
               sequence or from the inferior sequence which we have opened
               here.%
                  BEGIN
                  IF NOT &cm1tmp THEN %IF no inferior sequence is open%
                     BEGIN
                     gotone _ FALSE;
                     % if viewspec i or k is on and the user has a content
                     filter invoked we must skip to first statement that passes
                     filter%
                        IF (sw.swvspec.vscapf OR (sw.swvspec.vscakf AND NOT
                        sw.swkflg)) AND sw.swcacode > 0 AND sw.swstid # endfil
                        THEN
                           DO
                              BEGIN
                              stid _ sw.swcstid;
                              FIND SF(stid);
                              IF [sw.swcacode](&sw) OR inptrf THEN
                                 BEGIN
                                 sw.swstid _ stid; %may have been zapped by the
                                 c.a. program%
                                 gotone _ TRUE;
                                 END
                              ELSE      seqnxt(&sw);
                              END UNTIL gotone OR sw.swstid = endfil;
                     IF  sw.swstid  = endfil THEN %Primary sequence has been
```

```
exhausted%
    BEGIN
    sport( &sw); %Note: this performs a return. We won't get
    control back until we are called in sqcls mode%
    END;
tp _ sw.swstid;   tp[1] _ 1;
IF FIND tp > "INCLUDE" ^tps THEN %Here we go: process the
include%
    BEGIN
    FIND SE(tps) ^tpe;
    IF FIND tps > $NP ^tps THEN NULL;
    struc _ CASE READC OF
        = 'G, ='g: gstruc;
        = 'P, ='p: pstruc;
        = 'S, ='s: sstruc;
        = 'B, ='b: bstruc;
        ENDCASE bstruc;
    IF tpe[1] <= tps[1] THEN REPEAT CASE(2); %no link found
    - sport the naked INCLUDE%
    ON SIGNAL ELSE
        BEGIN
        %typeas($"Signal trap");%
        %This traps all signals generated in evaluating the
        link (file not online, illegal link syntax, etc). If
        there is any problem we simple sport the INCLUDE
        statement without expanding it.  It might be useful
        to append the error message to the bad INCLUDE
        statement before we sport it.%
        REPEAT CASE(2);
        END;
    lnkprs($tps,$lpds);
    %evaluate the link and save associated viewspecs,
    content analyzer code address , and sequence generator
    code address%
    civp1 _ caddexp( $lpds[ls], $lpds[le], lda(), $tp :
    civp2, cica, cius);
    CASE struc OF %Open a sequence with proper parameters.
    Note that we may (probably are) opening the sequence
    with include as the sequence generator%
        = bstruc:
            BEGIN
            civp1.vsbrof _ TRUE;
            &cmltmp _ openseq( tp, tp, civp1, civp2, %0% cius
            , cica);
            END;
        = sstruc:
            BEGIN
            civp1.vslev _ 0;
            &cmltmp _ openseq( tp, tp, civp1, civp2, %0% cius,
            cica);
            END;
        = pstruc:
            BEGIN
            tp _ gethed(tp);
            tp2 _ getail(tp);
            civp1.vsplxf _ TRUE;
```

```
               &cm1tmp _ openseq( tp, tp2, civp1, civp2, %0%
               cius, cica);
               END;
            = gstruc:
               BEGIN
               lnkprs($tpe,$lpds);
               tp2 _ tp; tp2[1] _ tp[1];
               %evaluate the second link relative to the first%
               civp1 _ caddexp( $lpds[ls], $lpds[le], lda(), $tp2
               : civp2, cica, cius);
               IF NOT grptst(tp,tp2) THEN err($"illegal group");
               civp1.vsplxf _ TRUE;
               &cm1tmp _ openseq( tp, tp2, civp1, civp2, %0%
               cius, cica);
               END;
            ENDCASE;
         ON SIGNAL ELSE;
         poffset _ sw.swclvl;
         soffset _ -1;
         %save current state of primary sequence, sw%
         cpysw (&sw, $lswa);
         %remember the fact that the sequence &cm1tmp is an
         inferior of &sw%
         addsw(&sw,&cm1tmp);
         REPEAT CASE(3);
         %process the statement from the inferior sequence%
         END
      ELSE REPEAT CASE( 2); %not an INLCUDE, sport it%
      END
   ELSE REPEAT CASE(3); %we are processing an inferior sequence%
   END;
= 2: %process satement from primary sequence%
   BEGIN
   sport( &sw);
   %find next statement in sequence (viewspec i,k will be take
   care of in CASE 1)%
   seqnxt( &sw);
   REPEAT CASE(1);
   END;
= 3: %process satement from inferior sequence%
   IF seqgen( &cm1tmp) = endfil THEN
      %inferior sequence exhausted%
      BEGIN
      %close inferior sequence%
      clssons(&cm1tmp);
      svcm1 _ &cm1tmp;
      closeseq( &cm1tmp := 0);
      FOR i _ 0 UP UNTIL >= swtl DO
         BEGIN
         IF swtable[i].LH = svcm1 THEN swtable[i].LH _ 0;
         IF swtable[i].RH = svcm1 THEN swtable[i].RH _ 0;
         END;
      %restore &sw work area%
      cpysw($lswa, &sw);
      %find next statement of primary sequence%
      seqnxt( &sw);
```

```
               REPEAT CASE(1);
               END
            ELSE
               BEGIN
               IF soffset = -1 THEN soffset _ cm1tmp.swclvl;
               %copy inferior work area into superior work area%
               cpysw( &cm1tmp, &sw);
               %adjust level of included structure to reflect the level of
               the INCLUDE%
               sw.swclvl _ cm1tmp.swclvl + poffset - soffset;
               sport( &sw);
               %go get another statement%
               REPEAT CASE(1);
               END;
         ENDCASE;
      END;
   =sqcls: % called at close %
      BEGIN
      % We must be sure that any sequences inferior to us are also closed%
      clssons(&sw);
      END;
   ENDCASE err ($"Illegal call to a seq gen program"); % called for any
   other purpose -- error %
RETURN;
END.
%%
```

```
(addsw) PROCEDURE (father,son); % Add a sequence work area pair to swtable%
   LOCAL i;
      FOR i _ 0 UP UNTIL >= swtl DO
         IF NOT swtable[i] OR (swtable.LH = father AND swtable.RH = 0) THEN
            BEGIN
            swtable[i].LH _ father;
            swtable[i].RH _ son;
            RETURN(TRUE);
            END;
      RETURN(FALSE);
      END.
(clssons) PROCEDURE (father); % Closes all sequences which are descendents of
the father%
   LOCAL i;
      IF NOT father THEN RETURN(FALSE);
      FOR i _ 0 UP UNTIL >= swtl DO
         IF swtable[i].LH = father  THEN
            BEGIN
            %close son's sons%
            clssons(swtable[i].RH);
            IF swtable[i].RH THEN closeseq(swtable[i].RH := 0);
            END;
      RETURN(FALSE);
      END.
FINISH
```

The INCLUDE Sequence Generator

A new sequence generator called INCLUDE now exists as <xprograms,include.sq,>. This sequence generator scans each statement to see if it begins with the keyword "INCLUDE". If so the sequence generator replaces the statement by the Structure pointed to by the link(s) in the statement.

To use this sequence generator issue the following command:

```
Execute Programs Load Program (xprograms,include.sq,)<CA>
```

After executing this command setting viewspec O turns the sequence generator on, viewspec P turns it off. This sequence generator may be used in conjunction with the output processor and with the compilers, as well as with all other commands which employ viewspecs. The legal structure types to include are BRANCH, GROUP, PLEX, and STATEMENT, with BRANCH being the default.

EXAMPLES:

```
INCLUDE BRANCH (directoryname,filename,statementname)
INCLUDE GROUP (link) (.s:e)
```

This would include the group from the satement named link in the current file through its successor with viewspec e , relative level clipping, on.

```
INCLUDE PLEX (maynard,dsm,journal.d:wli;SINCE (14-OCT-76 00:00);)
```

This would include all journal mail received by Maynard since October 14, 1976.

Notes:

Any comments in the INCLUDE statement should appear after the link, or between the two links in the case of INCLUDE GROUP. IF statement numbers are on, they are generated from the pointed to file.

INCLUDES may be nested up to a depth of three structures. Viewspec O in an INCLUDE link will cause the INCLUDES, in the structure pointed at by the link, to be expanded. Viewspec P in an INCLUDE link will turn off the expansion process. The limitation on the number of nested includes will be increased from 2 to 8 in the near future.

Due to the way NLS does line clipping, line clipping viewspecs (s,t, and r) will be ignored in INCLUDE links. The line clipping in effect for the top level file will stay in force over all included structures. All other viewspecs in an INCLUDE link are merged with the current viewspecs and are applied to the included structure. Thus level clipping, content filters, statement numbers, statement signatures, etc., may be controlled for each structure included.

Warning:

A this is an experimental sequence generator. Although there are no known bugs, we can't rule out unknown bugs. Please report any problems to FEEDBACK.

Acknowledgments:

The Include sequence generator was originally written by Ken Victor (KEV) for his program support LIBRARY subsystem. David Maynard (DSM) modified this and extended it to allow the inclusion of structures other than branches, and Dave Hopper (JDH) rearranged the address space of NLS to allow this sequence generator to work in conjunction with the Output Processor, and the L10 and CML compilers.

Sequence Generator Extension (37074,)

< AJOURNAL, 37956.NLS;1, >, 3-DEC-76 12:31 XXX ;;;; .HJOURNAL="DCE 3-DEC-76 11:18 37956"; Title: .H1="Memo to ISE Directors re ARC directorship and management"; Author(s): Douglas C. Engelbart/DCE; Sub-Collections: SRI-ARC; Clerk: DCE; .IGD=0; .SNF=HJRM; .RM=HJRM-7; .PN=-1; .YBS=1; .PES;

After much thought and a deep examination of my personal, family, and professional situation, I have decided that the following position is best for me:

1 I am committed to pursue the same basic professional goals as over the past 25 years, and as now embodied in the "ARW Program."

2 To pursue that program at SRI, I need to have clear control of the program, and of the program's support organization (which, to me, is what ARC is).

If a special role for management of ARC operations is established, I would need that role to be subordinate to me, and would need the filling of that role to be accomplished in the normal manner of establishing requirements, and reviewing candidates. Since that role would interact heavily with SRI line management, I would expect more than ordinary involvement on its part in defining and filling this role, but I can't accept an arbitrary, unilateral, role definition and personnel assignment.

3 If SRI finds that it must act against these above conditions, then:
   a. I will immediately submit my resignation from the Institute;
   b. I will concentrate for a few months on reconstituting a desolated home environment;
   c. I will then pursue support and a home-location situation for putting together a program that is likely to compete directly with what SRI-ARC is doing now. (Note: At this time, I have no private understandings with any ARC staff or ARC sponsors about such a plan.)

4 If SRI leaves me in place, then:
   a. I will bring a basic formulation to Earle Jones next week as to how I propose to satisfy both what SRI vitally needs in the way of financial performance and limitations of risk, and what I need for a home base from which to build the program. This first draft will be worked out with Jim Norton and Jon Postel initially representing ARC-staff position, and before any final ratification the agreement would be discussed seriously with ARC staff.
   b. If, within a reasonable period, we can't reach mutual agreement, or if after agreement there is subsequent failure to meet its terms, then I expect to give up trying to pursue this program at SRI.

To: ARC staff
cc: Earle Jones, Bert Raphael
Re: Appropriate role-change protocol, and cancelling the meeting

Friday evening, Bert Raphael told Bonnie Mosher to announce an all ARC meeting for Monday. This memo cancels that meeting.

Until there is an official announcement to the contrary, I am still Director of ARC; the only people who can appropriately order an all-ARC meeting are me, or someone that I deputize to do so, or Earle Jones, Don Scheuch or Charles Anderson. If Bert wants such a meeting, he should make arrangements through me or Earle.

There has been no official announcement that I have seen; such an announcement would be officially distributed to each of you in advance of a role change, describing the titles, duties, and authority. It is quite unbelievable that an official transfer is taking place this Monday without any of us having seen such an announcement. My latest discussions with Earle on Friday left me with the clear indication that the re-organizations plans under consideration prior to that time were to be re-considered in light of the position expressed by me in (37956,), and that I would be duly informed of the management's reaction as a next step in the dialog.

The issue clearly is in limbo. There is no basis for Bert to begin acting otherwise. The ball is in Earle's court. Until he says otherwise, I am still boss and I really am not in a mood to tolerate any assumptions to the contrary.

We have been interacting for a year and a half with the Swedish industrial/government group that begins it's week-long AKW Seminar on Monday morning. And there are many other solid things for us all to be doing. I know that we are all uneasy about the uncertainties, but the best thing we can do is to dig in and do our work. For me, I am concentrating on the positive, and unafraid of the negative. And I'm betting on the positive.

Journal Accession Number: 37986

Version 9

INTRODUCTORY TNLS

A Course Outline

22 MAR 77

Applications Development

Augmentation Research Center
Stanford Research Institute
Menlo Park, California 94025

THE INTRODUCTORY TNLS COURSE OUTLINE

TABLE OF CONTENTS

INTRODUCTION TO NLS 2

HOW TO TYPE COMMANDS AT THE TERMINAL 3

GETTING TO NLS 4

LEAVING NLS 5

1. FILES 6

2. TYPING IN INFORMATION 7

Figure 1: Example of structured file 9

3. TYPING OUT INFORMATION 10

4. EDITING 11

5. COMMUNICATING 14

6. TROUBLE SHOOTING AND HELP 18

PRACTICE 19

INDEX 20

INTRODUCTION TO NLS

NLS = oN Line System

TNLS = Typewriter Version

CAPABILITIES OF SYSTEM:

- Composing
- Editing
- Studying (Browsing)
- Printing
- Publishing (Various formats)
- Graphics
- Calculating
- Communicating (Messages, Teleconferencing)
- Storing and retrieving (Management and organizational information)

Purpose and Description of the Introductory Course

The commands which are included in this first course have been selected to let a user write, edit, store, and communicate typewritten information [text]. Those items labeled [Optional] may be covered on the second day of the course.

The course is organized by concepts, ordered as one would need them to use the system. Under each concept are the exact commands needed for a particular task. The course is designed for use when terminals are available for all participants to use or view easily.

How to type commands at the terminal:

You type characters in response to the system's prompting to tell the computer what to do. Throughout this outline we spell out both what should be typed and what prints in response at the terminal.

You type the characters underlined and capitalized in this outline (you do not have to type the letters in as uppercase characters!) All other characters will be filled out for you by the computer.

Words that are all in uppercase and underlined in this outline represent something appropriate that you type, i.e., for PASSWORD, you type your own password.

Special Keys:

Special keys you should type are enclosed in < and > in this outline. For example:

<SP> means strike the SPACE bar.

<CR> means strike the carriage return [RETURN] key.

CTRL followed by a letter means you hold down the control [CTRL] key while typing the specified character.

For example, <CTRL-X> means hold down the CTRL key while typing an x. This will abort a command.

GETTING TO NLS

If dialing in, turn on the terminal and make sure it is online, set for upper and lower case characters, and the speed to 30 cps. Dial the TIP number, and place the receiver in the cradle after hearing the tone.

[I] Type E

[To get the Network's attention if the TIP hasn't identified itself]

[II] After you see something like: `ARPA TIP 371 #: 22`
Type @ 0 <SP> 43 <CR>

[This opens a connection to Office-1]

[III] After you see something like:

```
Trying...
Open

TENEX 1.33.9, OFFICE-1 EXEC 1.53.19
@
```

Type LOG <SP> DIRECTORYNAME <SP> PASSWORD <SP> <CR>

[The last SPACE fills in the account number automatically; you're then ready to call NLS.]

[IV] After you see something like:

```
JOB # ON TTY# DATE TIME
PREVIOUS LOGIN: DATE TIME
@
```

Type TNLS <CR>

[If you are logged into a group directory you will be prompted for your ident.]

READY SIGNALS:

Ready Signals are characters printed on your terminal to prompt you for a command.

@ is the TENEX ready signal. It means you can type in a TENEX command.

BASE C: is the TNLS ready signal. It means you can type in an editing command [like home base...].

SEND C: is the Sendmail subsystem ready signal. It means you can type in a Sendmail command.

LEAVING NLS

[I] Use the NLS command: BASE C: <SP>Logout OK: <CR>

[II] After you see something like:

```
LOGOUT JOB 3, USER ACCT 3, TTY 3, AT DATE TIME, USED TIME
```

Type @ C <CR>

[To close the TIP connection to Office-1]
When you see "Closed", hang up the phone and turn off the terminal.

PROMPTS in NLS COMMANDS:

A capital letter followed by a colon will be printed on your terminal when it's your turn to type. For instance,

C: indicates you are to type a command word. The underlined, uppercase characters should be typed, the lowercase part of the word will be filled out and sometimes a helpful phrase in parentheses.

T: indicates you are to type some text. Since words that are all in uppercase and underlined in this outline represent something appropriate that you type, TYPEIN means to type a set of characters on the keyboard (maybe a paragraph), ended with a confirmation, often a carriage return.

INTRODUCTORY TNLS

1. FILES

To create a new file:

```
BASE C: <SP>CReate C: File T: FILENAME <CR>
< DIRECTORY, FILENAME.NLS;1, >
```

[A filename can be eny short "word", like a folder label. It may not contain spaces, commas, periods or semicolons.]

To see a list of all your files:

```
BASE C: <SP>SHow C: Directory (of) OK/T: <CR>
OK: <CR>
```

You are always in a file when working in NLS. When you type NLS, a file called the "initial file" is the one you're automatically in. Its name is based on the initials of the person logged in. It is your mailbox for NLS mail and will be discussed in the Sendmail section on page 14. If you want to work in another file, use the command described below.

** If you forget to specify a file upon entering NLS **
** any work you do will be in your initial file. **

To work in a file that is already created:

```
BASE C: Jump (to) C: Link T: FILENAME, <CR>
```

2. TYPING IN INFORMATION

To enter sentences, headings, or paragraphs:

```
BASE C: Insert C: Statement (to follow) A: STATEMENT NUMBER <CR>
L: <CR>
T: TYPEIN <CR>
```

[TYPEIN means type whatever you want for the statement]

To enter a series of sentences, headings, or paragraphs:

Continue to insert statements without repeating the Insert Statement command by typing a <CTRL-E> instead of the last <CR>. <CTRL-E> puts you in the "Enter mode" -- you type <CR> to end each statement, when you are finished inserting statements, type a <CTRL-X> after the prompt L:

To correct errors immediately while typing:

<CTRL-A> =	backspace character [The deleted character will be printed after a slash]
<CTRL-W> =	backspace word [An underline or backarrow will be printed]
<CTRL-X> =	abort command

[Optional] To type in statements at different levels:

When a statement is typed in "down a level" each line of that statement will be indented 3 spaces to the right of the statement it follows. Its statement number will reflect the level, see an example on the next page.

The L: prompt in the Insert Statement command is prompting you for the level of the new statement being typed in. You may respond to that prompt in one of the following ways:

<CR> - new statement will be at the same level as the statement you address it to follow.

u <CR> - new statement will be up a level from the statement you address it to follow. You can type more than one u if you need to go up more than one level.

d <CR> - new statement will be down a level from the statement you address it to follow.

Example of structured file:

```
< BAIR, MENU.NLS;1, >, 28-FEB-77 17:29 JHB ;;;;

1 SOUP

 1A VEGETABLE

 1B CREAM OF MUSHROOM

2 ENTREE

 2A  FRIED CHICKEN

 2B  SALMON

  2B1 WITH CREAM SAUCE

 2C  PRIME RIBS

3 DESSERT

 3A  PIE

  3A1  APPLE

   3A1A A LA MODE

  3A2  BLUEBERRY

 3B  ICE CREAM

  3B1  VANILLA

  3B2  PEPPERMINT

  3B3  MAPLENUT

  3B4  CHOCOLATE

4  BEVERAGE

 4A  TEA

 4B  COFFEE
```

Figure 1

3. TYPING OUT INFORMATION

To print the current statement:

```
BASE C:                    [Backslash prints the
                           statement where you are]
```

To print the file:

```
BASE C: Print C: File OK: <CR>
```

The origin statement

A newly created file contains one statement called the origin statement. It contains the file name and other information about the file for the system. It should not be edited. It is numbered 0, but no number will be printed.

To stop printing anytime type a <CTRL-O>; it takes some time to get through.

To print any statement:

```
BASE C: Print C: Statement (at) A: STATEMENT NUMBER <CR>
V: <CR>
```

[Optional] To print a section of the file:

```
BASE C: Print C: Branch (at) A: STATEMENT NUMBER <CR>
V: <CR>
```

A branch is a statement plus all its substatements. See preceding page for an example.

4. EDITING

To correct typos, add, delete or replace phrases:

```
BASE C: Substitute C: Text (in) C: Statement (at)
A: STATEMENT NUMBER <CR>
(New TEXT) T: TYPEIN <CR>
(for all occurrences of old TEXT) T: TYPEIN <CR>
Finished? S/Y/N: Y [Type Y for yes and N for no]
Substitutions made: Number
```

[Replaces the old text with the new text every time it occurs in the statement]

[Optional] To insert text at the end of an existing statement, particularly useful if the text is more than one line long.

```
BASE C: Insert C: Text (to follow) A: STATEMENT NUMBER +e <CR>
T: TYPEIN <CR>
```

[+e means the end of the statement]

To delete statements:

```
BASE C: Delete C: Statement (at) A: STATEMENT NUMBER <CR>
OK: <CR>
```

To replace statements:

```
BASE C: Replace C: Statement (at) A: STATEMENT NUMBER <CR>
(by) T: TYPEIN <CR>
```

To move statements:

```
BASE C: Move C: Statement (from) A: STATEMENT NUMBER <CR>
(to follow) A: STATEMENT NUMBER <CR>
L: <CR>
```

[Optional] L: d or u's can be typed to specify the level of the statement relative to the one you addressed.

To copy statements:

```
BASE C: Copy C: Statement (from) A: STATEMENT NUMBER <CR>
(to follow) A: STATEMENT NUMBER <CR>
L: <CR>
```

[Optional] L: d or u's can be typed to specify the level of the statement relative to the one you addressed.

To delete a file:

```
BASE C: Delete C: File T: FILENAME <CR>
OK: <CR>
```

[Careful, this removes all versions of the file. You can "undelete" a file anytime before logout with the Undelete File command, see p. 18.]

To update a file: [Should be done periodically to save changes for backup, not imperative]

```
BASE C: Update C: File OK/C: <CR>
<DIRECTORY, FILENAME.NLS;2,>
```

[Optional] To change whole sections:

A branch is a statement plus all its substatements.

You can't delete or move a statement with substatements, instead you must delete or move a branch.

To delete a branch:

```
BASE C: Delete C: Branch (at) A: STATEMENT NUMBER <CR>
OK: <CR>
```

To move a branch:

```
BASE C: Move C: Branch (from) A: STATEMENT NUMBER <CR>
(to follow) A: STATEMENT NUMBER <CR>
L: <CR>
```

To copy a branch:

```
BASE C: Copy C: Branch (from) A: STATEMENT NUMBER <CR>
(to follow) A: STATEMENT NUMBER <CR>
L: <CR>
```

[Optional] Formatting Technique:

For lines within a statement to be shorter than the right margin, as in a mailing address, carriage returns must be typed as part of the text of the statement.

To type a carriage return within a statement without ending the statement, type <CTRL-V> <CR> at the appropriate places in the text each time you want a carriage return.

5. COMMUNICATING

[Optional] Communicating in NLS: SENDMAIL (Subsystem)

The following is an example of sending a message using the Interrogate command which prompts you for all necessary information. Items are sent to idents [or .lastname].

```
BASE C: Goto (subsystem) C: Sendmail OK: <CR>

SEND C: Interrogate OK: <CR>

(distribute for action to:) T: JMB,FEEDBACK,.WOLD <CR>

(distribute for information-only to:) T: JCN <CR>

(title:) T: YOUR EXAMPLE <CR>

(type of source:) C: Message T: TYPEIN <CR>

(show status?) Y/N: Y  [The status typed by the system:]

TITLE: Your Example
AUTHOR(S): jhb
DISTRIBUTE FOR ACTION TO: jmb feedback paw2
DISTRIBUTE FOR INFO-ONLY TO: jcn
MESSAGE:  [Typein of message will be repeated]

(send the mail now?) Y/N: Y [for yes]

Completed

SEND C: Quit OK/C: <CR>
```

To send a statement that is already online, type an S for statement for the type of source in the Interrogate command. If the statement is in your current file, you will need to address its statement number. If the statement is not in your current file, you will need to include the filename followed by a comma before the statement number.

```
(type of source:) C: Statement A: FILENAME, STATEMENT NUMBER <CR>
```

To send a file, use File instead of Message for the type of source:

```
(type of source:) C: File A: FILENAME, <CR>
```

[Optional]

```
(type of source:) C: Branch A: FILENAME, STATEMENT NUMBER <CR>
```

To read Sendmail items you've received:

The mail box for Sendmail items is in your initials file under a statement which begins "(Journal)". The initials file automatically becomes your current file when you type NLS, and is named after your initials. It must be your current file in order to read your mail with the following command:

```
BASE C: Print C: Journal (mail) OK: <CR>
```

[Optional] To empty your mail box use the Delete Branch command. The Journal will automatically create a new mail box for the next delivery.

[Optional] Communicating in TENEX: SNDMSG

To get back to TENEX:

```
BASE C: Goto (subsystem) C: Tenex <CR>
                                        [You will be able to get
                                        back to NLS by typing
                                        QUIT <CR>]
```

To send a Message in TENEX:

The following is an example of sending a message using the Sndmsg command which prompts you for all necessary information. Items are sent to directory names.

```
@ SNDMSG <CR>

(To (? for help):) BECK,FEEDBACK,ALLEN <CR>

(cc (? for help):) BAIR <CR>

(Subject:) YOUR SUBJECT <CR>

(Message ? for help): TYPEIN            [Backspace to correct
                                        errors. You may include
                                        CR's in the TYPEIN.]

<CTRL-Z>                                [To end the message]

(Q, S, ?, carriage return:) <CR>        [To send the message]

@ QUIT <CR>                             [To go back to where you were in TNLS]
```

To abort a message before sending it type a <CTRL-C>

[Optional] To send a message to someone at another computer:

```
(To (? for help):) BECK@@ISIC <CR>

(cc (? for help):) BAIR@@ISIC <CR>

  [Three @'s will be printed]
```

To read new messages in TENEX at Office-1:

```
@ MESS <CR>
```

Communicating in TENEX: LINKING

To type directly to another user, "link" in TENEX.

[First use the following to find out if the user is logged in and if it is ok to link to him.]

```
@ WHERE <SP> DIRECTORYNAME <CR>
```

[Do not link when user is in SNDMSG, OUTPRC, or NOUTPRC]

[OR:]

```
@ SYSTAT <CR>
```

[To get a list of everyone logged on]

[THEN:]

```
@ LINK <SP> DIRECTORYNAME <CR>
```

[Type a semi-colon (;) and then type your comment. End the comment with a <CR> then wait for the other person to respond in the same way. If your comment reaches 3 lines long, type the <CR> and then another semi-colon immediately to start a new comment. If someone links to you do the same thing.]

```
@ BREAK <CR>
```

[To break the link]

6. TROUBLE SHOOTING AND ASSISTANCE

If you can't remember what to do, try typing ? for a list of all the possible command words.

If nothing is happening and you want to check to see if the system is up, type two <CTRL-T>'s. Note the words RUNNING or WAIT -- the system should be either running or waiting for you. No bell means something is down.

Call SRI-ARC, [415 326-6200, ext. 3630]
Or link to Feedback, Bair, Wold, Allen, Hysmith or Beck at Office-1

If you have any complaints, questions, problems, or suggestions send them via SNDMSG or Sendmail to FEEDBACK (it's both an ident and a directory name). Response should be no later than 1 working day.

If your connection is broken:

Re-dial the phone, type E and @ 0 <SP> 43 <CR>

To check if you are detached, use the where command:

```
@ WHERE <SP> DIRECTORYNAME <CR>
```

If you are detached, instead of logging in, attach to the old job:

```
@ ATT <SP> DIRECTORYNAME <SP> PASSWORD <SP> <CR>
<CTRL-O>                      [To wake up NLS if that's
                              where you were, or if that gets
                              no response:]
<CTRL-C> NLS <CR>             [To start over again]
```

If you accidently delete a file: [Before you log out]

```
BASE C: <SP>Undelete C: File T: FILENAME <CR>
Undeleted files are:
(FILENAME)
```

If you get an error message such as Illegal Instruction and you can't continue working in NLS, type:

```
<CTRL-C>, RESET <CR>, NLS <CR>
```

[Use <CTRL-C> only in emergencies to get to TENEX, or to abort a sendmessage.]

PRACTICE

In addition to trying each command, there is an instructional document called "INTRODUCTORY TNLS TUTORIAL" which can be used for practice. It shows the commands used for writing, editing, and distributing a memo. It is designed to be used by a person sitting at a terminal trying each command as it's introduced. Practice is necessary to learn NLS, either through regular use or through a practice document like this one. We recommend at least 5 hours a week until the next level course.

INDEX FOR THIS COURSE

COMMUNICATING

- Sendmail 14
- Sndmsg 16

CONTROL CHARACTERS

- <CTRL-A> = BACKSPACE CHARACTER 7
- <CTRL-C> = TO ABORT NLS OR SNDMSG TO RETURN TO TENEX 16
- <CTRL-E> = CONTINUE TO INSERT STATEMENTS 7
- <CTRL-O> = STOP PRINTING 10
- <CTRL-T> = SYSTEM STATUS 18
- <CTRL-V> <CR> = CARRIAGE RETURN [formatting] 13
- <CTRL-W> = BACKSPACE WORD 7
- <CTRL-X> = ABORT COMMAND 7
- <CTRL-Z> = TERMINATE A SNDMSG 16

EDITING

- Copy Branch 13
- Copy Statement 12
- Delete Branch 13
- Delete Statement 11
- Insert Text 11
- Move Branch 13
- Move Statement 11
- Replace Statement 11
- Substitute Text In Statement 11

FILES

- Create File 6
- Delete File 12
- Jump Link 6
- Show Directory 6
- Update File 12

IDENTS 14

LEVELS 8

LINKING 17

LOGGING IN AND OUT (TENEX and NLS)

- Login 4
- Goto Tenex 16
- Logout 5

PRINTING

Parts of a file 10
Mail 14

PROMPTS

A: = ADDRESS, usually a statement number 7
C: = Command 5
L: = LEVEL-ADJUST 7
OK: = Is this OK? 5
T: = TYPEIN 5
V: = VIEWSPECS 10
Y/N: = Type y for yes or n for no 11

TYPING IN TEXT

Insert Statement 7
Insert Text 11

< AJOURNAL, 38253.NLS;1, >, 27-DEC-76 15:31 XXX ;;;; .HJOURNAL="RPU 27-DEC-76 14:42 38253"; Title: .H1="Idea for a text edit and markup system"; Author(s): Ronald P. Uhlig/RPU; Distribution: /BR([ACTION]) JCN([ACTION]) RLL([ACTION]) DJF([INFO-ONLY]) STEF([INFO-ONLY]) ESV([INFO-ONLY]) PAUL([INFO-ONLY]) FGB([INFO-ONLY]) ; Sub-Collections: NIC; Clerk: RPU; .IGD=0; .SNF=HJRM; .RM=HJRM-7; .PN=-1; .YBS=1; .PES;

In a discussion I had recently with Dave Farber, he identified a need which I have frequently had, and which should be easy to satisfy with a small nls subsystem. Perhaps one already exists and I don't know about it.

The need is for a way for multiple people to work on writing and rewriting a document and easily identify what "the other guy" has said. We can easily make corrections to draft documents now, but it is not quite so trivial to find out what was in that spot before the correction was made. Dave commented that in working on a paper with a colleague at another geographic location, they had to resort to sending each other marked up hardcopies of the text. What was specifically needed was some way to get across the information that you have when you see a page of text in front of you with some text marked out, and the replacement text written in over it. I tend to do that now by some fairly tedious methods, such as going in to an nls file and enclosing my suggested changes in brackets, with several *'s to highlight the proposed changes.

I suggest that someone who knows some l10 (I don't) write a small program which will allow one to take the material "bugged" (text, word, character, or any other nls structure) and mark it in some way (enclose it in square brackets?), and then insert in a line immediately above it the replacement string. A companion L10 program is needed which will allow the colleague to say "I accept that change", and have it take effect as a normal nls change.

To be more explicit, I'll give a simple example. Suppose the text I want to change is:

   "The quick brown fox jumped over the lazy dog."

I want to insert the word "agile" in front of the word "quick", and I want to change the text string "brown fox" to "green turtle".

So, working in my edit program to markup the text (let's call it MARKUP), I would give the command "Insert Word (to follow)" and I would bug the word "The".

Still working in the MARKUP subsystem, I would give the command "Replace Text" and bug the "q" for the starting address and the "x" for the end adress. Then I would type in the text "green turtle".

Now, I would return to BASE and when I printed out the statement I would see:

```
       agile   green turtle
   The ^ quick [brown fox] jumped over the lazy dog.
```

Would ARC be interested in putting such a system together and how hard would it be?

Ron Uhlig

< AJOURNAL, 38335.NLS;1, >, 4-JAN-77 05:35 XXX ;;;; .HJOURNAL="STEF 3-JAN-77 15:58 38335"; Title: .H1="Blue Pencil Subsystem Suggestion"; Author(s): Einar Stefferud/STEF; Distribution: /BR([ACTION]) RPU([ACTION]) RLL([INFO-ONLY]) PAUL([INFO-ONLY]) ESV([INFO-ONLY]) DJF([INFO-ONLY]) JBP([INFO-ONLY]) FGB([INFO-ONLY]) ; Sub-Collections: NIC; Clerk: STEF; .IGD=0; .SNF=HJRM; .RM=HJRM-7; .PN=-1; .YBS=1; .PES; Origin: < NMA, BLUE-PENCIL.NLS;1, >, 3-JAN-77 15:49 STEF ;;;;####;

1. TO: Uhlig
2. FROM: STEF
3. CC: RLL ESV DJF BR PAUL JBP FGB
4. SUBJ: Blue Pencil SubSusytem Ideas
5. REF: <38253> <38318>
6. TEXT:

It is clear that in the long term we need something very sophisticated, but I doubt that we are in any position now to design and implement it. What I suggest then, is something that will let us explore the area, and have a useful tool in the meantime.

I suggest that a small effort be allocated to mapping some of the common BASE Subsystem Commands into a new Subsystem to be called Blue-Pencil (or something that will yield a unique Herald for NLS). This new subsystem would use special delimiters, such as <<>> and [[]] to identify <<insertions>> and [[deletions]]. Maybe it would be easier and less ambiguous to use <*...*> and [*...*], or what ever? My idea is that Insert Text (to follow) would do what it does in BASE, but would include the appropriate brackets. Then Delete Text would not delete the identified text, but would enclose it in the "delete" brackets instead. And, Replace Structure would work the same, in that it wuld "delete-bracket" the old and "insert-bracket" the new, and leave both. Transopose would require some thnking to make sense of it in this new Blue-Pencil Subsystem.

With NLS, this would not cause much trouble with typed out text, since it is not kept in "line" format internally. The NLS Print Structure commands would not even know anything had been done. The resultiant text from Blue-Pencil Commands would simply be compatible with the current NLS system in all regards.

Then, in addition, we would need some commands to accept and reject the Blue Pencil Marked text. This could work in normal NLS form such as Accept Mark (at) A: so that TNLS users could expect to "see" where the command would take effect by use of the pointer with "/" as in the BASE Subsystem.

I suggest this as an initial step toward what Ron suggested, becauuse it will be easier to use with TNLS and with DNLS, and because it should be very cheaply done, and give us the kind of experience that we need for now.

Best, Stef

Summary of 'Goals' meeting

This expresses my current view of ARC goals and needs, as influenced by recent 'Steering Committee' discussions.

Summary of 'Goals' meeting

An exhausting (but not yet exhaustive) discussion Thursday afternoon Jan 20 had the following results: 1

ARC's Meta-goals -- 1a

Social Utility
Survival
Reputation for recognized quality
Coherence 1a1

(General consensus that the above, with possible variations in priority or interpretation, have been and should continue to be basic motivating goals, for ARC and perhaps for most organizations). 2

ARC's Goal (specific target, relatively unique to us) -- 3

To Pursue/Promote/Do R&D upon Deliver -- "Make Happen" -- evolving, often computer-based, aids to the productivity and capacity of people and groups who work with information and knowledge on important, complex problems. 3a

(General consensus that the above statement captures the idea of ARC's past orientation, and one that we can all agree with and focus upon in the future, as our medium for achieving the meta-goals.) 4

Strategy for achieving the above goal: 5

Historically: 5a

Trial tool construction and evolution
Bootstrapping 5a1

Last three years: 5b

Create the "Bootstrap Community," including ARC and a growing number of outside organizations. 5b1

(General agreement that this is a reasonable-appearing strategy, with some questioning of its optimality -- e.g., has emphasis on Bootstrapping led to an inbred neglect for Analysis and lack of attention to Alternatives? -- and of the strategie's viability -- see below.) 6

Implementation plan (way of supporting the strategy): 7

Historically: 7a

R&D contracts, with heavy reliance on one source (ARPA) 7a1 last three years: 7b

Split into Development and Applications. (Here the discussion became a bit more lively; let's consider D and A separately.) 7b1

DEVELOPMENT I (BR) had somehow gotten the impression that Development was chartered only to develop changes & improvements to the NLS system -- i.e. clean up NLS 8.5, complete NLS 9, prepare new subsystems, design NLS 10, etc. -- thereby contributing to the evloution of the Bootstrap Community; In other words, adding specific tools to the AKW presently in use. My tentative conclusion had been that this effort is doomed to failure, because -- since ARPA's withdrawal -- no one will fund such pedestrian modifications of existng technology. Therefore I thought a major change in the charter, and thinking, of the Development people would be necessary. Instead, I was delighted to learn that ARC (as represented in this meeting) views Development as free to work toward the Goal ("evolving aids to productivity...") by working on the systems of ten and twenty years from now, as well as next year's NLS. Such far-rangng work is important for the Goal and Meta-Goals, and is research of the type that NSF, ARPA, and others would like to fund; we just need to get our ideas and proposals into shape. However, a strategy highly dependent upon the Bootstrap Community may not have enough room for such activities, so perhaps the strategy needs restating/broadening. (Also, JCN and RLL thought that adequate contract support for adding bells and whistles to existing NLS version could also be obtained. I'm skeptical, but willing to try whatever leads they can suggest in parallel with broaden, back-up promotion.) 7b2

APPLICATIONS: ARC's proposal and promotion effort has gradually been shifting its emphasis from Participation or Membership in an experimental community, to service of current value to the Utility subscribers. Several ARC staff, including Dave Potter and me, are still unclear about which way we have been trying to lean. The meeting, and, I believe, most of the present subscribers, say SERVICE! I think this is the right answer, because there is no significant market for "Participation;" the real world of corporate finance does not support visionary experiments without much more-specific return-on-investment arguments than ARC has thus far been able to marshal. (JCN, DCE, and some others still seem to think there is a large market for participation in the experiment, and may want to pursue growth of an AKW community independently of short-term benefits to its members. Again, I encourage them to try -- this activity could be Doug's principal responsibility -- but in parallel with other ARC service activities that have more-clearly identified markets.) In the Service area, however, much needs to be done: 7b3

We are too expensive 7b3a

We need more-modular systems -- at least conceptually and in our promotion and documentation, and eventually in the implementation itself -- to address specific markets: text editing, document preparation, file maintenance, communication, front-end systems, etc. 7b3b

We need to monitor alternative systems more closely, and see what features users like; we should understand the favorite capabilities of TVEDIT, PUB, WORD/ONE, etc., whether we like them or not, and then either add those features to the appropriate parts of NLS, or clearly know why not. 7b3c

We need to be more responsive to user needs: better able to configure, supply, & maintain hardware and computer service; better prepared with user-oriented documentation; better organized to evaluate and react to FEEDBACK; and so on. 7b3d

7b4

Conclusions: ARC's Meta-goals and Goal are clear. So is a general inclination for Development to be more longer-range Research oriented, and Applications to be more short-term Service oriented, than they are right now. In the next meeting (tentative time: Feb. 3,, 2pm) we can begin to really get down to specifics: how to allocate our limited resources, set priorities on NLS 8.5 fixes vs NLS 9 completion, selection of target markets and how to evaluate them and promote to them, etc. Please send me specific comments/suggestions/issues that need to be addressed. The general theme is, how to reexpress our strategy for achieving the ARC goal and how to get support to implement the new strategies. 8

RLL 3-FEB-77 12:15 38808
DRAFT: outline of training section for CHASE proposal
Location: (FJOURNAL, 38808, 1:w)

<<<<<<<<<<<<<<<<<<<<<< Text of Cited Document Follows
>>>>>>>>>>>>>>>>>>>>>

A. Introduction
  1. Need for architect
    a) Official point of contact
    b) Coordinator of training, etc.
    c) Resource for SRI to discuss problems about contract
    d) Source of information about organization
    e) Acts as in-house consultant after SRI leaves
  2. Need for SRI coordinator
    a) For training
    b) Main point of contact
    c) Resource for consulting on various applications
  3. Formal courses are flexible and customized to specific application areas. Low instructor to student ratio.
  4. Terminals for each student.
  5. Practice sessions with coaching
  6. Expect practice beyond scheduled classes.
  7. Remote help via phone & computer.
B. On site application & training coordinator.
  1. Our experience why this is needed.
  2. List of responsibilities.
    a) Coordinates all SRI visits and training schedules.
    b) Conducts management introduction.
    c) Assets in coaching
    d) Advises on application areas indicating both system and social procedures.
  3. Schedule of visits.
  4. Background & level of typical coordinator.
C. Special Management Seminars
  1. The need - the originators of textual input and, at least, one level of management above should be well aware of the overall capabilities of the system, potential for the future, and understanding of the project.
  2. Schedule of Seminars.
D. Training for terminal operators
  1. What is to be taught @ session.
  2. Separation into sessions.
  3. Schedule duration, expected user participation.
  4. Separation between sessions.
  5. Coaching.
  6. Specific application orientation.
  7. Remote training via phone & computer
    - need for computer access
  8. Typical background of SRI personnel performing these tasks.
E. Training for system operators
  1. Initial training by DEC.
  2. What we assume as the level of operators.
  3. Schedule of training.

```
    4.  Remote coaching (need for computer access from member).
    5.  Typical background of personnel.
  F.  Training for programming support personnel
    1.  Initial training by DEC.
    2.  What we assume as the level of operators.
    3.  Schedule of training.
    4.  Remote coaching (need for computer access from member).
    5.  Typical background of personnel.
  G.  Continual training
    1.  Reasons for training beyond initial start-up.
      a) Future holds many more applications.
      b) More direct involvement of originators.
      c) Need for training of trainers.
    2.  Recommended level of on-going training.
      a) Terminal operators.
      b) System operators.
      c) Programmers.
      d) Managers, etc.
    3.  Reference cost for such additional services in cost section.

  <<<<<<<<<<<<<<<<<<<<<<<<<< End of Cited Document
  >>>>>>>>>>>>>>>>>>>>>>>>>>
```

39000

EKM GPA DHB DVN RA3Y JBP RLL PWO JHB 28-FEB-77 13:36 39000

Comments on Implementation Plan for Project IMPACT

February 24, 1977

GENERAL COMMENTS

Introduction

In December 1976, the headquarters Electronic Systems Division of the Air Force Systems Command distributed the Implementation Plan for Project IMPACT, dated 15 October 1976, to solicit comments and recommendations on the proposed approach.

Project IMPACT (IMproved Administrative Capability Test) entails the planning, design, implementation, and test of a prototype automated office system designed to demonstrate the optimum cost-effective set of personnel, procedure, and equipment of modern office systems technology that will provide the greatest benefits in manpower reductions and economics through increased efficiency.

Due to the broad ranging capabilities, experience, and on-going research and interest in the subject among a number of distinct organizational groups within SRI, a coordinated approach has been undertaken. The following SRI departments have participated in the development of the response:

Augmentation Research Center of the Information Science and Engineering Division

Electronic Industries Department of the Management and Economics Systems Division

Information Science Laboratory of Information Science and Engineering Division

Information Systems Management Group of the Management and Economics Systems Division

Manufacturing and Distribution Systems Department of the Management and Economics Systems Division

Telecommunications Science Center of the Information Science and Engineering Division

We believe that these groups draw on an unmatched range of text and data processing, management, and telecommunication expertise.

SRI's comments are divided into three parts. The first, GENERAL COMMENTS, expresses our overall agreement with the goals and methods of the plan and offers specific suggestions that apply to the plan as a whole. The second, SPECIFIC COMMENTS, responds to specific paragraphs in the RFC by number. In the third, SRI BACKGROUND, we will briefly inform AFSC about SRI so that readers will understand the kind of interest and experience that lie behind these comments and to establish our position as a possible future contractor.

Overview

"Office automation" is still in an early phase of integration. A broad set of technology and tools is currently available, but they have not been coherently applied to the office environment. Concrete progress is measured chiefly in the use of centralized dictation systems, automatic typewriters, display editing work stations, and text handling and composition services. For very practical reasons, equipment manufacturers have tried to develop products with little thought of standardization, total system planning, or consideration of the larger context in which the office operates. Individual tasks have benefitted from this technological progress, but the office environment has not been rationalized.

At the same time as automation efforts are directed at office functions, a wealth of research and development is progressing in allied areas. Large-scale projects in teleconferencing, document and text production, electronic mail, man-machine decision systems, and novel technologies, such as voice data input and alphanumeric hand-held wands, portend the reduction of mechanical and clerical tasks and the streamlining of professional work patterns. Further, new devices are becoming available that stem from the confluence of existing technologies, such as very high-speed, high-quality facsimile transmission augmented with micrologic and memory to facilitate the storage and forwarding of images.

While a number of these developments do not directly address traditional office functions, when considered collectively they open the door to a revolution in the world of work. It is difficult, however, to unify these scattered developments or evaluate them in terms of pragmatic cost benefits.

There is strong motivation for change. The cost of personnel, the largest single office expense, is continually rising. The quality of available labor is steadily declining, and it is becoming increasingly difficult to attract people to clerical work. At the other end of the spectrum, managers must make faster decisions in a more complex environment and considering higher volumes of relevant information. While automation has been beneficially applied to the industrial and agricultural sectors, increasing productivity twelve fold over the last 50 years, clerical work has seen production improve on the order of 1.5 to 2 times, and no clear statement can be made about the increased productivity in decision making.

.Grab=2;To remain cost effective, many organizations will have to change the way they work. However, change is frightening to those it affects, and fundamental change is almost by definition traumatic. SRI is pleased to see that AFSC recognizes that automating office systems involves difficult organizational changes.

People who work in organizations resist change because it challenges work habits, perceived status, and the comfort of established routines and relationships. Therefore, if all of the impetus comes from supplier developed technology with a notable absence of a systems approach, integration of automation into the office will be slow, delaying many potential benefits. The willingness of AFSC to consider a potential thorough restructuring of its operations to take maximum advantage of available and near-term technology could lead to a major breakthrough that would benefit other governmental and private organizations as well as AFSC. Project IMPACT should be designed to provide that breakthrough.

During the last two years, commercial and government sectors have become increasingly aware that office productivity is improved and costs are reduced most effectively when text processing and data processing are handled in the same coherent, computer-supported environment. For example, a recent article in Word Processing World predicts that systems of the next five years will offer video terminal text editors that are able to perform accounting and other data processing tasks as well as support for dialog and electronic distribution of documents through remote communication facilities.

We believe that certain other Air Force activities not mentioned in the RFC are highly relevant to this study. We recommend that a task be added to the eventual RFP to consider them. In addition to documenting existing office systems and studying automated systems now in use, the project staff should also consider the data processing requirements of AFSC and inform themselves about the Phase 4 Base Level Data Automation Standardization Program. This program is the design and implementation plan through 1985 for all Air Force business data processing computer and systems support. The project should include at least a conceptual approach for integrating the data processing functions with text processing and management information support to ensure that the system is able to evolve and assimilate new technology without a radical change in design.

* It is essential that the system chosen for the prototype be modular in design and contain features that enable it to evolve easily to accomodate both changing needs and new applications. The system should be designed to interact with and take advantage of new commercially available technology in both software and hardware.

.Grab=2;There are several systems presently in use on an experimental basis at Rome Air Development Center (RADC) that should also be studied. Of particular interest to an integrated office support system is an Air Force project financial management system that operates with a text processing and cataloging system.

The Document Production and Control System Design Study, also a current project at RADC, should be examined by the IMPACT team. This project encompasses an in-depth study of the document production process, a survey of existing systems from stand-alone devices through large-scale computer-based systems, a conceptual design of the ideal system, and an implementation plan.

COMMENTS ON SPECIFIC ITEMS IN THE RFC

This section comments on the RFC, paragraph by paragraph. We hope to contribute to AFSC's thinking on a variety of points. Many of our comments fall within these three areas:

Vigorous support for AFSC's emphasis on personnel and human engineering as opposed to mere hardware integration.

Integration of the prototype system with other Air Force data processing planning.

The need for detail in spelling out several tasks so that prospective contractors may realistically estimate their costs, particularly in the area of management information systems.

SECTION 1: Introduction and Background

1.1 The advantages to be gained are more complex and broader than reduction of staff. The RFP should stipulate and formulate the following distinct objectives:

Increased quality.

Increased productivity by reduction of staff.

Increased productivity by increased throughput.

Decreased exposure to information leaks, data loss, etc.

Increased capacity to perform new and additional tasks.

1.1.2a The study should consider that present functions may no longer be required and new ones may become necessary. The contractor should be alert to identify new as well as existing AFSC office management and support functions.

1.2.a Over the past 12 months the 'state-of-the-art' in text processing systems has changed very rapidly and will continue to do so over the next several years. We suggest that the RFP clearly spell out the relevant word and information processing techniques or suggest criteria by which the contractor should choose them.

2.1 'Centralization of services' has worked well only in a very few industries (e.g., large insurance companies). The study should not be committed to this concept a priori.

This seems to put too much emphasis on 'word processing' and not enough on the more costly office tasks such as distribution, dialog support, record keeping, communication, and data base and data processing tasks. Commercially available systems have heretofore given little or no attention to these aspects of office management.

Productivity increases of '500%' have occurred only in very narrowly defined work categories and have relatively small impact on the overall cost and effectiveness in getting the whole job done. Expectations of productivity improvements should be more realistically established.

2.2 The emphasis on the need for "personnel and procedural changes" is very good. The project should include contractor personnel attuned to and capable of dealing with these issues.

SECTION 2: Objectives of the Program

2.0 This section very correctly indicates that some consideration should be given to office tasks other than text processing. However, the tasks should be clearly specified.

Performance of analysis and cost tracing are extremely important but difficult and expensive to achieve, unless the prototype system has been designed to provide them. Experience has shown the impossibility of establishing retroactive measurements.

2.c Repertoires of automated management science tools have generally proved very expensive and of doubtful value. This failure is in large part due to the lack of a coherent user interface to the systems and tools that are now available. The model should include provisions to integrate such tools from commercially available sources with a coherent user interface. This necessity further points out the need for the contractor's team to include members sensitive to personnel issues and human engineering.

3.a The study should emphasize that the design must leave room for evolutionary development of newer tools. The field is developing and changing rapidly, and AFSC must be careful that the short time constraints of the project do not dictate limited solutions. It has been the experience of several government agencies that the use of new technology and improved systems sometimes shows cost benefits only after a long period of use.

3.b Cost benefit techniques and personnel reductions for various organizations may differ. The RFC quite correctly points out that a "broad range" of functions must be considered for the AFSC prototype system.

3.c As noted above, the application should include management science, administrative support, and other related topics, as well as word and information processing.

SECTION 3: Schedules and Tasks

1.0 The time scale seems short for some aspects of the project, although the overall set of tasks and phases is well thought out. In particular, the overlap between the "design of the prototype office systems" and the "implementation of the prototype office systems" seems inappropriate. Another trouble spot is the overlap between the "development of test plans and procedures" and the "test and evaluation" phases. It also seems wise to allow more time for analysis and reporting at the end of the project since the experience in constructing the prototype may be at least as valuable as the prototype itself.

.Grab=2;The RFP should indicate, at least roughly, the level of effort in terms of full time staff equivalents. We have assumed that staff needed for the task described in Section 3 is about three to four full-time staff members, excluding any personnel associated with the prototype system hardware or software. If any software effort is needed to interconnect the off-the-shelf portions of the prototype system, then this estimate should be revised.

1.c It is not clear what the MIS plan is. Is it a general Management Information System, or is it a plan to keep management informed about Project IMPACT? The definition must be clarified to enable potential contractors to estimate costs of designing and building the system.

2.0 In a number of situations it has been observed that system builders work faster and better when they are forced to depend on their own tools. AFSC should suggest that the personnel involved in the study should as far as possible use the tools they are proposing for the prototype office for their own work to ensure effectiveness and understanding on their part.

2.1.1 The examples of functions include a very wide range of potentially highly complex activities. For example, the item "Cost and schedule management" could mean complex capital budgeting programs and a full blown computerized PERT system. It will be difficult to estimate the cost of complying with this item without further definition.

> Personnel management is also mentioned here. Is this a complete, automated AFSC personnel system? If so, how is this expected to fit in with existing and planned Air Force personnel data processing?

2.1.3 As in the case of functions, it may be a formidable task to identify and document in detail the information needs and requirements of all offices. This task should be very broad rather than detailed, at least until a selected candidate office is established.

2.1.4 We agree that this task is important to ensure the successful transfer of new tools and techniques into an organization.

2.2.1 This task is critical to the proper selection of the prototype system and represents an enlightened view of the potential payoffs. However, the contractor must be cautious to avoid sacrificing long-term benefits for near-term payoffs because of constraints imposed by the anticipated program test. The RFP should specify proper consideration of longer term cost benefits.

2.2.2 .Grab=2;Communications among offices is very important to this study. We advise selecting two offices geographically separated but in frequent communication or, through some other strategy, providing a realistic model of the communication function.

SRI experience with introducing similar systems into office environments of NLS Utility users (see section 3) for the last three years has shown that success depends on finding someone in the user organization who grasps the system clearly and actively promotes its use. One criterion of selection of an office for prototype installation should be the presence of such a person.

2.2.3 If we are to judge by allied areas, the office environment may be facing many more changes in the years ahead. One criterion for selection of the prototype system should be that it helps people cope with the future. Does its philosophy and the flexibility of its capabilities prepare the office staff to face as yet unforeseen changes more gracefully?

2.3.1 We heartily agree with this document's emphasis on "combining capabilities into integrated office systems."

The study of existing systems should include, as far as possible, information on real applications. This is important because manufacturers' literature often claims that capabilities are more effective than they turn out in practice. On the other hand, real use of systems often unearths capabilities not imagined by their designers.

2.3.2 It is unclear whether this paragraph refers to a relatively straightforward Project IMPACT reporting system or a full-scale management information system. This must be clarified before realistic estimates and bids can be made.

2.4.2 Very good. Careful consideration has been given in the RFC to training users. This should be expanded to include a specific plan for training Air Force trainers.

2.4.3 Test plans and procedures should appear earlier in the schedule and should be carefully considered throughout the design phase. Few existing systems provide adequate measurement facilities such as counts of words, pages, or even provide basic accounting data, much less a thorough analysis.

The test plan should consider the difficulty of measuring the performance of capabilities that did not previously exist.

2.5.2 .Grab=2;This task could be very time consuming, impossible, or fairly easy, depending on what detailed cost, resource, and manpower figures are already available or obtainable. The RFP should spell out this task more clearly.

4.1.3 The questionnaire approach to quantitative measurement frequently yields questionable results. Questionnaires may be successfully used to obtain subjective evaluations of system performance and user preferences. They may also be used to collect information about the user population. Such information should be supplemented by objective, controlled experiments in which many performance measures are obtained through online monitoring of tasks supplemented by manual data collection. The combined data from all sources should then be subjected to rigorous statistical analysis.

4.3.2.a.1 Why is this limited to stand-alone systems? Why not include users of shared-logic and time-sharing systems?

4.3.2.g Does this mean that new software and hardware can be recommended in the design for future implementation?

4.4.2.a.1 Current software to support office functions exists only in nonintegrated, isolated packages. These packages offer no coherent user interface and have been implemented for widely different hardware architectures. A minimum software effort should include provisions for software that will enable users to make use of a variety of packages with a single command language. This may not be a trivial programming task.

SRI'S BACKGROUND

General Capabilities of SRI

SRI is an independent, nonprofit corporation that performs a broad spectrum of research, development, and services under contract to business, industry, and government. Most of SRI's work is directed toward problem solving rather than research in the abstract. The staff of SRI numbers over 3,000. More than 500 SRI staff members hold Ph.D. degrees, over 600 hold Master's degrees, and approximately 800 hold Bachelor's degrees. Facilities at SRI's main offices include extensive data processing, library, and laboratory support. The comprehensive technical libraries are well supplied with literature in the fields of document generation and handling systems analysis, computers, coding, and management control systems.

Research operations at SRI are organized into seven divisions representing major disciplinary fields. Overall supervision of research is vested in the Office of Research Operations which reports directly to the Office of the President. Both formal and informal arrangements of long standing exist to facilitate interdisciplinary research and development among the divisions and their subgroups. Work related to Project IMPACT has taken place in the Management and Economics Division and in the Information Science and Engineering Division.

Management and Economics Division

The Management and Economics Divisionn is concerned with solving problems for business, industry, and government through the application of modern management and systems techniques. The division has over twenty years of experience in bridging gaps in effectiveness and efficiency and keeping client management current with useful, practical systems that support all aspects of management from research and development through operation and maintenance. Strong emphasis is given to studies of the interaction of technology and business and the resultant problems of managing technological change.

The Electronic Industries Department.Grab=4;

The Electronic Industries Department has conducted extensive research in the area of office automation from the marketing and technical viewpoints as well as the motivational and organizational perspectives. During the course of our multi-client study covering business equipment, of which there were 30 sponsors, several hundred personal interviews were conducted, supplemented by over 5,000 mailed questionnaires covering the fields of word processing and small business computers. In addition, a number of proprietary studies have been conducted on a wide variety of office automation products and markets. Long range planning studies have covered future business ccmmunication equipment, and other recent guideline report from SRI's Business Intelligence Program was directly concerned with the office of the future.

Information Systems Management Department

This department provides management assistance in the area of business electronic data processing and related planning and management requirements. The department has extensive experience in the development and implementation of techniques and methods to improve the effectiveness of EDP requirements planning, systems development, computer operations, equipment utilization, EDP cost benefits and systems analysis. Assistance has been provided to many organizations in the planning of their overall systems development, personnel training, project management, and EDP organization development.

Information Science and Engineering Division.Grab=4;

The activities of the Information Science and Engineering Division are carried out in four laboratories and four research centers: the Augmentation Research Center, the Information Sciences Laboratory, the Engineering Sciences Laboratory, the Sensory Sciences Research Center, the Transportation Research Center, the Artificial Intelligence Center, the Electronics and Bioengineering Laboratory, the Computer Science Laboratory and the Telecommunications Sciences Center. Each of the laboratories is composed of a number of groups with complementary interests and skills.

Information Science Laboratory.Grab=4;

ISL personnel plan, design, and develop computer systems for business and industry, and carefully mold them to meet management and operational needs. Current and long-term company goals, user requirements, corporate growth factors, functional EDP needs, applicable technology, and cost effectiveness are taken into account, so that the system will become an integral part of the organization and enhance the productivity of its users.

Their objective is to provide clients with the complete services essential to the achievement of a responsive, growth-oriented computer system. Actively involved in concept formulation, design, implementation, and evaluation of computer systems, ISL has engaged in the development of large-scale data processing systems in such diverse areas as banking, transportation, medical services, education, process control, computer-aided design, criminal justice, and commercial marketing services.

ISL work spans a wide spectrum of activities, from the preparation of performance specifications to the actual implemention of systems, with a dominating principle being the pragmatic synthesis of hardware, system software, and application software to form an integrated system that meets specific organizaional goals.

Augmentation Research Center

The Augmentation Research Center develops and applies a a broad-based computer support system that improves effective use of human capacities in environments such as offices. A brief description of some of the accomplishments of ARC over the past 13 years will attest to its leading position in the development of effective services for people working with office information.

Early explicit recognition of the potential that online computer and communication technologies have in areas outside of straight numeric or accounting computation in enhancing the effectiveness and efficiency of managers, scientists, technical writers, engineers, programmers, and their supporting staffs in their daily work.

Participation in the implementation of the ARPANET, a nationwide network connecting over 3,000 remote terminals to 150 different computers.

.Grab=2;Early explicit recognition of the importance to system building of an integrated system of text handling and system building tools.

.Grab=2;Pioneered the two-dimensional text work to be the foundation of an intelligent terminal system, and developed many highly interactive tools and concepts for working and browsing in the information that is characteristic of office work.

Pioneered input device and work station design.

Pioneered in high quality formatted publication of hard copy through line printers, typewriters, photocomposition, and Computer Output to Microfilm.

Pioneered the concept of an integrated coherent workshop of many office tools with a uniform user interface.

First with a comprehensive system for electronic mail control, addressing distribution, delivery, individual and group identification, cross-linking, and indexing.

Over 100,000 hours of hands-on console experience with the use of NLS technology in daily work, both at ARC and at other sites.

.Grab=2;Recognition of the importance of integrating into the system building process mechanisms for studying and facilitating technology transfer, including establishment of training and other application support services.

For the last three years, the Augmentation Research Center has offered a system of aids for information-based work through a time-sharing service, the ARC Utility. About 15 government and private organizations supporting about 300 users now subscribe. In many applications, the use of the Utility closely resembles the office environment envisioned in the RFC. Utility services normally include substantial training and applications consulting, which we believe gives SRI unique insights into the problems of implementing a plan such as IMPACT.

TITLE PAGE .IgD=0;.IgText=0;.Post=1;.Trun=All;.IgLS;
.H1Sw=0;.H2Sw=0;.H3Sw=0;.H4Sw=0;.SN=0;.SNFShow=Off;.YBS=0;.YBL=0;
.TM=25; .YBHJTM=13; .PES;

.GCR;

.SP=C;.FSw=0;Comments on Implementation Plan.GCR;for Project IMPACT.GYEL=5;.Gcr;February 24, 1977

The LEVLEFILTER Subsystem

The Design goal the LEVELFILTER subsystem is to facilitate the process of extracting summaries from NLS data bases by passing different content filters over different levels of statements in the data base. The original motivation for this came from a need JAKE had for extracting summaries from the NIC Resource Handbook. I generalized this, yielding the LEVELFILTER Subsystem. I view this as an intermediate tool, between content filters and the NEWRETRIEVE subsystem. It is easier to learn than NEWRETRIEVE and more powerful than content filters. Of course it is also harder to learn than content patterns and less powerful than NEWRETRIEVE. I hope the compromise proves useful to some. To try it: 1

Execute Programs Load Program xprograms,levelfilter.subsys, 1a

LEVELFILTER will aid you in constructing a single content analysis filter which effectively passes differing filters over different levels of statements in a file. I have constructed a simple (I hope) language for specifying how these patterns are applied to different levels. 2

I have defined an IF statement , the syntax is: 3

IF <PATTERN> THEN <VALUE> ELSE <VALUE> 3a

where: 3b

<VALUE> stands for one of the following 3b1

TRUE % Pass this statement % 3b1a

FALSE % Do Not pass this statement % 3b1b

TRUE,TRUE % Pass this statement plus all of its substructure% 3b1c

TRUE,FALSE % Pass this, but none of its substructure % 3b1d

FALSE,TRUE % Don't pass this, but pass all of its substructure % 3b1e

FALSE,FALSE % Don't pass this or any of its substructure % 3b1f

<PATTERN> stands for either: 3b2

a content pattern, exactly as you would type in in to the SET CONTENT <"Pattern"> TO Command., or 3b2a

an asterix followed by the name of an already loaded content analysis program, for example: 3b2b *myfilter 3b2b1

There are two other legal types of statements in the language. They are: 4

TRUE 4a

This means pass all statements at this level 4a1

FALSE 4b

This means do not pass any statements at this level 4b1

To use LEVELFILTER you simply arrange combinations of these three types of statements somewhere in a branch in an NLS file. You then GOTO LEVELFILTER and give the command DEFINE <"filter from branch at"> and either point at or otherwise address that branch. LEVELFILTER will then build you a content pattern which will apply each filter statement you have defined to all statements in a data base at the corresponding level. Thus the level one filter statement you define is applied to all level one branches in the data file, etc. Turning viewspec "i" on then turns this "virtual filter" on. You may of course use this in conjunction with any NLS command. 5

EXAMPLE: 6

The RESOURCE HANDBOOK data base is structured something like the following, it contains a branch for each host. Data elements are stored below the header statement for each element. 7

(SERVERS) 8

(OFFICE-1) STANFORD RESEARCH INSTITUTE
AUGMENTATION RESEARCH CENTER 8a

(FUNCTION) 8a1

(PERSONNEL) 8a2

(PROTOCOLS) 8a3

(SERVER) 8a3a

(USER) 8a3b

(NCP-INTERFACE-FROM-LOCAL-PROGRAMS) 8a3c

(HARDWARE) 8a4

(COMPUTER) 8a4a

(PERIPHERALS) 8a4b

(TERMINALS) 8a4c

(INTERESTS) 8a5

(DOCUMENTATION) 8a6

(REFERENCES) 8a6a

(ORDER-INFORMATION) 8a6b

(if pattern appears anywhere in branch)

The following filter branch would create a pattern which would list the personnel branch from each site: 9

TRUE % pass all level one statements % 10

TRUE % pass all level two statements (host name) % 10a

IF "(PERSONNEL)" THEN TRUE,TRUE ELSE FALSE,FALSE 10a1

The following filter branch would create a pattern which would list the computer at each site: 11

TRUE % pass all level one statements % 12

TRUE % pass all level two statements (host name) % 12a

IF "(HARDWARE)" THEN FALSE ELSE FALSE,FALSE % IF it says (HARDWARE) then consider substructure otherwise don't pass any substructure. In any case we don't want to print (pass) any level three statements.% 12a1

IF "(COMPUTER)" THEN TRUE,TRUE ELSE FALSE,FALSE % Pass only the branch (COMPUTER) % 12a1a

if "(computer)" # " ( )" (operating system)

The LEVLEFILTER Subsystem

This is the second version of documentation I have journalized, it corrects an error in an example - Please ignore the previous item

on Postel (SRI-ARC)
Dave Crocker (UC Irvine)
RFC 726, NIC 39237 (Mar. 8, 1977)
Obsoletes NIC 18492 and NIC 19859

Remote Controlled Transmssion and Echoing Telnet Option

1. Command name and code:

RCTE 7

2. Command meanings:

IAC WILL RCTE

The sender of this command REQUESTS or AGREES to use the RCTE option, and will send instructions for controlling the other side's terminal printer.

IAC WON'T RCTE

The sender of this option REFUSES to send instructions for controlling the other side's terminal printer.

IAC DO RCTE

The sender REQUEST or AGREES to have the other side (sender of WILL RCTE) issue commands which will control his (sender of the DO) output to the terminal printer.

IAC DON'T RCTE

The sender of this command REFUSES to allow the other side to control his (sender of DON'T) terminal printer.

IAC SB RCTE <cmd> [BC1 BC2] [TC1 TC2] IAC SE

where:

<cmd> is one 8-bit byte having the following flags (bits are counted from the right):

Bit	Meaning
0	0 = Ignore all other bits in this byte and repeat the last <cmd> that was sent. Equals a 'continue what you have been doing'. 1 = Perform actions as indicated by other bits in this byte.
1	0 = Print (echo) break character 1 = Skip (don't echo) break character
2	0 = Print (echo) text up to break character 1 = Skip (don't echo) text up to break character
3	0 = Continue using same classes of break characters. 1 = The two 8-bit bytes following this byte contain flags for the new break classes.
4	0 = Continue using same classes of transmit characters. 1 = Reset transmit classes according to the two bytes following 1) the break classes bytes, if the break classes are also being reset, or 2) this byte, if the break classes are NOT also being reset.

Value (decimal) of the <cmd> byte and its meaning:

0 = Continue what you have been doing

Even numbers greater than zero (i.e. numbers with the right most bit off) are in error and should be interpreted as equal to zero. When the <cmd> is an even number greater than zero, classes bytes TC1 and TC2 and/or BC1 and BC2 must not be sent.

1 = Print (echo) up to AND INCLUDING break character

3 = Print up to break character and SKIP (don't echo) break character

5 = Skip text (don't echo) up to break character, but PRINT break character

7 = Skip up to and including break character

Add one of the previous non-zero values to one of the following values, to get the total decimal value for the byte (Note that classes may not be reset without also resetting the printing action; so an odd number is guaranteed):

8 = Set break classes (using the next two bytes [BC1 BC2])

16 = Set transmission classes (using the next two bytes [TC1 TC2])

24 = Set break classes (using the next two bytes [BC1 BC2]) and the transmission classes (using the two bytes after that [TC1 TC2]).

Sub-commands (IAC SB RCTE...) are only sent by the controlling host and, in addition to other functions, functionally replace the Go-Ahead (IAC GA) Telnet feature. RCTE also functionally replaces the Echo (IAC ECHO) Telnet option. That is the Suppress Go-Ahead option should be in force and the Echo option should not be in force while the RCTE option is in use. The echo mode on terminating use of the RCTE option should be the default state, that is DON'T ECHO, WON'T ECHO.

Classes for break and transmission (the right-most bit of the second byte (TC2 or BC2) represents class 1; the left-most bit of the first byte (TC1 or BC1) represents the currently undefined class 16:

1: Upper-Case Letter (A-Z)

2: Lower-case Letters (a-z)

3: Numbers (0-9)

4: Format Effectors (<BS> <CR> <LF> <FF> <HT> <VT>)

The sequence <cr><lf> counts as one character when processed as the Telnet end of line, and is a single break character when class 4 is set. The sequence <cr><nul> counts as one character and is a break character if and only if <cr> is a break character (i.e. class 4 is set).

5: Non-format Effector Control Characters including <DEL> and <ESC>

6: . , ; : ? !

7: { [(< >)] }

8: ' " / \ % @ $ & # + - * = ^ _ | ~

9: <Space>

And Telnet commands (IAC . . .) sent by the user are always to have the effect of a break character. That is, every instance of an IAC is to be treated as a break character, except the sequence IAC IAC.

The representation to be displayed when printing is called for is the obvious one for the visible characters (classes 1, 2, 3, 6, 7, and 8). Space (class 9) is represented by a blank space. The format effectors (class 4) by their format effect. The non-format effector controls (class 5) print nothing (no space).

Initially no break classes or transmission classes are in effect.

Please note that if all the bits are set in a Telnet subcommand argument byte such as TC2 or BC2 then that byte must be preceeded by an <IAC> flag byte. This is the common convention of doubling the escape character to use its value as data.

Sub-commands (IAC SB RCTE...) are refered to as "break reset commands".

3. Default

WON'T RCTE -- DON'T RCTE

Neither host asserts special control over the other host's terminal printer.

4. Motivation for the option

RFC's 1, 5, and 51 discuss Network and process efficiency and smoothness.

RFC 357, by John Davidson, introduces the problem of echoing delay that occurs when a remote user accesses a full-duplex host, through a satellite link. In order to save the many thousands of miles of transit time for each echoed character, while still permitting full server responsiveness and clean terminal output, an echo control similar to that used by some time-sharing systems is suggested for the entire Network.

In effect, the option described in this document involves making a using host carefully regulate the local terminal printer according to explicit instructions from the remote (serving) host.

An important additional issue is efficient Network transmission. Implementation of the Davidson Echoing Scheme will eliminate almost all server-to-user echoing.

The option described in this document also requests that using hosts buffer a terminal's input to the serving host until it forms a useful unit (with "useful unit" delimited by break or transmission characters as described below). Therefore, fewer messages are sent on the user-to-server path.

NOTE: This option is only intended for use with full-duplex hosts. The Go-Ahead Telnet feature is completely adequate for half-duplex server hosts. Also, RCTE should be used in place of the ECHO Telnet option, i.e., the Suppress Go-Ahead option should be in force and the Echo option should not be in force while the RCTE option is in use.

5. Explicit description of control mechanism

User Terminal Printing Action and Control Procedure

Negotiate the use of the RCTE option. Once the option is in force the user Telnet follows the following procedure.

1) Read an item from the network.

   If the item is data, then print it and go to 1.

   If the item is a command, then set the classes and go to 2.

2) If the terminal input buffer is empty, then go to 3, else go to 4.

3) Wait for an item to appear either from the terminal or from the network.

   If an item appears from the terminal, then go to 4.

   If a data item appears from the network, then print it and go to 3.

If a command appears from the network, then an error has occured.

4) Read an item from the terminal input buffer.

If the item is not a break, then print/skip it and go to 2.

If the item is a break, then print/skip it and go to 1.

NOTE: Output from the server host may occur at any time, such "spontaneous output" is printed in step 3.

Explanation

Both Hosts agree to use the RCTE option. After that, the using host (IAC DO RCTE) merely acts upon the controlling (serving) host's commands and does not issue any RCTE commands unless and until it (using host) decides to stop allowing use of the option (by sending IAC DON'T RCTE).

1) The using host is synchronized with the server by initially and when ever it returns to step 1 suspending terminal echo printing until it receives a command from the server.

The server may send either output to the terminal printer or a command, and usually sends a both.

The server may send output to the terminal printer either in response to user input or spontaneously. In the former case, the output is processed in step 1. In the latter case, the output is processed in step 3.

Server sends an RCTE command. The command may redefine break and transmission classes, action to be performed on break characters, and action to be performed on text. Each of these independent functions is controlled by separate bits in the <cmd> byte.

A transmission character is one which RECOMMENDS that the using host transmit all text accumulated up to and including its occurrence. (For network efficiency, using hosts are DISCOURAGED (but not prohibited) from sending before the occurrence of a transmission character, as defined at the moment the character is typed).

If the transmission classes bit (bit 4) is on, the two bytes following the two break classes bytes (or immediately following the <cmd> byte, if the break classes bit is not on) will indicate what classes are to be enabled.

If the bit is OFF, the transmission classes remain unchanged. When the RCTE option is first initiated, NO CLASSES are in effect. That is, no character will be considered a transmission character. (As if both TC1 and TC2 are zero.)

A break character REQUIRES that the using host transmit all text accumulated up to and including its occurrence and also causes the using host to stop its print/discard action upon the user's input text, until directed to do otherwise by another IAC SB RCTE <cmd> IAC SE command from the serving host. Break characters therefore define printing units. "Break character" as used in this document does NOT mean Telnet Break character.

If the break classes bit (bit 3) is on, the two bytes following <cmd> will indicate what classes are to be enabled. There are currently nine (9) classes defined, with room for expansion.

If the bit is OFF, the break classes remain unchanged. When the RCTE option is initiated, NO CLASSES are to be in effect. That is, no transmission will take place in the user to server direction until the first break reset command is received by the user from the server.

The list of character classes, used to define break and transmission classes are listed at the end of this document, in the Tables Section.

Because break characters are special, the print/discard action that should be performed upon them is not always the same as should be performed upon the rest of the input text.

For example, while typing a filename to TENEX, I want the text of the filename to be printed (echoed); but I do not want the <escape> (if I use the name completion feature) to be printed.

If bit 1 is ON the break character is NOT to be printed.

A separate bit (bit 2) signals whether or not the text itself should be printed (echoed) to the terminal. If bit 2 = 0, then the text IS to be printed.

Yet another bit (bit 0 - right-most bit) signals whether or not any of the other bits of the command should be checked. If this bit is OFF, then the command should be interpreted to mean "continue whatever echoing strategy you have been following, using the same break and transmission classes."

2) The user Telnet now checks the terminal input buffer, if it contains data it is processed in step 4, otherwise the user Telnet waits in step 3 for further developments.

3) The user Telnet waits until either the human user enters some data in which case Telnet proceeeds to step 4, or an item is received from the network. If the item from the network is data it is spontaneous output and is printed, Telnet then continues to wait. If the item from the network is a command then an error has occured. In this case the user Telnet may attempt to resynchronize the use of RCTE as indicated below.

4) Items from the terminal are processed with printing controlled by the settings of the latest break reset command. When a break character is processed, the cycle of control is complete and action re-commences at step 1.

Input from the terminal is (hopefully) buffered into units ending with a transmission or break character; and echoing of input text is suspended after the occurrence of a break character and until receipt of a break reset command from the serving host. The most recent break reset command determines the break actions.

In summary, what is required is that for every break character sent in the user to server direction there be a break reset command sent in the server to user direction. The user host initially has no knowledge of which characters are break characters, and so starts in a state that assumes that there are no break characters and also that no echoing is to be provided. The server host is expected to send a break reset command to establish the break classes and the echoing mode before it receives any data from the user.

Synchronization and Resynchronization

The serving and using hosts must carefully synchronize break reset commands with the transmission of break characters. Except at the beginning of an interaction, the serving host may only send a break reset command in response to the Using host's having sent a break character as defined at that time. This should establish a one-to-one correspondence between them. (A <cmd> value of zero, in this context, is interpreted as a break classes reset to the same class(es) as before.) The break reset command may be preceded by terminal output.

The re-synchronization of the break characters and the break reset commands is done via the exchange of the Telnet signal Abort Output (AO) in the server to user direction and the SYNCH in the user to server direction.

Suppose the server wants to resynchronize the break characters and the break reset commands.

a. The server should be sure all output to the terminal has been printed by using, for example, the Timing Mark Option.

b. The server sends the AO signal.

c. The user receives the AO signal. The user flushes all user to server data wheather it has been echoed or not. The user sends a SYNCH to the server. [The SYNCH consists of the Telnet Data Mark (DM) and the host-to-host interrupt (INS).] The user now enters the initial state at step 1.

d. The server receives the SYNCH and flushes any data preceeding the DM (as always). The server now sends a break reset command. (Actually the break reset command could be sent at any time following the AO.)

Suppose the user wants to resynchronize the break characters and the break reset commands.

a. The user should discard all user to server data whether it has been echoed or not.

b. The user sends the AO signal. The user now enters the algorithm at step 1.

c. The server receives the AO signal. The server discards all data buffered but not yet sent to the user. The server sends a SYNCH to the user. The server sends a break reset command to the user.

Notes and Comments

Even-numbered commands, greater than zero, are in error, since they will have the low-order bit off. The command should be interpreted as equal to zero, which means that any classes reset bytes ([TC1 TC2] [BC1 BC2]) will be in error. (The IAC SE, at the end of the command, eliminates any parsing problems due to this error.)

Serving hosts will generally instruct using hosts not to echo break characters, even though it might be alright to echo most break characters. For example, <cr> is usually a safe character to echo but <esc> is not. TENEX Exec is willing to accept either, during filename specification. Therefore, the using host must be instructed not to echo any break characters.

> This is generally a tolerable problem, since the serving host has to send an RCTE command at this point, anyhow. Adding an echo for the break character to the message will not cause any extra network traffic.

The RCTE Option entails a rather large overhead. In a true character-at-a-time situation, this overhead is not justified, but on the average, it should result in significant savings, both in network traffic and host wake-ups.

Buffering Problems and Transmission vs. Printing Constraints:

> There are NO mandatory transmission constraints. The using host is allowed to send a character a time, though this would be a waste of RCTE. The transmission classes commands are GUIDELINES, so deviating from them, as when the user's buffer gets full, is allowed.
>
> Additionally, the using host may send a break class character, without knowing that it is one (as with type-ahead).
>
> > If the user implementation is clever it may send the user entered data to the server before it is actually needed. This type ahead data may contain break characters.
> >
> > Assume that only space is a break character (that is the last break reset command specified print up to and including the break characters and set the break classes to class 9). Suppose the user had typed "abc<space>def<esc>ghi<cr>". The user side RCTE could send it all to the server, but it could print only "abc<space>", and would have to buffer

"def<esc>ghi<cr>" at least until a break reset command was received from the server. That break reset command could change the break classes, requiring rescanning of the buffered string.

For example, suppose the break reset command set the break characters to class 5 and the action to print up to, but not including, the break character. The user RCTE could then print "def" and discard the <esc>, but whould have to continue to buffer the "ghi<cr>".

The problem with buffering occurs when printing on the user's terminal must be suspended, after the user has typed a currently valid break character and until a break reset command is received from the serving host. During this time, the user may be typing merrily along. The text being typed may be SENT, but may not yet be PRINTED.

The more common problem of filling the transmission buffer, while awaiting a host to host allocate from the serving host, may also occur, but this problem is well known to implementors and in no way special to RCTE.

In any case, when the buffer does fill and further text typed by the user will be lost, the user should be notified (perhaps by ringing the terminal bell).

Text should be buffered by the using host until the user types a character which belongs to the transmission class in force at the moment the character is typed.

Transmission class reset commands may be sent by the serving host at any time. If they are frequently sent separate from break class reset commands, it will probably be better to exit from RCTE and enter regular character at a time transmission.

It is not immediately clear what the using host should do with currently buffered text, when a transmission classes reset command is received. The buffering is according to the previous transmission classes scheme.

The using host clearly should not simply wait until a transmission character (according to the new scheme) is typed.

Either the buffered text should be rescanned, under the new scheme; or the buffered text should simply be sent as a group. This is the simpler approach, and probably quite adequate.

It is possible to define NO BREAK CHARACTERS except Telnet commands (IAC ...). This seems undesirable and should not be done.

If this situation were to occur the using host should send a Telnet command to allow the server to know when he may reset the break classes, but the mechanism is awkward and this case should be avoided.

6. Sample Interaction

"S:" is sent from serving (WILL RCTE) host to using host.
"U:" is sent from using (DO RCTE) host to serving host.
"T:" is entered by the terminal user.
"P:" is printed on the terminal.

Text surrounded by square brackets ([]) is commentary.
Text surrounded by angle brackets (<>) is to be taken as a single unit, e.g., carriage return is <cr>, and the decimal value 27 is represented <27>.

The following interaction shows a logon to a Tenex, initiation of the DED editor, insertion of some text and the return to the Exec level.

An attempt has been made to give some flavor of the asynchrony of network I/O and the user's terminal input. Many other possible combinations, using the same set of actions listed below, could be devised. The actual order of events will depend upon network and hosts' load and the user's typing speed.

We assume that the user's Telnet is also in an "insert linefeed" mode. That is, whenever the user types carriage return <cr> the user Telnet sends both carriage return and linefeed <cr><lf> (the Telnet end of line signal). When space character occurs at the end of a line in the example description it is shown explicitly by <sp> to avoid confusion. Other uses of the space character are not so marked to avoid destroying the readability of the example.

A Telnet connection has already been opened, but the TENEX prompt has not yet been issued. The hosts first discuss using the RCTE option:

```
S: <IAC><WILL><RCTE>

U: <IAC><DO><RCTE>

S: TENEX 1.31.18, TENEX EXEC 1.50.2<cr><lf>@
   <IAC><SB><RCTE><11><1><24><IAC><SE>

   [Print the herald and echo input text up to a break character,
   but do not echo the break character. Classes 4 (Format
   Effectors), 5 (Non-format Effector Controls and <DEL>), and 9
   (<sp>) act as break characters.]

P: TENEX 1.31.18, TENEX EXEC 1.50.2<cr><lf>@

T: LOGIN ARPA<cr>

P: LOGIN

U: LOGIN<sp>

U: ARPA<cr><lf>

S: <sp><IAC><SB><RCTE><0><IAC>SE>

P: <sp>ARPA

S: <cr><lf>(PASSWORD): <IAC><SB><RCTE><7><IAC><SE>

P: <cr><lf>(PASSWORD):<sp>

T: WASHINGTON 1000<cr>

   [The password "WASHINGTON" is not echoed.  Printing of
   "1000<cr>" is withheld]

U: WASHINGTON<sp>

U: 1000<cr><lf>

S: <sp><IAC><SB><RCTE><3><IAC><SE>

S: <cr><lf>JOB 17 ON TTY41 7-JUN-73 14:13<cr><lf>@
   <IAC><SB><RCTE><0><IAC><SE>

P: <sp>1000
```

[Printing is slow at this point; so the account number is not printed as soon as the server's command for it is received.]

```
P: <cr><lf>JOB 17 ON TTY41 7-JUN-73 14:13<cr><lf>@

T: DED<esc><cr>

P: DED

U: DED<esc>

S: .SAV;1<IAC><SB><RCTE><0><IAC><SE>

P: .SAV;1

U: <cr><lf>

S: <cr><lf><lf>DED    3/14/73 DRO,KRK<cr><lf>:
   <IAC><SB><RCTE><15><1><IAC><255><IAC><SE>
```

[The program is started and the DED prompt ":" is sent. At the command level, DED responds to every character. The server sets the break classes to all classes.]

```
P: <cr><lf><lf>DED    3/14/73 DRO,KRK<cr><lf>:

T: IThis is a test line.<cr>This is another test line.<^Z>Q
```

["I" means Insert Text. The text follows, terminated by a Control-Z. The "Q" instructs DED to Quit.]

```
U: I

U: This is a test line.<cr><lf>

S: I<cr><lf>*<IAC><SB><RCTE><11><0><24><IAC><SE>
```

[DED prompts the user, during text input, with an asterisk at the beginning of every line. The server sets the break classes to classes 4 and 5, the format effectors and the non-format effector controls.]

```
P: I<cr><lf>*This is a test line.

S: <cr><lf>*<IAC><SB><RCTE><0><IAC><SE>

P: <cr><lf>*This is another test line.
```

U: This is another test line.<↑Z>

U: Q

[Note that the "Q" will not immediately be printed on the terminal, since it must wait for authorization.]

S: ↑Z<cr><lf>:<IAC><SB><RCTE><15><1><IAC><255><IAC><SE>

[The returned "↑Z" is two characters, not the ASCII Control-Z or <sub>.]

S: Q<cr><lf>@<IAC><SB><RCTE><11><1><24><IAC><SE>

P: Q<cr><lf>@

And the user is returned to the Exec level.

TELNET Output Line Width Option

1. Command name and code.

NAOL 8 (Negotiate About Output Line-width)

2. Command meanings

In the following, we are discussing a simplex connection, one half of a full duplex TELNET connection. On the simplex connection under discussion, by definition data passes from the data sender to the data receiver. If we consider the example of a computer transmitting data over a connection to a terminal where the data is printed, then the computer is the data sender and the terminal is the data receiver. Continuing to use this example, the NAOL option could be used to negotiate the line width to be used when printing lines from the computer on the terminal. To negotiate line width on the other half of the TELNET connection the parties involved reverse their data sender and data receiver roles; this can be done unambiguously as the sender of a DO or DON'T NAOL command can only be the data sender, thus defining the half of the TELNET connection under discussion, and the sender of a WILL or WON'T NAOL command can only be the data receiver.

IAC DO NAOL

The data sender requests or agrees to negotiate about output line width with the data receiver. In the case where agreement has been reached and in the absence of further subnegotiations, the data receiver alone is assumed to be handling output line width considerations.

IAC DON'T NAOL

The data sender refused to negotiate about output line width with the data receiver, or demands a return to the unnegotiated default mode.

IAC WILL NAOL

The data receiver requests or agrees to negotiate about output line width with the data sender. In the case where agreement has been reached and in the absence of further subnegotiations, the data receiver alone is assumed to be handling output line width considerations.

Position paper for KWAC

KWAC would Like a position paper from ARC setting out the upcomming events. Specifically the new computer, software, operAting policies, pricing policies, implementing schedules etc.

we would like to assure you that this does not indicate a position of distrust or ill faith, but rather something of mutual concern that we may lean upon in the upcomming months.

The reasons for this is mainly that our managers demand numbers and information to reinforce our "Beliefs" and our recommended direction that "we" should take during a period of transition.

Several users are contractors who could claim overruns in time and/or dollars if they are denied services.

Some of the specific concerns and topics we would like you to address in this position paper are listed here.

High level management users are being introduced during this period.

The satisfaction of these managers will have serious impact upon future growth.

Their satisfaction will very likely create acceleration in our activities.

Their dissatisfaction would result in no more than a continuation of the current level of activity at best.

Additional items we would like you to address are as follows:

Contention problems of system design

What happens if you do not get customer base you expect

Explain your depreciation method and how it differs from or is better than Tymeshare Inc.

Who picks up tab if 50% of the new machine is not sold.

We are now facing possible overruns on the current year subscription in view of the 26 unsold CRU's.

What happens if the mew facility is over subscribed.

Would you think in terms of expanded facility or commercial vendors. What would be your criteria for such a decision.

What are your plans regarding development of operating systems (NLS) at commercial vendor sites.

We would feel more comfortble when this happens, because it assures a back-up for additional CRU demand which may not be satisfied at Office-1.

An expression of your projected pricing policies.

We need as much information as we can have to use in our procurement projections. (As you know we have a minimum of 90 to 120 days procurement lead time.)

We believe special consideration should be given to the Bell Canada account.

Your policy on connect time CRU sales.

Describe the new scheduler and your implementation plan.

Your prediction of when the Architects' will be permitted use of the new system for testing.

Anticipated problems with the new operating system, such as what is the equivelent of <CTL-A>, and how are files named, etc.

What is the schedule for documentation to define differences in operating systems.

Your plans and schedule for implementing NLS 8.5 to NLS 9.

The transition from NLS 7 to NLS 8 was made less trumatic because of careful thought on the part of ARC and close coordination with the Architects.

Your time frames for transition to NLS 9.
    We would prefer that this not happen at the same time as the hardware transition.
What is the anticipated impact of the existing ARC Development load on the new facility.

We would like assurance that any future planning be closely coordinated with the Knowledge Workshop Architect Community.

We would appreciate this position paper to the KWAC via journal mail.

< HJOURNAL, 39410.NLS;1, >, 18-MAR-77 17:33 XXX ;;;; Title:
Author(s): Raymond R. Panko/RA3Y; Distribution: /SRI([INFO-ONLY])
SRI-ARC([INFO-ONLY]) BELL-CANADA([INFO-ONLY]) KWAC([INFO-ONLY
]) ; Sub-Collections: SRI-ARC BELL-CANADA KWAC; Clerk: RA3Y;

Draft of a Brief History of ARPANET Computer Mail

I would appreciate any suggestions for changing this paper. By the way, this is a LONG brief survey.

B. COMPUTER-BASED MEDIA 1

Compared to other forms of electronic mail, there is a strong kinship between ARPANET computer mail, computer teleconferencing, and commercial mailbox services on time-sharing networks. What the three forms have in common is a single ancestor; all can trace their heritage to the "mailbox" services that have been available on even the earliest time-sharing systems. Mailbox programs allow one user to send brief messages to other users or to operators. These messages either go directly to the terminal of the recipient or are delivered the next time the receiver logs into the system. Yet actual communication between designers of different media, although growing, is still very small. ARPANET designers have been mostly artisans, who have seldom reported on their systems in the open literature. Conferencing designers have tended to be social scientists, who have prepared voluminous reports, usually dealing with impacts on people. Commerical mailbox designers have business people, who have seldom even communicated with one another. 2

Before time-sharing, users had to hand-carry programs to the computer center. While this cumbersome process had numerous drawbacks, it did bring progammers into contact with their colleagues and with computer operators. If they had questions, they could walk a few feet and ask them. But time-sharing systems separated users from one another by hundreds of yards, so getting help became difficult. Mailbox programs were installed to ease communication problems. The very first time-shared computer, CTSS at MIT's Project MAC, offered a mailbox program (*), and few time-shared systems since then have not followed suit. 3

* Crisman, P.A., ed., The Compatible Time-Sharing System, A Programmer's Guide (2nd Ed.), The MIT Press, Cambridge, Massachusetts, 1965, Section AH.9.05, quoted in Stuart L. Mathison and Philip M. Walker, Computers and Telecommunications: Issues in Public Policy, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1970. 3a

A related facility is "linking," in which two terminals are tied together so that each user can see what the other is typing. If mailbox delivery resembles postal service or interoffice mail, linking resembles conversational interactions, such as telephone calls or face-to-face meetings. While linking and mailbox services can be quite distinct from one another on any given system, they really form a conceptual continuum of asynchronous and synchronous service. 4

1. ARPANET COMPUTER MAIL 5

During the 1960's, a substantial amount of the world's advanced computer research was funded by the Advanced Research Projects Agency (ARPA) of the U. S. Defense Department. To make ARPA-funded software more generally available, ARPA began funding the development of a national packet-switched computer network, later known as the ARPANET. The first lines and switching computers were installed in 1969, but it was not until the network's file transfer protocols (FTPs) were refined, in 1972, that the network became fully operational. In 1975, operation of the network was transferred to the Defense Communications Agency. Today, there are 182 host computers on the network. Half of these are large computers, the other half minicomputers. 6

The most common large computer on the network is Digital Equipment Corporation's PDP-10. Most of the 36 PDP-10s on the network use the TENEX operating system, developed by Bolt, Beranek and Newman, under ARPA funding, in the early 1970's. In 1972, just before the FTP was established, Ray Tomlinson of BBN developed message sending and reading programs for TENEX. The sending program was called SNDMSG, the reading program READMAIL. Originally, SNDMSG and READMAIL were written to handle mail flows within individual PDP-10s. Late in 1972, the package was rewritten to handle message distribution over the network, via the FTP. 7

The subsequent development of computer mail is difficult to characterize chronologically. There were several streams of development, in which successive programs refined earlier efforts. But each stream borrowed extensively from the others, adopting attractive innovations developed in other ARPANET systems. 8

The oldest stream grew out of refinements to READMAIL. Although SNDMSG was gradually refined, primarily by Julie Sussman, its evolution was gradual and limited. Just the opposite was true for the message reading side of the package. In 1973, Larry Roberts at ARPA wrote RD. This program was coded in TECO (a text editing program) macros. It offerred several new features, as did its successor NRD, which was written later in 1973, in the SAIL language, by Barry Wessler at Telenet. In 1974, Martin Yonke and John Vittal at U.S.C.'s Information Sciences Institute (ISI) wrote WRD in 1974. Later that year, Yonke wrote BANANARD, and, in 1975, Vittal wrote MSG. MSG is currently the most popular message-reading program on the ARPANET. 9

These programs, from READMAIL through MSG, were written for TENEX PDP-10s. Also written for TENEX machines were two other programs, developed more or less simultaneously in the middle of the decade at Bolt, Beranek and Newman. HG (the checmial symbol for mercury) was written by James Calvin in 1974. Another program, MAILSYS, was developed under Ted Myer in 1974. MAILSYS, which was also called XMAIL, was the larger project. As discussed below, it was later expanded to become HERMES (TM), for the Military Message-Handling Experiment. MAILSYS, and to some extent HG, combined both mail reading and mail composition functions. In contrast, the READMAIL-MSG stream either had no message composition tools or transferred the user to SNDMSG for composition. 10

In 1975, the Dynamic Modeling System project at MIT developed a message program called MSGDMS. This program was developed primarily by Mike Brooz, working under Al Vezza. MSGDMS's major innovation was its inverted index system, which allowed for very rapid retrival of old messages. Another major innovation was the sophisticated use of "background" processes, which periodically run housekeeping chores during unused computer cycles. In addition, MSGDMS was linked to the ARPANET Data Computer, a terrabit storage center. This allowed old messages to be archived, thus reducing online storage costs (which can be heavy for frequent computer mail users) yet still allowed reasonably fast retrieval of archived messages. MSGDMS was originally written under the ITS operating system developed at MIT. It was later modified to run under TENEX, under the name XMAIL. 11

The fourth major stream of development began in 1973, when ISI released its report "Consolidation of Telecommunications on Oahu" (COTCO). The report, based on an extensive study of naval communications on Oahu, recommended the the application of computer mail to operational military environments. ARPA then funded the Information Automation project at ISI, under Rob Stotz, to develop a terminal and computer mail software for a military environment. In 1975, ARPA expanded the effort and funded the development of competitive computer mail programs at MIT and BBN. The ISI program is called SIGMA. The BBN program, HERMES, was based on MAILSYS. The MIT program is as yet unnamed. All three programs are written to run on the Hewlett-Packard 2645A terminal, under operating system software developed by ISI. 12

This year, one of the three programs will be tested extensively in the Military Message-Handling Experiment in Oahu. The other two programs will be tested, but on a more limited basis. MITRE is currently evaluating the three systems. During the experiment on Oahu, MITRE will conduct an evaluation of the results. 13

Our discussion has passed over many computer mail programs whose application has been more limited but which introduced many interesting and useful features. One current program that bears special mention is MS, being developed by Dave Crocker and Bill Crosby, under Bob Anderson, in RAND's Personal Computing project. MS (pronounced "Miz") runs on a RAND-UNIX PDP-11 minicomputer. The UNIX operating system, which was developed at AT&T, has sophisticated text-editing features. 14

ARPANET computer mail is almost bewildering for its diversity. Some programs were developed under intense direct funding. Others were written in programmers spare time. Despite this diversity, ARPA has been able to coordinate network mail development, albeit loosely. Minimal mail header standards for FTP have been created under ARPANET Requests for Comment 680 and 720. In addition, the ARPANET has a working committee on Computers and Human Communication (CAHCOM), which is chaired by Dave Farber at Irvine. There is also a loosely-knit Message Service Group (MSGGROUP), which has about 60 members and has been conducting an active general dialog over the network since mid-1975. Although much standardization still needs to be done, messages can already be sent among most TENEX and non-TENEX hosts on the ARPANET. 15

Perhaps the most useful service provided by the network to mail users is the ARPANET Directory, which is now funded by the Defense Communications Agency. The Directory, published by the Network Information Center (NIC) at SRI, is like a telephone book -- giving the names of ARPANET mail users, the host computers to which their mail is delivered, their postal addresses, their telephone numbers, their unique network idents (which are like telephone numbers, license plates, or TWX acronyms), and the names of network groups to which they belong. 16

A discussion of network communication would be incomplete without a discussion of "linking," in which two distant terminals are locked together. When two users link terminals, each can see what the other types. They can type messages back and forth in a completely conversational style. They can even execute a program together and discuss the results. Most large ARPANET hosts have some form of linking for two parties using the same host. In addition, through the RSEXEC system developed at BBN, a user on any TENEX computer can locate and link to any other loged-in of a TENEX machine on the ARPANET, and with users and several other types of machines as well. RSEXEC is essentially a distributed multi-host operating system with many interesting characteristics. RSEXEC linking across hosts has been available since 1972. A number of non-TENEX hosts now offer RSEXEC. 17

In NLS, a system developed under Engelbart at Stanford Research Institute, shared-screen teleconferencing allows two users to link video displays or to link a video display to a wall screen projector. Shared-screen teleconferencing in conjunction with a telephone call, has been used to train distant NLS users. Shared screen teleconferencing has also been used to augment face-to-face meetings. 18

Normally only two parties are linked at one time. Multi-party linking would require considerably more discipline than standard linking tools provide. Some progress has been made in providing multi-user, multi-host linking. The most notable example is TALK, developed by Jim Calvin at BBN. TALK allows group conferencing among users on several different hosts. 19

The ARPANET has been used on several occasions to handle true computer teleconferencing systems. Until late 1974, for example, the FORUM teleconferencing system, developed at the Institute for the Future under Vallee, Amara, Lipinski, Miller, and Helmer, used the ARPANET for its experiments. But FORUM has not played any long-term role on the NETWORK, except for some limited continuing uses at ISI. 20

Composition requires editing, and virtually all message-sending systems provide some editing tools. Editing can become quite extensive, including right-margin justification, automated spelling-correction, and customized formatting. In advanced systems, the dividing line between computer mail and word processing is becoming quite blurred. 21

We have held until the last our discussion of the most conceptually sophisticated computer mail system on the ARPANET: NLS Journal Mail. NLS, as noted above, was developed at SRI. Development began in 1963, primarily under ARPA funding. It continues today, under mixed funding. Overall, NLS is an integrated office automation system, offering extensive document composition tools, forms systems, and other office-related tools. In 1970 and 1971, SRI developed the Journal Mail subsystem, to distribute messsages, pre-prepared documents, data, line-drawn pictures, and other information. Because NLS was developed in an environment where long documents were common, it developed facilities for delivering long documents without inconveniencing users -- a facility no other ARPANET computer mail system provides. 22

Journal Mail was built as a working tool for a complex programming effort. As a result, careful thought was given, in 1969, to the problems raised by personnel turnover, limits on human recall, general communication processes on complex projects and other facts of life that require formal dialog in organizations. The Journal design embodied many correspondence control mechanisms, although most of these were not implemented fully. 23

The original design called for directory assistance-type functions and dialog recording in a multi-host environment, and many design innovations were made on this area. The original design also called for logical entities called "sets" of messages, which could be treated as personal message files, as teleconference transcript files, as successive versions of a controlled document, and so on.

Potentially, sets could provide a basic logical architecture for complex communication processes. 24

Many concepts in the original journal design were not implemented, but others were. Features that were developed, such as an ability to handle long documents and on-line "directory assistance" functions make journal mail quite powerful. In addition, NLS Journal Mail has "hooks" into other media. It can accept files prepared off-line on several terminals; it can deliver mail to a non-NLS user via standard ARPANET computer mail; and, if a user is not a network user, it prints the item for postal distribution. 25

A number of ARPANET mail programs are beginning to be used by nonresearchers. NLS Journal Mail, HERMES, and MSG are being used by operational military organizations with access to the ARPANET. The U.S. Army Material Development and Readiness Command (DARCOM) for example, has over 200 computer mail users. 26

Both NLS Journal Mail and HERMES have non-ARPANET users. HERMES service is available via Telenet, a commercial computer netwwork owned in part by BBN. NLS service, including Journal Mail, is sold f.o.b. Cupertino. 27

While ARPANET computer mail is the most visible communication-oriented activity on the network, there are scattered ancillary developments that indicate new ways to augment the usefulness of computer mail. We consider just three here: calendar, bulletin board, and forms systems. 28

As discussed in Working Paper II, office workers communicate most frequently with people who are "close" in the organizational sense, i.e., the average individual has far more communication with his or her office mates than with people in other divisions in thecompany, still less with people in other companies. It may follow also be true that a person's most intense form of communication is with himself or herself in the form of notes and reminders. At least one good appointment system has been built on the ARPANET. This is CALENDAR, written at BBN by Ted Strollo. CALENDAR is not a sophisticated system, nor is it stylistically easy to use, but it is quite workable and points to new directions for development. We have no statistics on its daily use by a typical user, but we suspect it rivals or surpasses the frequency of mail system use. 29

Moving up to the level of the working group, Richard Kahler at Stanford Hospital's SUMEX-AIM project has written a bulletin board program, called BBD, that runs under TENEX. BBD is written in SAIL, and it is a well-designed, well-conceptualized system. It is very similar in its command style to ARPANET computer mail systems, especially MSG; in fact, it is tied to MSG, although rather loosely.

Strictly speaking, BBD should not be listed under ARPANET computer mail, since it is not an ARPA-funded activity, but it is closely tied to the ARPANET community. 30

Conceptually, it would be possible to build calendar and bulletin board systems as integral parts of computer mail systems. Conceptually, both appointments and bulletin board entries can be viewed as messages, albeit with some special characteristics such as expiration dates and notification cycles. With careful design, it would be possible for users to read their bulletin boards with his or her normal mail reading commands, to send notices to their calendar via normal computer mail, to receive appointment reminders via computer mail, and to read through filed messages as easily as messages in the user's standard "in box" (this last feature is already common). A mail system well-tied to calendar and bulletin board systems would probably receive a much higher level of use than a stand-alone mail system. 31

There has only been sporadic development in forms systems, although there are a few examples of special-purpose systems designed to handle one or two specific types of forms. One interesting experimental system system is NEWFORMS, a subsystem of NLS. In addition, the three systtems designed for the Military Message Handling Experiment can handle form composition quite easily. The HERMES command "Compose," for example, can be extended to include the name of a form, e.g., "Compose Invoice." 32

Most existing forms systems, unfortunately, do little more than prompt the composer to input various fields of information. But a forms system should also be able to collect a good deal of background information itself, so that it can fill in various fields automatically, without requiring human assistance. The NLS forms system has some capabilities in this area, but they are limited. Future forms systems will amost certainly search data bases and do complex data manipulations in order to fill out fields. 33

Recently, Michael D. Zisman has suggested that the challenge in forms processing is not task assistance but task recognition (*). According to Zisman, a system should not only assist the user in filling in forms, but it should automate the forms process by sensing when a form should be sent out for completion. As part of his dissertation at the Wharton School of the University of Pennsylvania, Zisman is developing a task-recognition system to automate the process of reviewing and publishing journal articles. The main activity of Zisman's system is to send out "tickler" messages when certain events occur, then checking on the status of responses to these messages as time goes on. 34

(*) Presentation at Stanford Research Institute, March 6, 1977. 34a

KWAC – Position paper

BR JCN DCL 25-MAR-77 17:41 39535

Position Paper

Overview 1

ARC intends to move Office-1 service from the KA-10 at Tymshare in Cupertino to a new KL-10 at SRI. The purposes for the move are to reduce costs, improve computer responsiveness, and open the way to more-flexible pricing structures in the future. This move, which should result in substantial benefits for all Utility subscribers, was made possible only as a result of a complex series of negotiations and decisions by ARC, SRI top management, DEC, Tymshare and government auditors. The feasibility and schedules have just last week become definite enough, and the benefits clear enough, that we can now begin to make plans for the move. Therefore KWAC's request for a position paper on this issue is now both appropriate and timely. 1a

Although we realistically must expect any change of computer to be somewhat disruptive, the proposed transition will be carefully planned to minimize any negative impact upon Office-1 users. We are especially optimistic because, during the past two weeks, we have seen a KA user community of size similar to that of Office-1 move onto this KL extremely smoothly. 1b

Major Factors 2

In summary, the proposed move looks like this: 2a

COMPUTER 2b

KL-1090T Processor (approx. 4 times faster than a KA-10) 2b1

One million words of memory (double Office-1, and much more-highly interleaved) 2b2

Six 200 mega-byte disk drives (six times Office-1 file volume capacity, and much expanded swapping capacity with comparable page-transfer rate). 2b3

Three independent disk channels to memory, interchangeably useable for swapping and files (Office-1 has one swapping and one file channel). 2b4

This computer, which will be entirely supplied and maintained by DEC, will be shared with the research community that formerly used the SRI-AI KA-10. The Utility is prepared to purchase up to 50% of the KL system, if necessary, to ensure responsiveness that exceeds recent Office-1 experience. (The cost of the entire KL facility, at allowed Government rates, is about the same as the present cost of Office-1-- approximately

$50000 per month. Therefore the Utility stands to save at least half its computer costs by the move.) 2b5

SOFTWARE 2c

The identical NLS systems that can run under TENEX on Office-1 also can run under the TOPS-20 monitor on the KL10. In fact NLS 8.5, both Typewriter and Display versions, is already available on the KL. Command language differences between TENEX and TOPS-20 are minor but noticable. We shall prepare a summary for distribution to KWAC well before the move. 2c1

The principal weakness of TOPS-20 is its time-sharing scheduler, which wastes a substantial part of the system's resources (especially when the computer is heavily loaded). This is why the Utility is prepared to buy half the system, even though only about one-third of the power of this configuration should be plenty to dominate Office-1 performance. Then, as the SRI system-tuning team (which includes strong ARC repesentation) increases the scheduler's efficiency, the capacity and throughput of the system will grow ahead of our anticipated user needs. 2c2

OPERATING POLICIES 2d

Operating policies on the KL are substantially the same as those we have been used to at Tymshare. We do expect the PM down-time of the KL to be less than that at Office-1, the life of files on the disks to be longer (before automatic archiving), the disk allocation per directory to be increased-- perhaps doubled-- (at no extra cost), and the operators to be at least as highly-skilled and responsive as Tymshare's -- and ARC will have more-direct influence in their training and supervision. 2d1

PRICING POLICIES 2e

Utility service prices will remain as currently agreed, until October 1, 1977. The move will thus have no immediate affect upon costs, but should result in noticeably improved performance. After October 1 a significant price decrease may be possible, along with novel subscription options. We will be designing the future pricing structure during the next few weeks, and will inform KWAC as soon as any decisions are made. Of course, your inputs concerning desired pricing alternatives would be most welcome and valuable right now. 2e1

IMPLEMENTING SCHEDULES 2f

The KL-10 is now operational with one-half its memory (512K words) and one-half its disks (three drives) on one disk channel. The AI users are all on the system now and delighted with its performance, in spite of the poor scheduler. Scheduler work is proceding well, with significant improvements expected during the next two months. 2f1

The balance of the memory is scheduled for delivery about mid May. At that time KWAC will be invited to try the system, in a small experimental pie-slice. At least one more (the fourth) disk drive, and second disk channel, are due shortly thereafter. Assuming no problems, we would propose to cut over when all that hardware is solidly established in the system -- probably by June 15. Of course, any major difficulties would simply cause us to keep Tymshare on the air longer. 2f2

The final disk channel and two drives will come in July, but are not crucial for our use of the system and can be plugged in with minimum disruption. 2f3

Specific Details 3

The remainder of this paper addresses specific points raised in <39392,> in the order in which they occur: 3a

Contention problems of system design: 3b

Disk/Drum contention-- Since the system has no drums, swapping will be done on the half-way-out tracks of all disk drives, so that disk arms will never have to move more than half way in order to swap. Wide TENEX experience indicates that there are about ten swapping-page transfers for each file-page transfer, so the arms will spend almost all their time properly positioned for swapping. We estimate average page swap time, including seek and latency, to be about 20ms (compared to 18ms at Office-1)-- but the KL will be able to swap simultaneously on three channels. 3b1

NLS/LISP contention-- The major concern about NLS and LISP sharing a computer is due to the fact that they both move around rapidly in large address spaces, and therefore may be expected to compete for similar aspects of the system's resources. Interference is minimized by providing enough core memory; hence we have configured the system with the current maximum of 1M words. If we merely assigned half the memory to each system, we would essentially be preventing interference and yet clearly be better off than we were on isolated KA systems. Actually we hope to do much better than that, by developing a sophisticated core-management scheduler (to be discussed below). 3b2

What happens if we do not get customer base we expect: 3c

The KL is an SRI responsibility, assigned to the IS&E Division. Additional core and disks have been ordered at ARC'S request, so that the system will have the capacity needed for the WUS. The basic system was justified on the basis of the AI research program, which has been in existence, with solid support and growing computer needs, for longer than the WUS. In addition, the system appears to be competitively priced and therefore can sell excess capacity on the open time-sharing market. 3c1

The system is priced initially to break even when about 80% of its capacity is sold; and its total initial user community, including WUS, is now funded at a rate that can pay this total bill (and is used to receiving far less compute power for such amounts of money). If facility sold time were consistently to exceed 80%, rates would have to be lowered (since the facility is a zero-budget cost center) -- or additional equipment would be acquired to expand capacity and improve responsiveness. If usage drops much below 80% and no new customers could be found to take up the slack, rates would have to be raised. This eventuality is extremely unlikely, since the facility can use outside customers as a buffer to stabalize income when individual project needs vary. 3c2

These problems are the responsibility of Dan Lynch and IS&E. From ARC's point of view, the KL is a utility with relatively stable per-cycle rates, that we can draw upon precisely to the extent that our customers desire. 3c3

Depreciation method: 3d

Tymshare depreciated the Office-1 machine in 40 months. The KL will be depreciated to 10% of its initial cost over 60 months This slower rate, combined with the fact that we have a new generation of hardware and no facility overhead costs (SRI will absorb these), means that the total monthly cost of the full KL is approximately the same as what the WUS has been paying for the Office-1 KA at Tymshare -- and we only need to buy less than half of the KL. 3d1

What happens if the new facility is oversubscribed: 3e

First, note that there is considerable slack in the capacity of the new facility -- the difference between 80% and 100% of the KL is more than 50% of a KA's compute power. (By "100% of the

KL" we still mean a system capable of delivering reasonable service, i.e. maintaining the load average generally below 6 and almost never above 10.) Beyond that, as the machine approaches saturation (i.e. higher load averages), we would have exactly the same decision problem as if we were on Tymshare -- i.e., add equipment or send customers to other machines, depending upon the needs and options of the customers. However, the "add equipment" option is easier with the large KL system as a base, since e.g. even adding a CPU would only increase system costs by about 20%. Within a year, TOPS-20 is expected to be able to support configurations with as many as four PDP-11 front ends (as terminal and network handlers), four million words of core (or solid state) memory, and dual CPU's-- so there is pleanty of room for growth. 3e1

What are plans regarding development of operating NLS systems at commercial vendor sites: 3f

NLS now runs under DEC's standard TOPS-20, and therefore technically can be run on any of the approximately 100 KL-10's and KL-20's now in operation, including several at commercial vendor sites. 3f1

The changes we are making to the scheduler are really just tuning fixes to improve efficiency, and we intend to make them available to DEC and interested users of DEC equipment. 3f2

Although we have no specific plan yet to install NLS at new commercial sites, we are continually open to conversations with commercial vendors on this subject (at least two have expressed definite interest). These options will be pursued further when the CRU load demands it. 3f3

Projected pricing policies: 3g

As previously mentioned, the change should have no immediate pricing impact (subject to audit approval of currently-proposed pricing policy for the period from Jan. to Oct.). Pricing policy for after October 1, including the possibility of connect-time charging, is under study and will be discussed with KWAC as soon as we have a definite proposal. We understand the need for adequate lead time, and for special consideration for Bell Canada. 3g1

New scheduler and implementation plan: 3h

The scheduler as distributed by DEC is a first come, first served round robin scheduler. It has no notion of priority. We are implementing a flavor of the TENEX Pie-slice-scheduler so that service to individuals may be meted out on a known measured basis. The initial phase of the Pie-slice-scheduler consists of implementing the BBN 134 version. That has actually already been done and is being tested in the evenings this week. The next phase of the scheduler development involves giving the ability to suballocate pie-slice resources to a person known as the Group-Administrator. This will take about 2 months to develop and will not be started until work on more urgent areas (such as core management discussed below) is completed. This new feature will probably not be provided until sometime after the move. 3h1

One of the serious flaws in the distributed version of TOPS20 has to do with what is called core garbage collection. Since we will have about 1800 pages of user core available we cannot use the simple scheme that DEC uses for cleaning up core. They simply wait until they are almost "out of core" and then they look all through core and hunt for pages to throw out. This usually causes about 500 pages to be selected and put on the outbound swapper queue. That is a disaster. Since each disk operation takes about 20 milliseconds to accomplish, this sudden burst of activity will take about 10 seconds of real time. Meanwhile, the poor users have to wait 10 seconds to get any other pages into or out of core while this giant flushing operation takes place. 3h2

We plan to change all that. The idea is to do reads before writes and to run the garbage collector very often and have it only collect a few pages of core each time (say 10 pages), thus keeping the queues to the swapper much shorter. The trick is going to be to figure out what pages to throw out without looking at all of them. We are currently implementing a scheme that involves keeping the "age" of every page in core on a global basis so that we can have a pretty good prediction on what the "age" of old pages is as we scan through the core tables looking for unused pages to discard. This method is in the final throes of design and will be implemented during the next few weeks. 3h2a

Additional scheduler features, such as assignment of core portions to users in proportion to their CPU pie slices, are planned for later implementation. Therefore we plan to have the most glaring deficiencies of TOPS-20 repaired within the next two months, and then continue to improve the system's performance on a step by step basis during the following year. 3h3

Differences between TENEX and TOPS-20, and their documentation: 3i

The principal problems expected for new users will be in two areas: 1) editing characters to the EXEC and 2) file name difference in the version field. That is, control-A for erasing a character will not work at the EXEC level (one must use the delete or rubout key). Programs, such as NLS, will be able to continue to use any editing characters they choose. 3i1

File names under TENEX were of the form NAME.TYPE;VERSION-NUMBER . Under TOPS20 they will be NAME.TYPE.GENERATION-NUMBER . The use of the " . " instead of the " ; " will cause programs that have file names stored away in them to not work under TOPS20. It is a trivial matter to change them, of course, but they must be changed. Also, file names are not allowed to use many of the special characters like [@(without quoting them with ^V to get them parsed correctly. A complete list of these differences will soon be made available to the users. 3i1a

The documentation of the differences in operating systems already exists at the JSYS level. There are online files that describe the known differences to the level of an assembly language programmer. There are verbose and terse forms of the documentation. 3i2

Plans and schedule for transition from NLS 8.5 to NLS 9: 3j

First, please note that this transition is totally independent of the computer move -- except that the KL will have more room for running different NLS systems side by side. 3j1

The current tentative schedule is to bring up NLS-9 on the KL for Development purposes within the next few weeks-- even before the move. Soon after the move-- say, late June?-- the rest of ARC will be encouraged to join the shake-down process. 3j2

As with the transition from NLS 7 to NLS 8, ARC staff will use the NLS 9 version that is to be made the running version at Office-1 (on the SRI KL facility) for at least two months before it is made available to KWAC for another month trial use, with at least early documentation about the differences they will encounter. 3j3

We will urge KWAC to try NLS 9 during their one month early access not only to get the feeling for its characteristics, but also to test any user subsystems they may have to determine which, if any, need modification to run under NLS 9. 3j4

Then, with notice to all WUS users and documentation, NLS 9 will be made the running NLS, with NLS 8 still available as

OLDNLS for at least a month. OLDNLS will then be removed from the system. 3j5

We anticipate late Summer (September?) to be the start of the KWAC-testing phase at this time. This is because we still have work to do on NLS 9, and because we do not want to bring up NLS 9 at the same time that we change to the KL facility. 3j6

Impact of ARC Development load on the new facility: 3k

The basic KL facility, with limited core and disks, was configured on the assumption that ARC Development would be sharing the system with the AI'ers (even before considering a WUS move for Office-1). The current and projected Development needs remain small compared to the capacity of the KL, and we anticipate no noticeable interference. On the other hand, direct KL computer access to Development without having to go through Elfs, Imps, and network hops, will make for much more efficient progress on Development projects. 3k1

Future Coordination 4

ARC management hereby assures KWAC that future planning will continue to be closely coordinated with KWAC. 4a

In particular, the future pricing policy and any updates or revisions to the above schedules or plans, will be journalized to KWAC as soon as they are available. 4a1

BANDAID:

Letter to the editor:

Dear JAKE,

I do hope BANDAID is the start of something good! It has been my experience that so much new is available that it is difficult to select the most interresting. Anyhow, I will mention a few as they come to mind.

1) The AID group agreed to send suggestions to David Potter (Ident DAP) for compilation and publication.

2) I was impressed with the idea that ARC may supply each architect with a portable video/sound presentation of NLS.

3) Bell Canada is considering a visit NSF to discuss the results of their survey. This could happen within the next two months!

4) The NSF Architect, Paul Custer, was elected coordinator of the Knowledge Workshop Architect Committee (KWAK). Booo...

5) Who is this person Rob? And his lovely wife who entertains so lavishly! A giant thank you note is in order!

6) And, while we are discussing nice people I think we should hear it for someone known as BR. Let's not forget the nice travel agent who follows him arround! What a nice addition to our community.

7) Seems certain now that you government types will loose your treasurer services. Thanks, Stony, if you hadn't been such a nice guy it would be easier to give you up!

8) Our new editor, deserves also an hero award. We knew she could do it and the proof is in your hand.

Well.... enough for issue one, but it's been fun. Paul

< DJOURNAL, 39602.NLS;1, >, 30-MAR-77 13:33 XXX ;;;; .HJOURNAL="JCN 30-MAR-77 13:24 39602"; Title: .H1="IMPACT Visit to RADC: Status and Dialogue"; Author(s): James C. Norton/JCN; Distribution: /MARKET([INFO-ONLY]) ; Sub-Collections: NIC MARKET; AccessList: MARKET JCN; Clerk: JCN; .IGD=0; .SNF=HJRM; .RM=HJRM-7; .PN=-1; .YBS=1; .PES;

I called John McNamara to see about the inpending visit by ESD IMPACT people to RADC later in April (the 25th?). John said that they are coming to see NLS and RADC work at Col. Krutz' invitation. RADC wants to suggest a combined effort and plans to mention the Gunter effort too. John will steer them to us here at ARC for further discussions and will let me know what happens at the visit and the likelyhood of their visit to ARC. Sounds all good so far. Here is a culled set of Journal messages JLM has been sending around RADC getting the visit set up:

IMPACT 19 Jan 77 38601 McNamara

Robert Kent-478-5432,33,34

Stoney-I called his office and he was at a meeting.I asked that he call you back since I will be gone starting tommorow again.Of interest I guess is they answer as the 427M office.

I have to visit ESD the week of th 7th of Feb so I could stop in then if it works out that way.Take Care

IMPACT 19 Jan 77 38605 McNamara

A interesting fact is that the pdp-10 at offfice 1 is avaialable for purchase .A possibility is that it could be purchased by us for the IMPACT program and we could have a sustem up and running for them with only a few months involved and let the project learn and evolve from there.It seems to me the point that the impact people shpuld be in tune to is that NLS is a viable protoype now.D Stone commented that in his brief discussion with B Kent,he indicated they were going to be unable to spend their 77 monies.

IMPACT 3 Feb 77 38802 McNamara

I did not make contact with the IMPACT types at ESD/MCI.It turned out the two main players,Bob Kent and Lt.Col Crocco were both on travel.I did talk with one of the guys who wrote the original plan and he was very interested in getting up and running with a system early.He has become a real believer that system,s like this evolve and are not spec,d out to the nth degree.He also told me that they were being funded substanitally in 78 so it looks like we should push are noses in as best we can.I suspect it is important if SRI types like Doug and Jim also make personal contact.

IMPACT 17 Mar 77 39369 McNamara

Here is the phone for B Kent.I do not seem to have the name of the Lt.Col who is involved with Kent but I am sure there is one.Of interest is they are people coming of the 427M prgram in fact the secretary answers as the 427 office.

Robert Kent-478-5432,33,34

IMPACT/Visit 25 Mar 77 39525 McNamara

The quest for a appropiate day continues.It turns out he ,Lt.Col Crocco,would prefer to set the week of the 25th as a target date and get back to us on the exact date when he can schedule the T-39 so they can bring all of th project members.I said it was okay with me but I would check with you.I said I would get back to him on Monday so need answer ASAP.

IMPACT/VISIT/SURVEY 25 Mar 77 39528 McNamara

we are tenatively talking about the week of April 25 that the IMPACT people would visit us. Are we going to be here?

To all AIDs:

The-AID is now operative and located under the SYSJOB allocation. Each member of AID has an initial file under his IDENT in this directory. That is, you may sign-on AID password DIA (you will be required to enter your IDENT when prompted) and you have an initial file to do AID related work and it is not charged to your CRU allocation!

All feedback to the AID coordinator should be addressed via sendmail to Paul (This is my ident). That will place a citation in my initials file which I will move into the AID file PAUL.

When you sign-on AID you may access any files except each members initial file.

I will be changing the AID password, you will be notified. All protections are now set at 770000. Which means only we (AIDs) may view these files. Two cautions: Please update any file in which you make a change, and please do not alter the protection.

Please respond to this inquiry so I will know you are on our distribution.

I plan to respond to our priorities pertaining to the following journal. I will respond (To EKM) April 5 with or without your responses so....

DLS 22-MAR-77 06:46 39454
Priorities for Spending SRI Labor
Location: (JOURNAL, JRNL44, J39454:gw)

In the event you did not recieve information regarding the "Position paper" for KWAC I am including citations for our request and the ARC response. I would like comments good or bad!

PAUL 17-MAR-77 19:46 39392
Position Paper for KWAC
Location: (DJOURNAL, 39392, 1:w)

BR JCN DCL 25-MAR-77 17:41 39535
Position Paper
Location: (DJOURNAL, 39535, 1:w)

AID Membership

PAUL 23-MAR-77 09:04 39478
AID Group/Membership
Location: (DJOURNAL, 39478, 1:w)

Trip Report

PAUL 23-MAR-77 13:10 39485
Trip Report
Location: (DJOURNAL, 39485, 1:w)

Did you respond to Potter DVN? We promised to do this at the KWAC meeting. He needs your input to the sales Aids

Here are mine if you would care to look:

PAUL 23-MAR-77 11:08 39481
Sales Aids; Lots of Luck
Location: (DJOURNAL, 39481, 1:w)

I discussed the possibility with Jim Norton of A free (included with each subscription) brief case size projector and screen with 16 mm film and sound track explaining the story of NLS! I believe we should push this idea, what do you think?

REFERENCES

1

1. P.A. Crisman, ed., The Compatible Time-Sharing System, A Programmer's Guide (2nd Ed.), The MIT Press, Cambridge, Massachusetts, 1965, Section AH.9.05, quoted in Stuart L. Mathison and Philip M. Walker, Computers and Telecommunications: Issues in Public Policy, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1970. 2

2. Virtually all information in this report concerning ARPANET computer mail was collected in computer mail interviews with system builders. 3

3. McLaughlin, Richard A., "A Survey of 1974 Dp Budgets," Datamation, Vol. 20, No. 2, February 1974. 4

4. McLaughlin, Richard A., "1975 DP Budgets," Datamation, Vol 21., No. 3, March 1975. 5

5. McLaughlin, Richard A., "1976 DP Budgets," Datamation, Vol. 22, No. 2, February 1976. 6

6. Vezza, Albert, "A Model for an Electronic Postal System," Paper presented at the 1975 Airlie House Conference, Washington, D.C. 7

7. Crockett, C.H. Jr, "Computer Communications Network Links Sonoco Branch Operations," Pulp and Paper, Vol. 48, No. 13, pp. 76-78, December 1974. 8

8. Nanus, Burt, and M.L. Wooton, "Implications of Multinational

Computers," Business Horizons, Vol. 17, No. 1, pp. 5-17, February 1974. 9

9. Frankl, Gerald M., "A Common Message Switch Ties most Stock Transactions to a Single High-Speed Format," Data Communications, pp 35-39, January 1977. 10
10. Industry Week, "Computer-to-Computer Ordering Looms on the Horizon," Vol. 180, No. 4, pp. 46-48, January 28, 1974. 11
11. Communications News, "Communications in 1984," Vol. 11, No. 9, p. 102, September 1974. 12
12. Personal communication with Mr. David McKnight, Bell Canada. 13
13. Wiley, Donald, "From Wire to Westar: Western Union's Changing Image," Communications News, pp 39ff., March 1977. 14
14. Personal communication with Mr. Francis Aler, Western Union Telegraph Company. 15
15. Hirsch, Phil, "Sita: Rating a Packet-Switched Network," Datamation, Vol. 20, No. 3, pp 60-63, March 1974. 16
16. Johnson, William R., "Redesign Clears Bottlenecks in Order-Processing Network," Data Communications, pp 29-32, September/October 1974. 17
17. Costigan, Daniel M., FAX: The Principles and Practice of Facsimile Communication, Chilton Book Company, Philadelphia, Pennsylvania. 18
18. Data from a study by The Yankee Group, quoted in Communications News, "Special Report: Facsimile's Finest Hour," p. 29ff., December 1976. 19

19. Data taken from a study by International Resource Development, quoted in Communications News, "Special Report: Future is Bright fro Facsimile Communications," p. 49ff., December 1974. 20

20. Ford, Ken, "The Development of Our Office Copying Process," The Office, Vol. 77, No. 6, June 1973. 21

21. Lavery, Daniel P., "Trends in Copier Usage," The Office, October 1974. 22

22. Business Week, "The Office of the Future," (special issue) June 30, 1975. 23

23. Business Week, "System 6: IBM Enters 'The Office of the Future,'" No. 2470, pp 46H-46J, February 14, 1977. 24

24. Rosenthal, Neal H., "Projected Changes in Occupations, Monthly Labor Review, Vol. 96, No. 2, December 1973. These are U.S. statistics, but they are reasonably close to Canadian statistics. 25

25. Stewart, Rosemary "Managers and their Jobs" Pan Books, London, 1967. 26

26. Burns, Tom "Management in action" Operational Research Quarterly, volume 8, number 2, 1957, pp. 45-60 27

27. Horne, James and Lupton, Tom. "The work activities of middle managers - an exploratory study". Journal of Management Studies, volume 1, number 2, 1965, pp 14-33. 28

28. Dubin, R., and S.L. Spray, "Executive Behavior and Interaction," Industrial Relations, No. 3, 1964. 29

29. Croston, J. D. and H. B. Goulding "The Effectiveness of

Communication at Meetings: a Case Study", Operational Research Quaterly, vol. 17, no. 1, pp 45-47 (1967). 30

30. Mintzberg, H., "Managerial Work: Analysis from Observation," Management Science, Vol. 18, No. 2, pp. B-97-B-110 (October 1971). 31

31. Palmer, A.W. and R.J. Beishon, "How the Day Goes," Personnel Management (1970). 32

32. Notting, reported in Sune Carlson, Executive Behaviour (Stromberg, Stockholm, 1951). 33

33. Case Institute of Technology, "An Operations Research Study of the Scientific Activity of Chemists," (1958). 34

34. Hinrichs, J. R., Communications Activity of Industrial Research Personnel, Personnel Psychology, Vol. 17, pp. 194-204, November 1963. 35

35. Klemmer and Snyder, cited in J.E. Karlin (ed.), "Human Factors Research - Some Recent Findings and Future Problems," Proceedings of the 5th International Symposium on Human Factors in Telecommunications, 1970. 36

36. Conrath, David W., and James H. Bair, The Computer as an Interpersonal Communication Device: A Study of Augmentation Technology and Its Apparent Impact on Organizational Communication, Proceedings of the Second International Conference on Computer Communication, Stockholm, Sweden, August 1974. 37

37. Lukasic, S.J., Organizational Adjustments to the Electronic Post

Office, paper read at the National Telecommunications Conference, San Diego California, 1974. 38

38. Pye, R., "Analysis of the Data Collected by the Survey of GECC," Report P/73280/PY, Communications Studies Group, Joint Unit for Planning Research (1973). 39

39. Modern Office Procedures, WP: Path to the '80's, Vol. 20, No. 6, June 1975. 40

40. Westrum, Ron, "Communications Systems and Social Change," Draft of Unpublished Doctoral Thesis, Department of Sociology, Purdue University, Lafayette, Indiana (May 1972). 41

41. Aiken, Eric, "Record Profits: Makers of Business Forms Score them Year by Year," Barrons, pp 11, 17-18, May 27, 1974. 42

42. Personal Communication with Mr. Ed Maston, International Business Forms, Incorporated. 43

43. Bull, Ivan, "The Difference Between Paperwork and Information," The Office, Vol 83, No. 1, pp 70-71, January 1976. 44

44. Haslett, John W., The New Trend Is People Paperwork, The Office, Vol. 77, No. 1, pp. 71-72, January 1973. 45

45. Ferriera, Joseph, and John Marus, Regulation: More Heat under Business, Computer Decisions, Vol. 8, No. 8, pp. 20-29, August 1976. 46

46. Amara, Roy, and Jacques Vallee, "FORUM: A Computer-Based System to Support Interaction Among People," Proceedings of the International Federation of Information Processing Congress, August 1974. 47

47. Personal communication with Jacques Vallee, then with the Institute for the Future, now with Infomedia. 48

48. Roetter, Susan G., "Assorted NLS Journal Statistics," Augmentation Research Center, Stanford Research Institute, 1974. 49

49. Personal communication with Nicole Leduc, Bell Canada. 50

50. de Smith, Michael. "Message Services - Two pilot surveys". Communications Studies Group (22 Gordon Street London WC1) P/76177/DS. 1976. 51

51. Wickesberg, A.K., "Communications Networks in the Business Organization Structure," Acadamy of Management, September 1968. 52

52. Driver, Michael J., and Siegfried Streufert, "Integrative Complexity: an Approach to Individuals and Groups as Information Processing Systems," Administrative Science Quarterly, Vol. 14, No. 2, pp 272-285, June 1969. 53

53. Daniel, D. Ronald, "Reorganizing for Results," Harvard Business Review," Vol. 44, No. 6, pp 96-104, December 1966. 54

54. Galbraith, Jay, Designing Complex Organizations, Addison-Wesley, Reading, Massachusetts, 1973. 55

55. Prahalad, C. K., "Strategic Choices in Diversified MNCs," Harvard Business Review, Vol. 54, No. 4, pp. 67-78, July-August 1976. 56

56. Sloan, Alfred P., My Years with General Motors, Doubleday, New York, 1964. 57

57. Withington, Frederick G., Beyond 1984: A Technology Forecast, Datamation, Vol. 21, No. 1, January 1975. 58

58. Personal communication with Mr. Bert Liffman, General Conferencing Systems, Ltd., Montreal, Canada. 59

59. Personal communication with Mr. John McCann, vice president, Scientific Time Sharing Corporation. 60

60. Personal communication with Dr. Howard B. Shirley, Batelle Memorial Institute. 61

61. Raw data were supplied by Mr. Ted Myer, Bolt, Beranek and Newman. 62

62. Brown, David, "Teleconferencing: The Summer of '76," University of Wisconsin, Madison Academic Computing Center, 1976. 63

63. Lecture by Dr. Marvin Sirbu, Massachusetts Institute of Technology, on the subject of electronic mail, March 24, 1977. Data were taken from an unreferenced study by The Yankee Group. 64

64. Personal communication with Mr. Dan Lynch, Stanford Research Institute. 65

65. Konkel, Gilbert J., and Phyllis J. Peck, Traditional Secretarial Costs Compared to Word Processing, The Office, Vol. 83, No. 2, pp. 66-67, February 1976. 66

66. Agnew, Carson E.; Paul Baran, and Ronald C. Crain, ARPANET Management Study: New Appplication Areas, Second Quarterly Technical Report, R-160, Appendix G, "The Costs of Electronic Messages," Cabledata Associates, Palo Alto, CA. 67