

DLS 22-SEP-75 14:41 33525

Source Code for FORMATTER Subsystem

User documentation is contained in (.33523)

Source Code for FORMATTER Subsystem

< STONE, FORMATTER,NLS;9, >, 5-SEP-75 13:56 DLS ;;;

1

Source Code for FORMATTER Subsystem

```

FILE cformat % USING (cml,sav,) TO (format,cml,) %           1a
% DECLARATIONS %                                           1a1
  DECLARE PARSEFUNCTION                                     1a1a
    answ,          % reads answer construct %             1a1a1
    answer,        % for questions - returns 0/1 %       1a1a2
    sp,           % reads next char, TRUE if space %     1a1a3
    readconfirm,  % reads next char if ca %              1a1a4
    readbug,      % reads next char if BUG %             1a1a5
    readoption,  % TRUE if next char is optchar %       1a1a6
    readrepeat,  % TRUE if next char is repeat %        1a1a7
    lockansw,    % TRUE if next char is Y/CA %          1a1a8
    lockconfirm, % TRUE if next char is                 1a1a9
    CA/REPEAT/INSERT %
    lockbug,     % TRUE if next char is BUG %           1a1a10
    locknum,     % TRUE if next char is a number %      1a1a11
    clearname,   % clears the name area %              1a1a12
    notca;      % reads next char, TRUE if not CA char% 1a1a13
  DECLARE COMMAND WORD                                     1a1b
    "BRANCH" = 1 ,                                       1a1b1
    "GROUP" = 2 ,                                       1a1b2
    "PLEX" = 3 ,                                        1a1b3
    "STATEMENT" = 4 ,                                   1a1b4
    "CHARACTER" = 5 ,                                  1a1b5
    "CONTROLCHAR" = 6 ,                                1a1b6
    "INVISIBLE" = 7 ,                                  1a1b7

```

Source Code for FORMATTER Subsystem

"LINK" = 8 ,	1a1b8
"DIRECTORY" = 9 ,	1a1b9
"PASSWORD" = 10 ,	1a1b10
"NUMBER" = 11 ,	1a1b11
"TEXT" = 12 ,	1a1b12
"VISIBLE" = 13 ,	1a1b13
"WORD" = 14 ,	1a1b14
"FILE" = 15 ,	1a1b15
"NEWFILELINK" = 16 ,	1a1b16
"OLDFILELINK" = 17 ,	1a1b17
"NAME" = 18 ,	1a1b18
"IDENT" = 19 ,	1a1b19
"IDENTLIST" = 20 ,	1a1b20
"EDGE" = 21 ,	1a1b21
"MARKER" = 22 ,	1a1b22
"NLS" = 23 ,	1a1b23
"ITEM" = 24 ,	1a1b24
"ITEMNOVS" = 25 ,	1a1b25
"SUCCESSOR" = 26 ,	1a1b26
"PREDECESSOR" = 27 ,	1a1b27
"UP" = 28 ,	1a1b28
"DOWN" = 29 ,	1a1b29
"HEAD" = 30 ,	1a1b30
"TAIL" = 31 ,	1a1b31
"END" = 32 ,	1a1b32

Source Code for FORMATTER Subsystem

"BACK" = 33 ,	1a1b33
"NEXT" = 34 ,	1a1b34
"ORIGIN" = 35 ,	1a1b35
"FILEReturn" = 36 ,	1a1b36
"RETURN" = 37 ,	1a1b37
"FILENAME" = 38 ,	1a1b38
"FIRSTNAME" = 39 ,	1a1b39
"NEXTNAME" = 40 ,	1a1b40
"EXTNAME" = 41 ,	1a1b41
"FIRSTCONTENT" = 42 ,	1a1b42
"NEXTCONTENT" = 43 ,	1a1b43
"FIRSTWORD" = 44 ,	1a1b44
"NEXTWORD" = 45 ,	1a1b45
"DETACHED" = 46 ,	1a1b46
"TTY" = 47 ,	1a1b47
"AUTO" = 48 ,	1a1b48
"CONTINUE" = 49 ,	1a1b49
"ON" = 50 ,	1a1b50
"RECOVER" = 51 ,	1a1b51
"SLINKER" = 52 ,	1a1b52
"UPDATE" = 53 ,	1a1b53
"CLEAR" = 54 ,	1a1b54
"IDENTS" = 55 ,	1a1b55
"FILES" = 56 ,	1a1b56
"DELETE" = 57 ,	1a1b57

Source Code for FORMATTER Subsystem

"DEFERRED" = 58 ,	1a1b58
"IMMEDIATE" = 59 ,	1a1b59
"NOT" = 60 ,	1a1b60
"PREVENT" = 61 ,	1a1b61
"RESET" = 62 ,	1a1b62
"ARCHIVE" = 63 ,	1a1b63
"SEQUENTIAL" = 64 ,	1a1b64
"TWO" = 65 ,	1a1b65
"JUSTIFIED" = 66 ,	1a1b66
"ASSEMBLER" = 67 ,	1a1b67
"BOTH" = 68 ,	1a1b68
"UNDELETE" = 69 ,	1a1b69
"FOR" = 70 ,	1a1b70
"STATUS" = 71 ,	1a1b71
"TAPE" = 72 ,	1a1b72
"ACCOUNT" = 73 ,	1a1b73
"NO" = 74 ,	1a1b74
"VERSIONS" = 75 ,	1a1b75
"EXTENSION" = 76 ,	1a1b76
"DATE" = 77 ,	1a1b77
"CREATION" = 78 ,	1a1b78
"LAST" = 79 ,	1a1b79
"FIRST" = 80 ,	1a1b80
"READ" = 81 ,	1a1b81
"WRITE" = 82 ,	1a1b82

Source Code for FORMATTER Subsystem

"DUMP" = 83 ,	1a1b83
"EVERYTHING" = 84 ,	1a1b84
"LENGTH" = 85 ,	1a1b85
"MISCELLANEOUS" = 86 ,	1a1b86
"ACCESSES" = 87 ,	1a1b87
"PROTECT" = 88 ,	1a1b88
"SIZE" = 89 ,	1a1b89
"TIME" = 90 ,	1a1b90
"VERBOSE" = 91 ,	1a1b91
"SORT" = 92 ,	1a1b92
"BYTESIZE" = 93 ,	1a1b93
"ARCHIVED" = 94 ,	1a1b94
"ALL" = 95 ,	1a1b95
"MODIFICATIONS" = 96 ,	1a1b96
"UPPER" = 97 ,	1a1b97
"LOWER" = 98 ,	1a1b98
"MODE" = 99 ,	1a1b99
"SENDMAIL" = 100 ,	1a1b100
"BUSY" = 101 ,	1a1b101
"QUICKPRINT" = 102 ,	1a1b102
"JOURNAL" = 103 ,	1a1b103
"PRINTER" = 104 ,	1a1b104
"COM" = 105 ,	1a1b105
"TERMINAL" = 106 ,	1a1b106
"REMOTE" = 107 ,	1a1b107

Source Code for FORMATTER Subsystem

"REST" = 108 ,	1a1b108
"CASE" = 109 ,	1a1b109
"CONTENT" = 110 ,	1a1b110
"TEMPORARY" = 111 ,	1a1b111
"VIEWSPECS" = 112 ,	1a1b112
"EXTERNAL" = 113 ,	1a1b113
"TC" = 114 ,	1a1b114
"PRIVATE" = 115 ,	1a1b115
"PUBLIC" = 116 ,	1a1b116
"TENEX" = 117 ,	1a1b117
"ALLOW" = 118 ,	1a1b118
"EXECUTE" = 119 ,	1a1b119
"APPEND" = 120 ,	1a1b120
"LIST" = 121 ,	1a1b121
"SET" = 122 ,	1a1b122
"SELF" = 123 ,	1a1b123
"FORBID" = 124 ,	1a1b124
"DISK" = 125 ,	1a1b125
"DEFAULT" = 126 ,	1a1b126
"OLD" = 127 ,	1a1b127
"NEW" = 128 ,	1a1b128
"COMPACT" = 129 ,	1a1b129
"RENAME" = 130 ,	1a1b130
"ADD" = 131 ,	1a1b131
"SUBTRACT" = 132 ,	1a1b132

Source Code for FORMATTER Subsystem

"MULTIPLY" = 133 ,	1a1b133
"DIVIDE" = 134 ,	1a1b134
"RIGHT" = 135 ,	1a1b135
"LEFT" = 136 ,	1a1b136
"ACTION" = 137 ,	1a1b137
"AUTHORS" = 138 ,	1a1b138
"COMMENT" = 139 ,	1a1b139
"EXPEDITE" = 140 ,	1a1b140
"HARDCOPY" = 141 ,	1a1b141
"INFORMATION" = 142 ,	1a1b142
"INSERT" = 143 ,	1a1b143
"KEYWORDS" = 144 ,	1a1b144
"OBSOLETES" = 145 ,	1a1b145
"RFC" = 146 ,	1a1b146
"SUBCOLLECTIONS" = 147 ,	1a1b147
"TITLE" = 148 ,	1a1b148
"UNRECORDED" = 149 ,	1a1b149
"L10" = 150 ,	1a1b150
"PROCEDURE" = 151 ,	1a1b151
"SEGGENERATOR" = 152 ,	1a1b152
"BUFFER" = 153 ,	1a1b153
"NDDT" = 154 ,	1a1b154
"PARSERULE" = 155 ,	1a1b155
"CA" = 156 ,	1a1b156
"CD" = 157 ,	1a1b157

Source Code for FORMATTER Subsystem

"RPT" = 158 ,	1a1b158
"BC" = 159 ,	1a1b159
"BW" = 160 ,	1a1b160
"BS" = 161 ,	1a1b161
"LITESC" = 162 ,	1a1b162
"IGNORE" = 163 ,	1a1b163
"SC" = 164 ,	1a1b164
"SW" = 165 ,	1a1b165
"TAB" = 166 ,	1a1b166
"IMLAC" = 167 ,	1a1b167
"TI" = 168 ,	1a1b168
"NVT" = 169 ,	1a1b169
"EXECUPT" = 170 ,	1a1b170
"MENU" = 171 ,	1a1b171
"DNLS" = 172 ,	1a1b172
"TNLS" = 173 ,	1a1b173
"COMMAND" = 174 ,	1a1b174
"RULE" = 175 ,	1a1b175
"SUBSYSTEM" = 176 ,	1a1b176
"DISPLAY" = 177 ,	1a1b177
"FROZEN" = 178 ,	1a1b178
"HLPCOM" = 179 ,	1a1b179
"PROGRAM" = 180 ,	1a1b180
"TERSE" = 181 ,	1a1b181
"INDENTING" = 182 ,	1a1b182

Source Code for FORMATTER Subsystem

"UNIVERSAL" = 183 ,	1a1b183
"ENTRY" = 184 ,	1a1b184
"INCLUDE" = 185 ,	1a1b185
"BOTTOM" = 186 ,	1a1b186
"PAGE" = 187 ,	1a1b187
"OFF" = 188 ,	1a1b188
"FULL" = 189 ,	1a1b189
"PARTIAL" = 190 ,	1a1b190
"ANTICIPATORY" = 191 ,	1a1b191
"DEMAND" = 192 ,	1a1b192
"FIXED" = 193 ,	1a1b193
"CONTROL" = 194 ,	1a1b194
"CURRENTCONTEXT" = 195 ,	1a1b195
"FEEDBACK" = 196 ,	1a1b196
"HERALD" = 197 ,	1a1b197
"PRINTOPTIONS" = 198 ,	1a1b198
"PROMPT" = 199 ,	1a1b199
"RECOGNITION" = 200 ,	1a1b200
"STARTUP" = 201 ,	1a1b201
"LEVELADJUST" = 202 ,	1a1b202
"REVERSE" = 203 ,	1a1b203
"TEST" = 204 ,	1a1b204
"TASKER" = 205 ,	1a1b205
"LINEPROCESSOR" = 206 ,	1a1b206
"CENTER" = 207 ,	1a1b207

Source Code for FORMATTER Subsystem

"CNTLQ" = 208 ,	1a1b208
"ARC" = 209 ,	1a1b209
"COPIES" = 210 ,	1a1b210
"FORMATTED" = 211 ;	1a1b211

Source Code for FORMATTER Subsystem

```
% COMMON RULES %                                1a2
% ENTITY DEFINITIONS %                            1a2a
    editentity = textent / structure;             1a2a1
% TEXT ENTITY DEFINITIONS %                        1a2b
    textent = text1 / "TEXT" / "LINK";           1a2b1
    text1 = "CHARACTER" / "WORD" / "VISIBLE" / "INVISIBLE" /
    "NUMBER";                                     1a2b2
% STRUCTURE ENTITY DEFINITIONS %                  1a2c
    structure = "STATEMENT" / notstatement;      1a2c1
    notstatement = "GROUP" / "BRANCH" / "PLEX" ; 1a2c2
```

Source Code for FORMATTER Subsystem

```

SUBSYSTEM formatter KEYWORD "FORMATTER" %%                                1a3
COMMAND linsform = "INSERT"                                             1a3a
    dest _ FALSE sent _ FALSE ent _ FALSE                                1a3a1
    ( "MEMO"!1! <"format in file at">                                     1a3a2
        dest _ DSEL("#STATEMENT")                                         1a3a2a
        CLEAR <"Author's Ident / <^U> if login ident:">                 1a3a2b
            ( readoption() sent _ logid() / sent _                         1a3a2b1
                LSEL("#IDENT") )
            ckid(sent)                                                     1a3a2b2
        CLEAR <"Subject:">                                                 1a3a2c
            param _ LSEL("#TEXT")                                           1a3a2c1
        CLEAR <"Routing Symbol List (seperate with spaces):">           1a3a2d
            param2 _ LSEL("#TEXT")                                          1a3a2d1
        CLEAR <"Addressee:">                                              1a3a2e
            param3 _ LSEL("#TEXT")                                          1a3a2e1
        CLEAR <"Any Attachments ?">                                       1a3a2f
            (answ())                                                       1a3a2f1
                CLEAR <"Attachments (seperate with commas):">           1a3a2f1a
                    ent _ LSEL("#TEXT")                                     1a3a2f1a1
                    attach(ent)                                           1a3a2f1a2
                    CONFIRM                                               1a3a2f1a3
                    / ent _ FALSE                                         1a3a2f1b
                xmemo(dest,param,param2,param3)                            1a3a2g
            / "EVALUATORS"!1! <"memo in file at">                         1a3a3
                dest _ DSEL("#STATEMENT")                                  1a3a3a

```

Source Code for FORMATTER Subsystem

```

CLEAR <"Hardware or software buy?...H/S">          1a3a3b
    ent _ LSEL("#TEXT")                            1a3a3b1
CLEAR <"PR Number = PR-B-6-????">                1a3a3c
    param _ LSEL("#NUMBER")                       1a3a3c1
    ckprnum(param)                                1a3a3c2
CLEAR <"Author's Ident / <U> if login ident:">    1a3a3d
    ( readoption() sent _ logid() / sent _
      LSEL("#IDENT") )                            1a3a3d1
    ckid(sent)                                    1a3a3d2
CLEAR <"Second evaluator's ident:">              1a3a3e
    param2 _ LSEL("#IDENT")                       1a3a3e1
    ckid(param2)                                  1a3a3e2
CLEAR <"Third evaluator's ident:">               1a3a3f
    param3 _ LSEL("#IDENT")                       1a3a3f1
    ckid(param3)                                  1a3a3f2
    CONFIRM                                        1a3a3f3
    xeval(dest,ent)                               1a3a3g
/ "OUTLINE" <"for Workstatement or Solesource...W/S ?"> 1a3a4
    ent _ LSEL("#TEXT")                            1a3a4a
CLEAR <"In file at:">                             1a3a4b
    dest _ DSEL("#STATEMENT")                     1a3a4b1
    CONFIRM                                        1a3a4b2
    xout(ent,dest)                                1a3a4c
/ "LETTER"!2! <"format in file at">              1a3a5
    dest _ DSEL("#STATEMENT")                     1a3a5a

```

Source Code for FORMATTER Subsystem

```

CLEAR <"Author's Ident / <^U> if login ident:">      1a3a5b
    ( readoption() sent _ logid() / sent _
      LSEL("#IDENT") )                                1a3a5b1
    cKid(sent)                                       1a3a5b2
CLEAR <"Subject:">                                    1a3a5c
    param _ LSEL("#TEXT")                            1a3a5c1
CLEAR <"Addressee:">                                  1a3a5d
    param2 _ LSEL("#TEXT")                           1a3a5d1
CLEAR <"Company Name, (abbreviate):">                 1a3a5e
    param3 _ LSEL("#TEXT")                           1a3a5e1
CLEAR <"Street Address: ">                            1a3a5f
    param4 _ LSEL("#TEXT")                           1a3a5f1
CLEAR <"City State Zipcode:">                        1a3a5g
    ent _ LSEL("#TEXT")                              1a3a5g1
CLEAR <"Any Attachments ?">                          1a3a5h
    (answ())                                          1a3a5h1
    CLEAR <"Attachments (seperate with commas):"> 1a3a5h1a
        sent _ LSEL("#TEXT")                        1a3a5h1a1
        attach(sent)                               1a3a5h1a2
        CONFIRM                                    1a3a5h1a3
        / sent _ FALSE)                            1a3a5h1b
    xlett(dest,param,param2,param3,param4,ent)      1a3a5i
/ "TRIPREPORT"!3! <"format in file at">           1a3a6
    dest _ DSEL("#STATEMENT")                      1a3a6a
CONFIRM                                             1a3a6b

```


Source Code for FORMATTER Subsystem

```

xtrip(dest) 1a3a6c
/ "WORKSTATEMENT"!4! <"format in file at"> 1a3a7
  dest _ DSEL("#STATEMENT") 1a3a7a
  CLEAR <"PR Number = PR-B-6-?????"> 1a3a7b
    param _ LSEL("#NUMBER") 1a3a7b1
      ckprnum(param) 1a3a7b1a
  CLEAR <"Title of Procurement"> 1a3a7c
    param2 _ LSEL("#TEXT") 1a3a7c1
    cktitle(param2) 1a3a7c2
  CONFIRM 1a3a7c3
  xwork(dest) 1a3a7c4
/ "SOLESOURCE"!5! <"format in file at"> 1a3a8
  dest _ DSEL("#STATEMENT") 1a3a8a
  CLEAR <"Author's Ident / <^U> if login ident:"> 1a3a8b
    ( readoption() sent _ logid() / sent _
    LSEL("#IDENT") ) 1a3a8b1
    ckid(sent) 1a3a8b2
  CLEAR <"PR Number = PR-B-6-?????"> 1a3a8c
    param _ LSEL("#NUMBER") 1a3a8c1
      ckprnum(param) 1a3a8c1a
  CLEAR <"Title of Procurement"> 1a3a8d
    param2 _ LSEL("#TEXT") 1a3a8d1
    cktitle(param2) 1a3a8d2
  CONFIRM 1a3a8d3
  xsole(dest) 1a3a8d4

```

Source Code for FORMATTER Subsystem

);	1a3a9
END.	1a3b
FINISH	1a4

Source Code for FORMATTER Subsystem

```
FILE lformat % USING (110,) TO (format,subsys,) % 1b
  DECLARE STRING aname[100], name2[100], name3[100], aphone[50],
  numatch[10], atch1[100], atch2[100], atch3[100], atch4[100],
  atch5[100], prnum[10], prtitt[100]; 1b1
```

Source Code for FORMATTER Subsystem

(logid) PROCEDURE % put login ident in record %	1b2
%FORMALS%	1b2a
(result, %result record%	1b2a1
parsemode); %parsing, backup, cleanup%	1b2a2
REF result;	1b2b
LOCAL TEXT POINTER lptr1, lptr2;	1b2c
CASE parsemode OF	1b2d
= parsing:	1b2d1
BEGIN	1b2d1a
FIND SF(*initsr*) ^lptr1 SE(lptr1) ^lptr2;	1b2d1b
result _ lptr1; result[1] _ lptr1[1];	1b2d1c
result[2] _ lptr2; result[3] _ lptr2[1];	1b2d1d
END;	1b2d1e
ENDCASE;	1b2d2
RETURN(&result);	1b2e
END.	1b2f

Source Code for FORMATTER Subsystem

```

(cKid) %checks 3 idents...of author and evaluators %          1b3
PROCEDURE                                                       1b3a
  %FORMALS%                                                     1b3a1
    (result, %result record%                                    1b3a1a
    parsemode,          %parsing, backup, cleanup%           1b3a1b
    ident); %ident%                                           1b3a1c
  REF result, ident;                                           1b3a2
  LOCAL TEXT POINTER cptr1, cptr2 ;                             1b3a3
  LOCAL STRING idstr[1000], idinfo[2000], ext[100];           1b3a4
  CASE parsemode OF                                           1b3b
    = parsing:                                                 1b3b1
      BEGIN                                                    1b3b1a
        dismes (2,$"Checking ident...") ;                    1b3b1a1
        cptr1 _ ident; cptr1[1] _ ident[1];                  1b3b1a2
        cptr2 _ ident[2]; cptr2[1] _ ident[3];               1b3b1a3
        *idstr* _ +cptr1 cptr2 ;                              1b3b1a4
        IF NOT ckident($idstr, sidinfo, 0) THEN              1b3b1a5
          BEGIN                                               1b3b1a5a
            dismes(2,$"Invalid ident -- retype");           1b3b1a5b
            RETURN(FALSE);                                    1b3b1a5c
          END;                                                1b3b1a5d
        getifnf($idinfo, $idstr);                             1b3b1a6
        IF aname.L = 0                                        1b3b1a7
          THEN                                                1b3b1a8
            BEGIN                                             1b3b1a8a

```

Source Code for FORMATTER Subsystem

```

getiphone($idinfo, $aphone, 0, 0);                1b3b1a8a1
IF FIND SF(*aphone*) ['=] ^cptr1 4sCH ^cptr2
THEN                                              1b3b1a8a2
*aphone* _ cptr1 cptr2 ;                        1b3b1a8a3
*ext* _ "Phone extension is,,,", *aphone*;    1b3b1a8a4
dismes (2,$ext) ;                               1b3b1a8a5
*aname* _ *idstr*;                              1b3b1a8a6
END                                              1b3b1a8b
ELSE                                            1b3b1a9
BEGIN                                           1b3b1a9a
IF name2,L = 0                                  1b3b1a9a1
THEN *name2* _ *idstr*                          1b3b1a9a2
ELSE *name3* _ *idstr*                          1b3b1a9a3
END;                                           1b3b1a9b
END;                                           1b3b1b
= backup; = cleanup;                            1b3b2
ENDCASE;                                        1b3c
RETURN;                                         1b3d
END,                                           1b3e
(xout) %inserts outline for workstatements and solesource% 1b4
PROCEDURE                                       1b4a
%FORMALS%                                       1b4a1
(result, %result record%                        1b4a1a
parsemode, %parsing, backup, cleanup%         1b4a1b
ws, %workstatement or solosource%             1b4a1c

```

Source Code for FORMATTER Subsystem

```

        outptr); %pointer to file%                1b4a1d
REF result, ws, outptr;                          1b4a2
LOCAL TEXT POINTER optr1, optr2, csptr;          1b4a3
LOCAL STRING tempstr[100];                      1b4a4
CASE parsemode OF                                1b4b
= parsing:                                       1b4b1
    BEGIN                                        1b4b1a
        %Set up for recreate display%           1b4b1a1
        dspset (dspallf, endfil, endfil, endfil) ; 1b4b1a1a
        %initialize pointers and strings%       1b4b1a2
        csptr _ outptr ;                        1b4b1a2a
        csptr.stpsid _ origin;                  1b4b1a2b
        tempstr,L _ 0;                          1b4b1a2c
        dismes (2,$"Inserting Outline...") ;    1b4b1a2d
        %insert outlines%                       1b4b1a3
        optr1 _ ws; optr1[1] _ ws[1];          1b4b1a3a
        optr2 _ ws[2]; optr2[1] _ ws[3];      1b4b1a3b
        *tempstr* _ +optr1 optr2 ;            1b4b1a3c
        CCPOS SF(*tempstr*);                   1b4b1a3d
        CASE READC OF                            1b4b1a3e
            = 'W:                                1b4b1a3e1
                BEGIN                            1b4b1a3e1a
                    *tempstr* _ "Objective: "; 1b4b1a3e1a1
                    FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
                    ^optr2;                     1b4b1a3e1a2

```

Source Code for FORMATTER Subsystem

```

cinssta (csptr, 0, soptr1, soptr2);
csptr _ getnxt(csptr);
*tempstr* _ "Scope: ";
FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
^optr2;
cinssta (csptr, 0, soptr1, soptr2);
csptr _ getnxt(csptr);
*tempstr* _ "Background: ";
FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
^optr2;
cinssta (csptr, 0, soptr1, soptr2);
csptr _ getnxt(csptr);
*tempstr* _ "Tasks/Technical
Requirements: ";
FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
^optr2;
cinssta (csptr, 0, soptr1, soptr2);
csptr _ getnxt(csptr);
*tempstr* _ "Reporting Requirements:
";
FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
^optr2;
cinssta (csptr, 0, soptr1, soptr2);
csptr _ getnxt(csptr);
*tempstr* _ "Government Furnished
Property: ";
FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
^optr2;
cinssta (csptr, 0, soptr1, soptr2);

```


Source Code for FORMATTER Subsystem

```

END;                                                    1b4b1a3e1b
= 'S;                                                  1b4b1a3e2
BEGIN                                                  1b4b1a3e2a
    *tempstr* _ "BACKGROUND: ";                      1b4b1a3e2a1
    FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
    ^optr2;                                           1b4b1a3e2a2
    cinssta (csptr, 0, soptr1, soptr2);
                                                    1b4b1a3e2a3
    csptr _ getnxt(csptr);                            1b4b1a3e2a4
    *tempstr* _ "OBJECTIVE: ";                        1b4b1a3e2a5
    FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
    ^optr2;                                           1b4b1a3e2a6
    cinssta (csptr, 0, soptr1, soptr2);
                                                    1b4b1a3e2a7
    csptr _ getnxt(csptr);                            1b4b1a3e2a8
    *tempstr* _ "SOURCE: ";                           1b4b1a3e2a9
    FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
    ^optr2;                                           1b4b1a3e2a10
    cinssta (csptr, 0, soptr1, soptr2);
                                                    1b4b1a3e2a11
    csptr _ getnxt(csptr);                            1b4b1a3e2a12
    *tempstr* _ "JUSTIFICATION: ";                   1b4b1a3e2a13
    FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
    ^optr2;                                           1b4b1a3e2a14
    cinssta (csptr, 0, soptr1, soptr2);
                                                    1b4b1a3e2a15
    csptr _ getnxt(csptr);                            1b4b1a3e2a16
    *tempstr* _ "ACTION TAKEN TO AVERT
    SUBSEQUENT SOLE SOURCE: ";                       1b4b1a3e2a17
    FIND SF(*tempstr*) ^optr1 SE(*tempstr*)
    ^optr2;                                           1b4b1a3e2a18
    cinssta (csptr, 0, soptr1, soptr2);
                                                    1b4b1a3e2a19

```

Source Code for FORMATTER Subsystem

```

csptr _ getnxt(Csptr);                1b4b1a3e2a20
*tempstr* _ "PAST PERFORMANCE: ";    1b4b1a3e2a21
FIND SF(*tempstr*) ^optr1 SE(*tempstr*) 1b4b1a3e2a22
^optr2;                                1b4b1a3e2a23
cinssta (csptr, 0, soptr1, soptr2);  1b4b1a3e2b
END;                                    1b4b1a3e3
ENDCASE;                                1b4b1b
END;                                     1b4b2
= backup: = cleanup:                   1b4c
ENDCASE;                                 1b4d
RETURN;                                  1b4e
END.

```

Source Code for FORMATTER Subsystem

```

(attach) %determine attachments for memo %                                1b5
PROCEDURE                                                                1b5a
    %FORMALS%                                                            1b5a1
        (result, %result record%                                       1b5a1a
        parsemode, %parsing, backup: cleanup%                          1b5a1b
        atch); %attachments%                                           1b5a1c
    REF result, atch;                                                  1b5a2
    LOCAL TEXT POINTER aptr1, aptr2 ;                                    1b5a3
    LOCAL STRING char[1], atchs[300];                                    1b5a4
    LOCAL noa; %number of attachments%                                   1b5a5
CASE parsemode OF                                                    1b5b
    = parsing:                                                         1b5b1
        BEGIN                                                            1b5b1a
            numatch,L _ atch1,L _ atch2,L _ atch3,L _ atch4,L _
            atch5,L _ 0;                                               1b5b1a1
            BEGIN                                                        1b5b1a2
                aptr1 _ atch; aptr1[1] _ atch[1];                    1b5b1a2a
                aptr2 _ atch[2]; aptr2[1] _ atch[3];                1b5b1a2b
                *atchs* _ aptr1 aptr2;                                  1b5b1a2c
                noa _ 1;                                               1b5b1a2d
                FIND SF(*atchs*) [L] ^aptr1 _aptr1 SE(*atchs*)
                SNP ^aptr2 ;                                           1b5b1a2e
                IF (char _ READC ) = ', THEN                            1b5b1a2f
                    FIND ^aptr2;                                       1b5b1a2f1
                *atchs* _ aptr1 aptr2;                                  1b5b1a2g
                CCPOS SF(*atchs*);                                       1b5b1a2h

```

Source Code for FORMATTER Subsystem

```

UNTIL (char = READC) = ENDCHR DO                                1b5b1a21
    IF char = ', THEN BUMP noa;                                  1b5b1a2i1
CASE noa OF                                                    1b5b1a2j
    = 1:                                                         1b5b1a2j1
        BEGIN                                                  1b5b1a2j1a
            *numatch* = "1 Atch";                               1b5b1a2j1a1
            *atch1* = *atchs*;                                  1b5b1a2j1a2
        END;                                                    1b5b1a2j1b
    = 2:                                                         1b5b1a2j2
        BEGIN                                                  1b5b1a2j2a
            *numatch* = "2 Atchs";                              1b5b1a2j2a1
            FIND SF(*atchs*) "aptr1 [',] "aptr2
            _aptr2;                                             1b5b1a2j2a2
            *atch1* = "1. ", aptr1 aptr2;                     1b5b1a2j2a3
            READC;                                              1b5b1a2j2a4
            FIND SNP "aptr1;                                    1b5b1a2j2a5
            *atch2* = "2. ", aptr1 SE(aptr1)                   1b5b1a2j2a6
        END;                                                    1b5b1a2j2b
    = 3:                                                         1b5b1a2j3
        BEGIN                                                  1b5b1a2j3a
            *numatch* = "3 Atchs";                              1b5b1a2j3a1
            FIND SF(*atchs*) "aptr1 [',] "aptr2
            _aptr2;                                             1b5b1a2j3a2
            *atch1* = "1. ", aptr1 aptr2;                     1b5b1a2j3a3
            READC;                                              1b5b1a2j3a4
            FIND SNP "aptr1 [',] "aptr2 _aptr2;               1b5b1a2j3a5

```

Source Code for FORMATTER Subsystem

```

*atch2* _ "2. ", aptr1 aptr2;      1b5b1a2j3a6
READC;                               1b5b1a2j3a7
FIND SNP ^aptr1;                     1b5b1a2j3a8
*atch3* _ "3. ", aptr1 SE(aptr1); 1b5b1a2j3a9
END;                                   1b5b1a2j3b
= 4:                                   1b5b1a2j4
BEGIN                                  1b5b1a2j4a
*numatch* _ "4 Atchs";                1b5b1a2j4a1
FIND SF(*atchs*) ^aptr1 [^,] ^aptr2
_aptr2;                               1b5b1a2j4a2
*atch1* _ "1. ", aptr1 aptr2;        1b5b1a2j4a3
READC;                               1b5b1a2j4a4
FIND SNP ^aptr1 [^,] ^aptr2 _aptr2; 1b5b1a2j4a5
*atch2* _ "2. ", aptr1 aptr2;        1b5b1a2j4a6
READC;                               1b5b1a2j4a7
FIND SNP ^aptr1 [^,] ^aptr2 _aptr2; 1b5b1a2j4a8
*atch3* _ "3. ", aptr1 aptr2;        1b5b1a2j4a9
READC;                               1b5b1a2j4a10
FIND SNP ^aptr1;                     1b5b1a2j4a11
*atch4* _ "4. ", aptr1 SE(aptr1); 1b5b1a2j4a12
END;                                   1b5b1a2j4b
= 5:                                   1b5b1a2j5
BEGIN                                  1b5b1a2j5a
*numatch* _ "5 Atchs";                1b5b1a2j5a1
FIND SF(*atchs*) ^aptr1 [^,] ^aptr2
_aptr2;                               1b5b1a2j5a2

```

Source Code for FORMATTER Subsystem

```

*atch1* _ "1. ", aptr1 aptr2;      1b5b1a2j5a3
READC;                               1b5b1a2j5a4
FIND SNP "aptr1 [',] "aptr2 _aptr2; 1b5b1a2j5a5
*atch2* _ "2. ", aptr1 aptr2;      1b5b1a2j5a6
READC;                               1b5b1a2j5a7
FIND SNP "aptr1 [',] "aptr2 _aptr2; 1b5b1a2j5a8
*atch3* _ "3. ", aptr1 aptr2;      1b5b1a2j5a9
READC;                               1b5b1a2j5a10
FIND SNP "aptr1 [',] "aptr2 _aptr2; 1b5b1a2j5a11
*atch4* _ "4. ", aptr1 aptr2;      1b5b1a2j5a12
READC;                               1b5b1a2j5a13
FIND SNP "aptr1;                    1b5b1a2j5a14
*atch5* _ "5. ", aptr1 SE(aptr1); 1b5b1a2j5a15
END;                                  1b5b1a2j5b
ENDCASE;                             1b5b1a2k
END;                                  1b5b1a3
END;                                  1b5b1b
= backup; = cleanup;                 1b5b2
ENDCASE;                             1b5c
RETURN;                               1b5d
END.                                  1b5e

```

Source Code for FORMATTER Subsystem

```

(xmemo) % format for printing as a memo %                               1b6
PROCEDURE                                                                1b6a
    %FCRMLS%                                                            1b6a1
        (result, %result record%                                       1b6a1a
        parsemode,          %parsing, backup: cleanup%                1b6a1b
        memoptr, %pointer to file to be formatted%                    1b6a1c
        subj,      %Subject of memo = param%                          1b6a1d
        rout,      %Routing symbol list = param2%                    1b6a1e
        adse); %Addressee name = param3%                              1b6a1f
    REF result, memoptr, subj, rout, adse;                               1b6a2
    LOCAL TEXT POINTER csptr, nsptr, ptr1, ptr2, ptr3, ptr4,
    ptr5;                                                                1b6a3
    LOCAL STRING tempstr[1000];                                         1b6a4
CASE parsemode OF                                                       1b6b
    = parsing:                                                            1b6b1
        BEGIN                                                            1b6b1a
            %Set up for recreate display%                               1b6b1b
                dpset (dspallf, endfil, endfil, endfil) ;              1b6b1b1
            %initialize pointers and strings%                            1b6b1c
                csptr _ memoptr ;                                       1b6b1c1
                csptr,stepsid _ origin;                                  1b6b1c2
                nsptr _ getnxt(csptr) ;                                  1b6b1c3
                tempstr,L _ 0;                                          1b6b1c4
                dismes (2,S"Inserting Memo Format,..");                1b6b1c5
            %Put space before each statement in body of memo%          1b6b1d

```

Source Code for FORMATTER Subsystem

```

UNTIL nsptr = endfil DO                                1b6b1d1
    BEGIN                                              1b6b1d1a
        csptr = nsptr;                                1b6b1d1a1
        ST nsptr = ' , SF(nsptr) SE(nsptr);          1b6b1d1a2
        nsptr = getnxt(csptr);                        1b6b1d1a3
    END;                                              1b6b1d1b
%Put directives in origin statement%                 1b6b1e
    csptr = memoptr;                                  1b6b1e1
    csptr, stpsid = origin;                           1b6b1e2
    FIND SF(csptr) [" ;;;"/ENDCHR] ^csptr;           1b6b1e3
    ST csptr = SF(csptr) csptr, ".TM=2;.H1=", " ", "FOR
    OFFICIAL USE ONLY", " ", " ; .H1P=C; .YFH=5; .RM=72;
    .F=", " ", "FOR OFFICIAL USE ONLY", " ", " ; .FP=C;
    .PN=0; .NUMDASH=0; .LBS=1; ", " , .ILev=5; .IMax=110;
    .LM=-55; .IFirst=55; .TABSTOPS=55; .SN=Off; .D=Off;
    .PEL;
    ";                                              1b6b1e4
%Insert "Reply Attn of"%                             1b6b1f
    *tempstr* = "ISIM/X", *aphone*, "      .GD;.GCR=2;"; 1b6b1f1
    ST csptr = SF(csptr) SE(csptr), *tempstr*;        1b6b1f2
%Insert "Subject"%                                    1b6b1g
    ptr1 = subj; ptr1[1] = subj[1];                   1b6b1g1
    ptr2 = subj[2]; ptr2[1] = subj[3];                1b6b1g2
    *tempstr* = ptr1 ptr2;                             1b6b1g3
%to deal with 2 line subjects%                       1b6b1g4
    IF NOT FIND SF(*tempstr*) [CR] THEN               1b6b1g4a
        *tempstr* = ptr1 ptr2, ".GCR=3;";            1b6b1g4a1
    ELSE *tempstr* = ptr1 ptr2, ".GCR=2;";            1b6b1g4b

```


Source Code for FORMATTER Subsystem

```

      ST csptr _ SF(csptr) SE(csptr), *tempstr*;           1b6b1g5
%Insert "To"%                                           1b6b1h
      %insert routing%                                   1b6b1h1
      ptr1 _ rout; ptr1[1] _ rout[1];                   1b6b1h1a
      ptr2 _ rout[2]; ptr2[1] _ rout[3];                 1b6b1h1b
      *tempstr* _ ptr1 ptr2;                             1b6b1h1c
      %trim ends of string and capitalize%              1b6b1h1d
      FIND SF(*tempstr*) SNP ^ptr1 SE(*tempstr*)
      SNP ^ptr2;                                         1b6b1h1d1
      *tempstr* _ +ptr1 ptr2;                           1b6b1h1d2
      %turn off FOUO if routing not to PM%             1b6b1h1e
      CCPOS SE(*tempstr*);                              1b6b1h1f
      IF NOT FIND ["MP"] THEN                           1b6b1h1g
      BEGIN                                             1b6b1h1g1
          FIND SE(csptr) ["EP, "] ^ptr1;                1b6b1h1g1a
          ST csptr _ SF(csptr) ptr1, " ,HSW=Off;
          ,FSW=Off; ,TM=8;", ptr1 SE(csptr);            1b6b1h1g1b
      END;                                               1b6b1h1g2
      CCPOS SF(*tempstr*);                              1b6b1h1h
      IF NOT FIND [SP] THEN                             1b6b1h1i
          *tempstr* _ "RADC/", *tempstr*                 1b6b1h1i1
      ELSE                                               1b6b1h1j
      BEGIN                                             1b6b1h1j1
          CCPOS SE(*tempstr*);                          1b6b1h1j1a
          LOOP                                           1b6b1h1j1b
          BEGIN                                          1b6b1h1j1b1

```

Source Code for FORMATTER Subsystem

```

IF FIND ENDCHR THEN EXIT LOOP ELSE 1b6b1h1j1b2
BEGIN 1b6b1h1j1b2a
    FIND SNP SL ^ptr1; 1b6b1h1j1b2a1
    IF FIND ENDCHR THEN EXIT LOOP
    ELSE 1b6b1h1j1b2a2
        *tempstr* _ SF(ptr1) ptr1,
        ",GCR;", ptr1 SE(ptr1); 1b6b1h1j1b2a3
    END 1b6b1h1j1b2b
END; 1b6b1h1j1b3

    *tempstr* _ "RADC/.IREST=60;", *tempstr*,
    ".IREST=0;
    IN TURN"; 1b6b1h1j1c
END; 1b6b1h1j2

ST csptr _ SF(csptr) SE(csptr), *tempstr*; 1b6b1h1k

%append addressee name, indenting and numbering
directives% 1b6b1h2

ptr1 _ adse; ptr1[1] _ adse[1]; 1b6b1h2a
ptr2 _ adse[2]; ptr2[1] _ adse[3]; 1b6b1h2b
*tempstr* _ ptr1 ptr2 ; 1b6b1h2c
%make sure its in the form of (D, Stone)% 1b6b1h2d

FIND SF(*tempstr*) [L] ^ptr1 _ptr1 ^ptr2 [NP]
[L] ^ptr3 _ptr3 ^ptr4; 1b6b1h2d1

*tempstr* _ '(, +ptr1 ptr2, ", ", +ptr3 ptr4,
-ptr4 SE(ptr4), ')'; 1b6b1h2d2

FIND SE(csptr) ("NRUT NI
" / TRUE) ^ptr1; 1b6b1h2e

ST csptr _ SF(csptr) ptr1, *tempstr*, ptr1
SE(csptr), ".PXN[1]=Dec+Period;
.PXN[2]=LL+Period; .PXN[3]=Dec+Parens;
.PXN[4]=LL+Parens; .PXNSHOW=ALL;"; 1b6b1h2f

```

Source Code for FORMATTER Subsystem

```

%Insert "Signature Block" statement%                                1b6b11
    csptr _ getend(csptr);                                          1b6b111
    ST csptr _ SF(csptr) SE(csptr),
    ",ILEV=0;PFIT;PXNSHOW=OFF;GRAB=10;";                            1b6b112
    astruc ($aname) ;                                              1b6b113
    *tempstr* _ ",GYBS=3;", *aname*, "
    Information Management Sciences Section
    Information Processing Branch
    ;                                                                1b6b114
    FIND SF(*tempstr*) ^ptr1 SE(*tempstr*) ^ptr2 ;                1b6b115
    cinssta (csptr, 0, sptr1, sptr2);                               1b6b116
%append attachments to Signature Block%                             1b6b1j
    csptr _ getnxt(csptr);                                          1b6b1j1
    IF FIND SE(csptr) [CR] ^ptr5 [CR] ^ptr4 [CR] ^ptr3
    [CR] ^ptr2 [CR] ^ptr1 THEN                                     1b6b1j2
    ST ptr1 _ SF(ptr1) ptr1, *numatch*, ptr1 ptr2,
    *atch1*, ptr2 ptr3, *atch2*, ptr3 ptr4, *atch3*,
    ptr4 ptr5, *atch4*, ptr5 SE(ptr5), *atch5*;                   1b6b1j3
%reset globals%                                                  1b6b1k
    aname,L _ aphone,L _ 0;                                       1b6b1k1
    END;                                                            1b6b1l
= backup: =cleanup:                                             1b6b2
    ENDCASE;                                                       1b6b3
    RETURN;                                                         1b6c
END.                                                                1b6d

```

```

(xlett) % format for printing as a letter %                               1b7
PROCEDURE                                                                    1b7a
    %FORMALS%                                                                    1b7a1
        (result, %result record%                                               1b7a1a
        parsemode,          %parsing, backup: cleanup%                         1b7a1b
        lettpr, %pointer to file to be formatted%                               1b7a1c
        subj,   %Subject of memo = param%                                       1b7a1d
        adsee,  %Addressee name = param2%                                       1b7a1e
        coname, %company name = param3%                                       1b7a1f
        costadr, %company street address = param4%                               1b7a1g
        cocsz); %company city, state and zip = ent%                             1b7a1h
REF result, lettpr, subj, adsee, coname, costadr,
cocsz;                                                                    1b7a2
LOCAL TEXT POINTER cspr, nspr, ptr1, ptr2, ptr3, ptr4,
ptr5;                                                                    1b7a3
LOCAL STRING tempstr[1000] ;                                             1b7a4
CASE parsemode OF                                                         1b7b
    = parsing:                                                            1b7b1
        BEGIN                                                            1b7b1a
            %Set up for recreate display%                                     1b7b1b
                dspset (dspallf, endfil, endfil, endfil) ;                 1b7b1b1
            %initialize pointers and strings%                                 1b7b1c
                cspr _ lettpr ;                                             1b7b1c1
                cspr, stpsid _ origin;                                       1b7b1c2
                nspr _ getnxt(cspr) ;                                         1b7b1c3
                tempstr, L _ 0;                                               1b7b1c4

```

Source Code for FORMATTER Subsystem

```

        dismes (2,s"Inserting Letter Format...");          1b7b1c5
%Put space before each statement in body of memo%      1b7b1d
        UNTIL nsptr = endfil DO                          1b7b1d1
                BEGIN                                    1b7b1d1a
                        csptr _ nsptr;                   1b7b1d1a1
                        ST nsptr _ ' , SF(nsptr) SE(nsptr); 1b7b1d1a2
                        nsptr _ getnxt(csptr);           1b7b1d1a3
                END;                                     1b7b1d1b
%Put directives in origin statement%                   1b7b1e
        csptr _ lettpr;                                  1b7b1e1
        csptr, stpsid _ origin;                          1b7b1e2
        FIND SF(csptr) [" ;;;"/ENDCHR] "csptr;         1b7b1e3
        ST csptr _ SF(csptr) csptr, ".TM=8; .RM=72; .FSW=0;
        .PN=0; .NUMDASH=0; .LBS=1;"; " , ILev=5; .IMax=110;
        .LM=-55; .IFirst=55; .NAMES=0; .TABSTOPS=55;
        .SN=Off; .D=Off; .PEL;
        ";                                              1b7b1e4
%Insert "Reply Attn of"%                               1b7b1f
        *tempstr* _ "ISIM/315-330-", *aphone*, "
        .GD; .GCR=2;";                                  1b7b1f1
        ST csptr _ SF(csptr) SE(csptr), *tempstr*;     1b7b1f2
%Insert "Subject"%                                     1b7b1g
        ptr1 _ subj; ptr1[1] _ subj[1];                 1b7b1g1
        ptr2 _ subj[2]; ptr2[1] _ subj[3];             1b7b1g2
        *tempstr* _ ptr1 ptr2;                          1b7b1g3
        %to deal with 2 line subjects%                  1b7b1g4
                IF NOT FIND SF(*tempstr*) [CR] THEN    1b7b1g4a

```

Source Code for FORMATTER Subsystem

```

        *tempstr* _ ptr1 ptr2, ",GCR=3;"                1b7b1g4a1
    ELSE *tempstr* _ ptr1 ptr2, ",GCR=2;";              1b7b1g4b
    SI csptr _ SF(csptr) SE(csptr), *tempstr*;          1b7b1g5
%Insert "To"%                                          1b7b1h
    ptr1 _ coname; ptr1[1] _ coname[1];                 1b7b1h1
    ptr2 _ coname[2]; ptr2[1] _ coname[3];              1b7b1h2
    *tempstr* _ ptr1 ptr2 ;                             1b7b1h3
    SI csptr _ SF(csptr) SE(csptr), *tempstr*, " (";   1b7b1h4
    ptr1 _ adsee; ptr1[1] _ adsee[1];                   1b7b1h5
    ptr2 _ adsee[2]; ptr2[1] _ adsee[3];                1b7b1h6
    *tempstr* _ ptr1 ptr2 ;                             1b7b1h7
    SI csptr _ SF(csptr) SE(csptr), *tempstr*,
    " ),GCR;";                                          1b7b1h8
    ptr1 _ costadr; ptr1[1] _ costadr[1];               1b7b1h9
    ptr2 _ costadr[2]; ptr2[1] _ costadr[3];             1b7b1h10
    *tempstr* _ ptr1 ptr2 ;                             1b7b1h11
    SI csptr _ SF(csptr) SE(csptr), *tempstr*, ",GCR;"; 1b7b1h12
    ptr1 _ cocsz; ptr1[1] _ cocsz[1];                   1b7b1h13
    ptr2 _ cocsz[2]; ptr2[1] _ cocsz[3];                1b7b1h14
    *tempstr* _ ptr1 ptr2 ;                             1b7b1h15
    SI csptr _ SF(csptr) SE(csptr), *tempstr*, ",GCR;
    ,PXN[1]=Dec+Period; ,PXN[2]=LL+Period;
    ,PXN[3]=Dec+Parens; ,PXN[4]=LL+Parens;
    ,PXNSHOW=ALL;";                                    1b7b1h16
%Insert "Signature Block" statement%                  1b7b1i
    csptr _ getend(csptr);                              1b7b1i1

```

Source Code for FORMATTER Subsystem

```

SI csptr = SF(csptr) SE(csptr),
",ILEV=0; ,PFIT; ,PXNSHOW=OFF; ,GRAB=10;";
                                                                    1b7b1i2
astruc (saname) ;
                                                                    1b7b1i3
*tempstr* = ",GYBS=3; ", *aname*, "
Information Management Sciences Section
Information Processing Branch

";
                                                                    1b7b1i4
FIND SF(*tempstr*) ^ptr1 SE(*tempstr*) ^ptr2 ;
                                                                    1b7b1i5
cinssta (csptr, 0, sptr1, sptr2);
                                                                    1b7b1i6
%append attachments to Signature Block%
                                                                    1b7b1j
csptr = getnxt(csptr);
                                                                    1b7b1j1
IF FIND SE(csptr) [CR] ^ptr5 [CR] ^ptr4 [CR] ^ptr3
[CR] ^ptr2 [CR] ^ptr1 THEN
                                                                    1b7b1j2
SI ptr1 = SF(ptr1) ptr1, *numatch*, ptr1 ptr2,
*atch1*, ptr2 ptr3, *atch2*, ptr3 ptr4, *atch3*,
ptr4 ptr5, *atch4*, ptr5 SE(ptr5), *atch5*;
                                                                    1b7b1j3
%reset globals%
                                                                    1b7b1k
aname,L = aphone,L = 0;
                                                                    1b7b1k1
END;
                                                                    1b7b1l
= backup: =cleanup:
                                                                    1b7b2
ENDCASE;
                                                                    1b7b3
RETURN;
                                                                    1b7c
END.
                                                                    1b7d

```

```
(xtrip) % format for printing as a trip report %          1b8
PROCEDURE                                                  1b8a
  %FORMALS%                                               1b8a1
    (result, %result record%                               1b8a1a
     parsemode,      %parsing, backup: cleanup%          1b8a1b
     trip); %trip%                                         1b8a1c
  REF result, trip;                                        1b8a2
  LOCAL TEXT POINTER cptr1, cptr2 ;                       1b8a3
CASE parsemode OF                                         1b8b
  = parsing:                                              1b8b1
    BEGIN                                                 1b8b1a
      dismes (2,$"Not implemented yet...") ;            1b8b1b
      %reset globals%                                     1b8b1c
    END;                                                  1b8b1d
  = backup: =cleanup:                                     1b8b2
ENDCASE;                                                 1b8b3
RETURN;                                                  1b8c
END,                                                      1b8d
```


Source Code for FORMATTER Subsystem

```

(ckprnum) % check for valid PR number %                                1b9
PROCEDURE                                                                1b9a
  %FORMALS%                                                              1b9a1
    (result, %result record%                                           1b9a1a
    parsemode, %parsing, backup: cleanup%                               1b9a1b
    prstr); %PR number%                                                1b9a1c
  REF result, prstr;                                                    1b9a2
  LOCAL TEXT POINTER ptr1, ptr2;                                        1b9a3
  CASE parsemode OF                                                    1b9b
    = parsing:                                                            1b9b1
      BEGIN                                                                1b9b1a
        prnum,L - 0;                                                    1b9b1b
        ptr1 - prstr; ptr1[1] - prstr[1];                               1b9b1c
        ptr2 - prstr[2]; ptr2[1] - prstr[3];                           1b9b1d
        *prnum* - ptr1 ptr2 ;                                           1b9b1e
        IF FIND SE(*prnum*) [sNP] ^ptr2 4D ^ptr1 THEN                  1b9b1f
          *prnum* - ptr1 ptr2                                           1b9b1f1
        ELSE                                                                1b9b1g
          BEGIN                                                            1b9b1g1
            dismes (2,s"invalid number, please retype");              1b9b1g2
            RETURN(FALSE);                                              1b9b1g3
          END;                                                            1b9b1g4
        END;                                                            1b9b1h
      = backup: =cleanup:                                                1b9b2
    ENDCASE;                                                            1b9b3

```

Source Code for FORMATTER Subsystem

RETURN(TRUE);

1b9c

END,

1b9d

Source Code for FORMATTER Subsystem

```

(cktitle) %collects title for (xwork) and (xsole) %           1b10
PROCEDURE                                                     1b10a
  %FORMALS%                                                  1b10a1
    (result, %result record%                                1b10a1a
    parsemode,      %parsing, backup: cleanup%            1b10a1b
    title); %Title of procurement%                          1b10a1c
  REF result, title;                                         1b10a2
  LOCAL TEXT POINTER tptr1, tptr2;                          1b10a3
CASE parsemode OF                                           1b10b
  = parsing:                                                 1b10b1
    BEGIN                                                    1b10b1a
      prtit,L _ 0;                                          1b10b1a1
      tptr1 _ title; tptr1[1] _ title[1];                  1b10b1a2
      tptr2 _ title[2]; tptr2[1] _ title[3];               1b10b1a3
      *prtit* _ +tptr1 tptr2;                               1b10b1a4
    END;                                                     1b10b1b
  = backup: =cleanup;                                       1b10b2
ENDCASE;                                                    1b10b3
RETURN(&result);                                           1b10c
END,                                                         1b10d

```

Source Code for FORMATTER Subsystem

```

(xwork) % format for printing as a workstatement %          1b11
PROCEDURE                                                    1b11a
  %FORMALS%                                                  1b11a1
    (result, %result record%                                1b11a1a
    parsemode,      %parsing, backup: cleanup%            1b11a1b
    workptr);      %pointer to file to be formatted%      1b11a1c
  REF result, workptr;                                       1b11a2
  LOCAL TEXT POINTER wptr, wptr1, wptr2;                    1b11a3
  LOCAL STRING titpage[1000];                               1b11a4
  CASE parsemode OF                                         1b11b
    = parsing:                                              1b11b1
      BEGIN                                                1b11b1a
        %Set up for recreate display%                       1b11b1b
          dpset (dspallf, endfil, endfil, endfil);         1b11b1b1
        %initialize pointers and strings%                   1b11b1c
          wptr = workptr ;                                  1b11b1c1
          wptr, stpsid = origin;                             1b11b1c2
          titpage.L = 0;                                     1b11b1c3
          dismes (2, $"Inserting workstatement format..."); 1b11b1c4
        %insert directives in origin statement%             1b11b1d
          FIND SF(wptr) [" ;;;"/ENDCHR] ^wptr;            1b11b1d1
          ST wptr = SF(wptr) wptr, ".H1=", " ", "FOR OFFICIAL
          USE ONLY", " ", "; .H1P=C; .H2=", " ", "PR=B-6-",
          #prnum*, " ", "; .H2P=FR; .RM=72; .F=", " ", ".GPN;
          FOR OFFICIAL USE ONLY", " ", "; .FP=C; .FSW=1;
          .PNTYPE=41; .NUMDASH=0; .LBS=1; ", ".Ilev=0;
          .NAMES=0; .TABSTOPS=6,9,20,62; .PXNSHOW=Off;

```

Source Code for FORMATTER Subsystem

```

.D=Off; .PEL;
";
1b11b1d2

%insert table of contents in origin statement%
1b11b1e

ST wptr = SF(wptr) SE(wptr), ".CENTER;TABLE OF
CONTENTS,GCR=3;", " PARAGRAPH
SUBJECT PAGE,GCR=2;", "
1.0 Objective #,GCR=2;", " 2.0
Scope #,GCR=2;", " 3.0 Background
#,GCR=2;", " 4.0 Tasks/Technical
Requirements #,GCR=2;", " 5.0
Report Requirements #,GCR=2;", " 6.0
Government Furnished Property #,PXNSHOW<2;,PES;"; 1b11b1e1

%turn numbering on for all levels under 4.0%
1b11b1f

wptr = getnxt(wptr); 1b11b1f1
wptr = getsuc(wptr); 1b11b1f2
wptr = getsuc(wptr); 1b11b1f3
wptr = getend(wptr); 1b11b1f4
CCPOS SF(wptr); 1b11b1f5
ST wptr = SF(wptr) SE(wptr), ".PXNSHOW=All;"; 1b11b1f6

%turn numbering off for lower levels for 5,0 & 6.0%
1b11b1g

wptr = getnxt(wptr); 1b11b1g1
wptr = getend(wptr); 1b11b1g2
CCPOS SF(wptr); 1b11b1g3
ST wptr = SF(wptr) SE(wptr), ".PXNSHOW<2;"; 1b11b1g4

%insert directives at end of file%
1b11b1h

wptr = getnxt(wptr); 1b11b1h1
wptr = getall(wptr); 1b11b1h2
wptr = getend(wptr); 1b11b1h3
ST wptr = SF(wptr) SE(wptr),

```

Source Code for FORMATTER Subsystem

```

",PXNSHOW=Off;.H2SW=Off;.F=", " ", "FOR OFFICIAL USE
ONLY", " ", ";.PES;";
1b11b1h4

%insert title page at end of file%
1b11b1i

CCPOS SF(wptra);
1b11b1i1

*titpage* _ ".GCR=3;.CENTER=50;"; "
ROME AIR DEVELOPMENT CENTER", "
GRIFFISS AIR FORCE BASE", "
NEW YORK.GCR=3;"; "
STATEMENT OF WORK.GCR=3;"; "
FOR.GCR=3;"; "
, *prtita*, ".GCR=10;"; "
PR-B-6-"; *prnum*, "
.GCR=10;.GD;";
1b11b1i2

FIND SF(*titpage*) ^wptra SE(*titpage*) ^wptra2 ;
1b11b1i3

cinssta (wptra, 0, swptra1, swptra2);
1b11b1i4

%reset globals%
1b11b1j

prnum,L _ prtita,L _ 0;
1b11b1j1

END;
1b11b1k

= backup; =cleanup;
1b11b2

ENDCASE;
1b11b3

RETURN(TRUE);
1b11c

END,
1b11d

```

Source Code for FORMATTER Subsystem

```

(xsole) % format for printing as a sole source statement %      1b12
PROCEDURE                                                         1b12a
  %FORMALS%                                                       1b12a1
    (result, %result record%                                       1b12a1a
    parsemode, %parsing, backup: cleanup%                          1b12a1b
    soleptr); %pointer to file to be formatted%                    1b12a1c
  REF result, soleptr;                                           1b12a2
  LOCAL TEXT POINTER ssptr, ssptr1, ssptr2;                       1b12a3
  LOCAL STRING sigstr[500]; %signature block string%              1b12a4
CASE parsemode OF                                               1b12b
  = parsing:                                                       1b12b1
    BEGIN                                                         1b12b1a
      %Set up for recreate display%                                  1b12b1b
        dpset (dspallf, endfil, endfil, endfil);                  1b12b1b1
      %initialize pointers and strings%                              1b12b1c
        ssptr = soleptr ;                                          1b12b1c1
        ssptr, stpsid = origin;                                     1b12b1c2
        sigstr.L = 0;                                             1b12b1c3
        dismes (2, s"Inserting solesource format...");           1b12b1c4
      %insert directives in origin statement%                       1b12b1d
        FIND SF( ssptr) [" ;;;;"/ENDCHR] = ssptr;                1b12b1d1
      ST ssptr = SF( ssptr) ssptr, ".H1=", "", "FOR
OFFICIAL USE ONLY", "", "; .H1P=C; .H2=", "",
"PR-B-6-", *prnum*, "", "; .H2P=FR; .RM=72; .F=",
"", ".GPN;
FOR OFFICIAL USE ONLY", "", "; .FP=C; .FSw=1;
.PNTYPE=41; .NUMDASH=0; .LBS=1;", ".ILev=0;

```

Source Code for FORMATTER Subsystem

```

        ,NAMES=0; .PN=0; .PXNSHOW=Off; .D=Off; .PEL;
        ";
        %insert title origin statement%
        ST ssptr _ SF( ssptr) SE( ssptr), ".CENTER=4;SOLE
        SOURCE STATEMENT,GCR=3;", *prtit*, ".GCR=2;";
        %Insert "Signature Block" statement%
        ssptr _ getend(ssptr);
        ST ssptr _ SF(ssptr) SE(ssptr),
        ".ILEV=0;.PFIT;.PXNSHOW=OFF;.GRAB=10;";
        astruc (saname) ;
        *sigstr* _ ".GYBS=3;", *aname*, "
        Information Management Sciences Section
        Information Processing Branch,GCR=3;", "
        FRANK J. TOMAINI, Chief
        Information Processing Branch
        Information Processing Division";
        FIND SF(*sigstr*) ^ssptr1 SE(*sigstr*) ^ssptr2;
        cinssta (ssptr, 0, sssptr1, sssptr2);
        %reset globals%
        prnum,L _ prtit,L _ aname,L _ 0;
        END;
        = backup; =cleanup;
        ENDCASE;
        RETURN(TRUE);
        END,
        (xeval) % format for printing as a evaluator's memo%
        PROCEDURE
        %FORMALS%

```


Source Code for FORMATTER Subsystem

```

(result, %result record%                                1b13a1a
parsemode,      %parsing, backup: cleanup%            1b13a1b
evalptr, %pointer to file to be formatted%            1b13a1c
soft); %hard or software buy%                          1b13a1d
REF result, evalptr, soft;                              1b13a2
LOCAL TEXT POINTER eptr, eptri, eptri2;                1b13a3
LOCAL STRING tempstr[1000], hsstr[10];                 1b13a4
CASE parsemode OF                                     1b13b
= parsing:                                             1b13b1
  BEGIN                                               1b13b1a
  %Set up for recreate display%                       1b13b1b
    dpset (dspallf, endfil, endfil, endfil);          1b13b1b1
  %initialize pointers and strings%                   1b13b1c
    eptr _ evalptr ;                                  1b13b1c1
    eptri, stpsid _ origin;                            1b13b1c2
    tempstr, L _ hsstr, L _ 0;                          1b13b1c3
    dismes (2, s"Inserting evaluators' memo...");     1b13b1c4
  %insert directives in origin statement%              1b13b1d
    FIND SF(eptri) [" ;;;"/ENDCHR] ^ eptri;           1b13b1d1
    ST eptri _ SF(eptri) eptri, ".H1=", " ", "FOR OFFICIAL
    USE ONLY", " ", " ", .H1P=C; .RM=72; .YFH=5; .F=" ", " ",
    "FOR OFFICIAL USE ONLY", " ", " ", .FP=C; .FSW=1;
    .NUMDASH=0; .LBS=1; " ", .ILev=5; .NAMES=0; .PN=0;
    .PXN[1]=Dec+Period; .PXN[2]=LL+Period;
    .PXNSHOW=All; .D=Off; .PEL;
    " ", "Subject: List of Evaluation Criteria and
    Evaluators for PR-B-6-", *prnum*, ".GCR=3;IO:
    RADC/PM.GCR; TABSTOPS=50;";                       1b13b1d2
  %insert first statement%                             1b13b1e

```

Source Code for FORMATTER Subsystem

```

eptr1 _ soft;   eptr1[1] _ soft[1];           1b13b1e1
eptr2 _ soft[2]; eptr2[1] _ soft[3];         1b13b1e2
*hsstr* _ +eptr1 eptr2;                       1b13b1e3
CCPOS SF(*hsstr*);                             1b13b1e4
CASE READC OF                                  1b13b1e5
  = 'H:                                         1b13b1e5a
    *hsstr* _ '5;                               1b13b1e5a1
  = 'S:                                         1b13b1e5b
    *hsstr* _ '6;                               1b13b1e5b1
ENDCASE;                                       1b13b1e5c

*tempstr* _ " AFSC form 6", *hsstr*, " is attached
and indicates the only criteria established for
evaluating this PR.";                          1b13b1e6
FIND SF(*tempstr*) ^eptr1 SE(*tempstr*) ^eptr2; 1b13b1e7
cinssta (eptr, 0, septr1, septr2);             1b13b1e8
%Insert second statement%                      1b13b1f
eptr _ getnxt(eptr);                           1b13b1f1

*tempstr* _ " At least three persons will
technically evaluate (by a joint team ", "effort)
(on an individual basis) the offeror/bidder
response to this PR, ", "The following are
considered most qualified in the area of this ",
"procurement and have current indoctrination
certificates on file.", ".GCR=2;
SYMBOL";                                       NAME
                                              1b13b1f2
FIND SF(*tempstr*) ^eptr1 SE(*tempstr*) ^eptr2; 1b13b1f3
cinssta (eptr, 0, septr1, septr2);             1b13b1f4
%insert first evaluator, down a level%         1b13b1g
eptr _ getnxt(eptr);                           1b13b1g1

```

Source Code for FORMATTER Subsystem

```

astruc ($aname);                                1b13b1g2
*tempstr* _ ' , *aname*, "    ISIM (Chairman)"; 1b13b1g3
FIND SF(*tempstr*) ^eptr1 SE(*tempstr*) ^eptr2; 1b13b1g4
cinssta (eptr, -1, sep1, sep2);                 1b13b1g5
%insert second evaluator%                       1b13b1h
eptr = getnxt(eptr);                            1b13b1h1
astruc ($name2);                                1b13b1h2
*tempstr* _ ' , *name2*, "    ISIM";           1b13b1h3
FIND SF(*tempstr*) ^eptr1 SE(*tempstr*) ^eptr2; 1b13b1h4
cinssta (eptr, 0, sep1, sep2);                 1b13b1h5
%insert third evaluator%                       1b13b1i
eptr = getnxt(eptr);                            1b13b1i1
astruc ($name3);                                1b13b1i2
*tempstr* _ ' , *name3*, "
ISIM,PXNSHOW=OFF;,ILEV=0";                     1b13b1i3
FIND SF(*tempstr*) ^eptr1 SE(*tempstr*) ^eptr2; 1b13b1i4
cinssta (eptr, 0, sep1, sep2);                 1b13b1i5
%Insert "Signature Block" statement%           1b13b1j
eptr = getnxt(eptr);                            1b13b1j1
astruc ($aname) ;                               1b13b1j2
*tempstr* _ ",GYBS=3;", *aname*, "    1 Atch
Information Management Sciences Section      AFSC
Form 6", *hsstr*, "
Information Processing Branch,GCR=4;", "
FRANK J. TOMAINI, Chief
Information Processing Branch
Information Processing Division";              1b13b1j3
FIND SF(*tempstr*) ^eptr1 SE(*tempstr*) ^eptr2; 1b13b1j4

```

Source Code for FORMATTER Subsystem

```
        cinssta (eptr, 0, septr1, septr2);           1b13b1j5
%reset globals%                                   1b13b1k
        prnum,L _ aname,L _ name2,L _ name3,L _ 0;  1b13b1k1
    END;                                           1b13b1l
    = backup; =cleanup;                             1b13b2
    ENDCASE;                                       1b13b3
    RETURN(TRUE);                                  1b13c
    END,                                           1b13d
FINISH                                           1b14
```

Source Code for FORMATTER Subsystem

(J33525) 22-SEP-75 14:41;;; Title: Author(s): Duane L. Stone/DLS;
Distribution: /JPC([INFO-ONLY]) FSL([INFO-ONLY]) JHB([
INFO-ONLY]) ; Sub-Collections: RADC; Clerk: DLS;

33525 Distribution

Joseph P. Cavano, Frank S. LaMonica, James H. Bair,

Re Usage type

Do you want the frequency of command use by individuals or by Bell as a whole?

1

RA3Y 22-SEP-75 15:38 33526

Re Usage type

(J33526) 22-SEP-75 15:38;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /GCE([ACTION]); Sub-Collections:
SRI-ARC; Clerk: RA3Y;

33526 Distribution
Gwen C. Edwards,

Summary Report of the September 12, 1975 MSGGROUP Meeting

21-SEP-75 2253-PDT STEFFERUD: Sept 12 Meet Report Draft
 Distribution: MSGGROUP
 Received at: 21-SEP-75 22:53:12-PDT

1

SUMMARY REPORT OF THE SEPTEMBER 12, 1975 MSGGROUP MEETING

1a

A. LOCATION AND ATTENDENCE

1b

The meeting was held at the offices of Scientific Applications, Inc. in Arlington (Rosslyn), Virginia on September 12, 1975.

1c

Attendees were:

Engelbart@OFFICE-1 Doug Engelbart
 SRI, 333 Ravenswood Ave.
 Menlo Park, CA 98625

1d

Farber@ISI David Farber
 University of California
 ICS, Irvine, CA 92664

1e

Jernigan@OFFICE-1 Mil Jernigan
 ACC, P.O. Box 174
 Annapolis Jct, MD 20701

1f

McLindone@ISI Connie McLindon
 ARPA, 1400 Wilson Blvd.
 Arlington, VA 22209

1g

MSB@ISI Michael Broos
 Project MAC-MIT, Room 20S
 545 Tech Square
 Cambridge, MA 02139

1h

Myer@BBNA Ted Myer
 Bolt Beranek & Newman, Inc.
 50 moulton Street
 Cambridge, MA 02138

1i

Stefferdud@ISI Einar Stefferud
 Network Management Associates
 3002 Midvale Ave, Suite 202
 West Los Angeles, CA 90034

1j

Uhlig@OFFICE-1 Ronald Uhlig
 HQ US Army Materiel Command
 ATTN: AMCMS-I
 5001 Eisenhower Ave.
 Alexandria, VA 22333

1k

Summary Report of the September 12, 1975 MSGGROUP Meeting

Vezza@MIT=DMS	Al Vezza Project MAC=MIT, RM 218 545 Tech Square Cambridge, MA 02139	11
vonGehren@OFFICE-1	Ed vonGehren HQ US Army Materiel Command ATTN: AMCMS-1 5001 Eisenhower Ave. Alexandria, VA 22333	1m
Walker@ISI	Steve Walker ARPA/IPT, 1400 Wilson Blvd. Arlington, VA 22209	1n

B. PURPOSE OF THE MEETING 1o

The purpose of the meeting was to hold informal face to face discussions among some of the participants in MsgGroup, at least those who could conveniently attend a meeting in conjunction with COMPCON'75. 1p

All discussions were informal, in keeping with the intention to simply explore ideas and get to know each other better after an extended period of SNDMSG Teleconference participation. 1q

The purpose was generally accomplished, with an improved mutual understanding of our various perspectives and definitions of terms. There was a great deal of cross fertilization, which in this case appears to have considerable net positive value. 1r

There were no official decisions, but a number of cooperative actions are to be taken as a result of the discussions. These are described in the following brief report of the subjects discussed. 1s

C. SUBJECTS OF DISCUSSION 1t

The following subjects were discussed, with discussion leaders indicated. 1u

It was decided at the meeting that each discussion leader will prepare a summary of their discussion subject for distribution to the MsgGroup. Thus, this report only identifies each subject with a brief discussion. 1v

1. MsgGroup issues matrix - Stefferud 1w

This discussion centered on the matrix proposed by Stefferud in

Summary Report of the September 12, 1975 MSGGROUP Meeting

MSGGROUP#137 (18-AUG-75), and it was agreed that the matrix did provide a general, though vague, framework for the subject of the meeting.

1x

The third dimension of Formality enjoyed most of the discussion. Formality appears to involve a complex mix of issues, including legal authentication and distinctions between command & control vs open dialogue.

1y

The most notable conclusion on this subject was that the message system must not prescribe nor proscribe any particular degree of formality of communications. It must accommodate and facilitate all levels and dimensions of formality in a single framework.

1z

2. MAILSYS Plans and Improvements - Ted Myers

1a@

This discussion centered on a presentation of the new design features of the next release of MAILSYS.

1aa

It was agreed that the MAILSYS documents handed out at the meeting should be distributed to MsgGroup for review and discussion. This has been done by sending MSGGROUP#165,166 (18-SEP-75) to MsgGroup@ISI.

1ab

The plans for MAILSYS are to embody the desirable features of MSG in the message accessing and reading facilities of MAILSYS. It was agreed that care should be exercised to avoid superficial copying of MSG features. MSG features need to be evaluated to identify those that the users find most valuable.

1ac

MsgGroup members are asked to review the MAILSYS documents and comment on them in terms of the desirability of the features and their implementation.

1ad

3. MSGIRS Concepts and Application - Vezza

1ae

This discussion grew out of a message from Al Vezza (MSGGROUP#27, 19-JUN-75) describing a Message System at MIT-DMS which builds a data base and provides access to messages through an Information Retrieval System, hence the name MSGIRS.

1af

MSGIRS offers facilities for message handling that none of the other systems offer to date (MSG, MAILSYS, ETC.). Specifically, it gives a user the ability to search for messages on the basis of the content of a variety of fields. The system uses file inversion techniques to provide rapid retrieval. Inversion takes a rather large amount of background processing, and this is normally done with a Daemon that runs during slack usage periods.

1ag

Summary Report of the September 12, 1975 MSGGROUP Meeting

It was agreed that we should have MSGIRS applied to the PROCEEDINGS,MSG files in [ISI]<MsgGroup>. MSGIRS is running now at ISIA and this application is to be accomplished in the near future.

1ah

Al Vezza will supply us with instructions for accessing MsgGroup messages via MSGIRS after the file has been processed into the Data Base.

1ai

This application promises to aid in processing of the <MsgGroup> files because it is easy in MSGIRS to modify messages to add KEY WORDS and MSGGROUP#, etc.

1aj

In addition, we hope to be able to construct some new daemons that will watch for requests for copies of MSGGROUP# messages and automatically forward the requested message via SNDMSG. This will solve a number of problems encountered with Teleconferences involving non-homogeneous hardware and software systems. (ie. TENEX and non-TENEX)

1ak

MSGIRS is based on the concept of "a message being a structured object consisting of a set of fields, with 'typed' data in the fields." It would appear that there is good reason for standardizing the fields and data types to facilitate message system implementation. Al Vezza plans to prepare a message on this topic for distribution to MsgGroup.

1al

4. NLS Concepts and its User Interface - Engelbart

1am

A brief discussion centered on the implementation structure of NLS with its Command Language, its Command Interpreter, and its basic processes. The interesting points brought out were that the NLS Front End might be applied to implement message systems with any processes behind the interface, and that this can be done with considerable ease using the design principles of the NLS interface, provided the basic processes are suitably modular.

1an

It would seem very clear that NLS embodies many of the concepts we are beginning to want for network message systems, but they are isolated to single HOSTs as now implemented. Also, it was not clear at the meeting that NLS is efficient enough of CPU and DISC to warrant its wide spread use in network message systems. The flexibility is clearly useful, but network application needs to be tested. The NSW Project is testing some of these concepts now.

1ao

The NLS interface to the ARPANET was an issue at various times. It seems that there is not any clearly specified message format for implementation of any given message system interface, which has led to NLS implementation of "non-standard" message fields.

Summary Report of the September 12, 1975 MSGGROUP Meeting

(eg. "Distribution:") This issue ties in with the "network transmission of 'typed' data fields in messages" raised by Al Vezza.

1ap

5. MSG/XED - Open Group Discussion - No Assigned Leader

1aq

This discussion resulted from the fact that MSG has become the standard of comparison for ARPANET Message Systems, all other efforts not withstanding,

1ar

Although it was not clearly articulated during the meeting, it was clear that MSG, with XED attached, should not be allowed to pass out of existence before installation of equivalent or better message handling systems. MSG and its predecessors (RD, BANANARD, ETC.) should be recognized as major influential developments in message systems. We must take advantage of the lessons these MSG systems teach us. Somehow, we must blend the best parts of MSG into the new systems that are being implemented.

1as

Given this situation, it is important that MsgGroup Members enter comments regarding the values of MSG features and problems found in using MSG so that the best features can be identified for inclusion in new implementations of message systems.

1at

These conclusions regarding MSG derive from observations that many users generally feel more comfortable with MSG for handling their Network Mail than with any of the other systems. This comfort seems to result from the way MSG provides "intuitive ways to do what users want done with their messages."

1au

It must be noted here that the MSG user interface departs from the traditional TENEX Command recognition and completion rules, and that it adopts the perspective of a user who "Talks to MSG as one would talk to a human assistant." The MSG Command names are intuitively similar to the way non-programmers think about messages in files in other environments, such as non-computerized offices. This appears to be an important aspect of the MSG design.

1av

6. MsgGroup Proceedings Processing - Stefferud

1aw

At the end of the days meeting, it was agreed that MSGIRS will be applied to [ISI]<MSGGROUP>PROCEEDINGS.MSG as an experiment. PROCEEDINGS.MSG is to be kept as it is now in addition to putting it into MSGIRS.

1ax

Also, other files, such as TRANSCRIPT.MSG and ADMINISTRATIVE.MSG are to be archived on a regular basis. One month appears to be a

Summary Report of the September 12, 1975 MSGGROUP Meeting

good period, with archiving to occur at the end of each month for the prior months messages. (eg, Archive Aug on SEP 30)

1ay

In addition to these procedures, we are to continue looking for ways to facilitate SNDMSG Teleconferences with new tools and ideas, such as the MSGIRS Daemon for answering requests for specific MSGGROUP# messages, to facilitate non-TENEX site participation.

1az

D. SUMMARY OF MEETING CONCLUSIONS

1b@

The sense of the meeting was that there is a great deal to be gained from cross polination among message system developers, and that such cross polination should be fostered in the future.

1ba

One way to do this is to make each of the various message systems available at each development site. This would include ISI, BBN, MIT-DMS, and OFFICE-1. It would be very nice if each of these sites had the full range of experimental systems available for testing and direct experimentation.

1bb

It was further concluded that MSG/XED are the defacto standards of comparison, and as such should be continued until replaced with equal or better message systems.

1bc

Finally, it was concluded that the meeting had been very valuable in bringing together a wide range of ideas so that they might begin to be melded into the message system of the future.

1bd

Summary Report of the September 12, 1975 MSGGROUP Meeting

(J33527) 22-SEP-75 15:59;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /SRI-ARC([ACTION]) ; Sub-Collections:
SRI-ARC; Clerk: RA3Y;

33527 Distribution

N. Dean Meyer, James E. (Jim) White, Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk H. Van Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson, Don I. Andrews, Susan K. Ocken, Raphaël Rom, David C. Smith, Mary Ann Kellan, Buddie J. Pine, Andy Poggio, David L. Retz, Laura J. Metzger, Carolyn J. Martin, Jan A. Cornish, Larry L. Garlick, Priscilla A. Wold, Pamela K. Allen, Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log Augmentation, Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter, Robert Louis Belleville, Rene C. Ochoa, Ann Weinberg, Adrian C. McGinnis, Robert S. Ratner, David S. Maynard, Robert N. Lieberman, Sandy L. Johnson, James H. Bair, Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M. Beck, Marcia L. Keeney, Elizabeth K. Michael, Jonathan B. Postel, Elizabeth J. Feinler, Kirk E. Kelley

Mailsys Summary

Mailsys, called XMail on ISI machines, is BBN's new mail handling subsystem.

Mailsys Summary

18-SEP-75 1801-EDT MOOERS at BBN-TENEXA: MSGGROUP# 166 MAILSYS SUMMARY

Distribution: MSGGROUP AT ISI
 Received at: 18-SEP-75 15:18:33-PDT

Message-ID: <[BBN-TENEXA]18-SEP-75 18:01:21-EDT,MOOERS>

Bolt Beranek and Newman Inc. -- D R A F T -- Not for Publication

MAILSYS SUMMARY
 9/18/75

This summary gives you the minimum information you need to process your mail and send simple messages. MAILSYS is much more versatile and powerful than can be indicated here. For more complete documentation, use the Tenex Exec to COPY <MAILSYS>NEWCONCEPTUALINTRO.TXT to LPT:. [The DESCRIBE feature is not yet implemented for the new MAILSYS commands,]

Mailsys command syntax is much like the Tenex Exec. You can type full commands Or a minimum number of characters to uniquely identify commands; the "Escape" key (ESC) triggers recognition and extension. The "Carriage Return" key (CR) terminates commands and starts execution. For quick info while typing in commands, type ?

Control-E aborts commands, Control-O stops printout; Control-A deletes a single character; Control-C returns you to the Exec (not recommended; use QUIT, instead). Type CONTINUE to Exec to get back to MAILSYS.

Mailsys command groups (1) process received messages, and (2) create and transmit out-going messages.

(1) MESSAGE-PROCESSING COMMANDS

When you enter MAILSYS, the system sets up a current MESSAGE-LIST consisting of all recent messages, i.e., all messages that have arrived since the last time you looked in the INBOX. To examine your RECENT mail, press the linefeed key (LF):

(LF) prints the first RECENT message.

Type (LF) again to print each subsequent RECENT message.

The message that is printed out with (LF) becomes the CURRENT MESSAGE, and is symbolized by the period "."

SHOW . (CR) prints the message number of the CURRENT MESSAGE.

Mailsys Summary

JUMP-TO <number> (CR) changes the message number of the CURRENT MESSAGE to <number>.
 (LF) changes CURRENT MESSAGE to the next message on MESSAGE-LIST and prints it on your terminal.
 ^ changes CURRENT MESSAGE to the preceding message on MESSAGE-LIST and prints it. 1m

You can dispose of the CURRENT MESSAGE or of all RECENT messages in a number of ways: 1n

PRINT . (CR) prints the CURRENT MESSAGE on your terminal.
 PRINT (CR) prints all messages on the current MESSAGE-LIST . 1o

LINEPRINT . (CR) prints the CURRENT MESSAGE on the lineprinter.
 LINEPRINT (CR) prints all messages on the current MESSAGE-LIST on the lineprinter. 1p

LONG MESSAGES OR OTHER PRINTOUT: Stop printout with Control-O (upper-case letter O). Then use LINEPRINT . (CR) to send the message Control-O leaves the system where it would have been if the printout had been completed. 1q

FILE . <file-name> (CR) adds the CURRENT MESSAGE to the specified TENEX file.
 FILE <file-name> (CR) files all messages on the current MESSAGE-LIST. 1r

DELETE . (CR) marks the CURRENT MESSAGE for deletion.
 DELETE (CR) deletes the current MESSAGE-LIST. Also D (CR) 1s

UNDELETE . (CR) removes the DELETE marking on the CURRENT MESSAGE.
 UNDELETE (CR) undeletes all the messages on the current MESSAGE-LIST. 1t

EXPUNGE (CR) physically removes all messages currently marked for deletion. (QUIT asks if it may expunge deleted messages.) 1u

Other useful commands: 1v

STATUS (CR) gives a quick check on the number of messages in your INBOX and their status.
 You can get the same result with SHOW INBOX (CR). 1w

SURVEY . (CR) prints a one-line summary of the CURRENT MESSAGE.
 SURVEY (CR) surveys all the messages on the current

Mailsys Summary

MESSAGE=LIST,
 SURVEY ALL (CR) surveys your INBOX, 1x

PRINT (CR), LINEPRINT (CR), FILE (CR), DELETE (CR), SURVEY (CR),
 etc., take <message-list> as an argument, e.g., PRINT
 <message-list> (CR), and default to the current MESSAGE=LIST,
 You can change the current MESSAGE=LIST with 1y

CONSIDER <message-list> (CR) 1z

This is the same as USE MESSAGE=LIST <message-list> (CR), 1a@

A message-list is an ordered set of message-groups, separated by
 commas: 1aa

<message-group>, <message-group>, ... , <message-group>
 A message-group may be: A literal message number or range of
 numbers:
 ex.: 3
 1-4 = 1,2,3,4
 9-5 = 9,8,7,6,5
 A special symbol for a message number:
 % = last message in INBOX
 . = the CURRENT MESSAGE
 A predefined, named list of messages: ALL or *, RECENT=MSGS,
 OLD=MSGS, NEW=MSGS, SEEN=MSGS, UNSEEN=MSGS, DELETED=MSGS,
 UNDELETED=MSGS
 A named list of messages defined by the user
 A message-group acted upon by a FILTER:
 <message-group>/<filter>
 Ex.: ALL/FROM JONES
 DELETED=MSGS/SUBJECT BOSTON
 23=%/UNSEEN 1ab

It is also possible to use one-time message-lists without
 changing the current MESSAGE=LIST,
 Ex.: PRINT 1,3,ALL/FROM JONES (CR)
 DELETE OLD=MSGS (CR) 1ac

The current MESSAGE=LIST and the CURRENT MESSAGE are part of the
 MAILSYS "ENVIRONMENT", which also includes other "OBJECTS", such
 as, any number of other, named MESSAGE=LISTS, a current FILTER,
 other FILTERS, a current TEMPLATE, other TEMPLATES, a current
 SURVEY-TEMPLATE, a current output STREAM, and a current SWITCHES
 that sets the default options for several commands in the system, 1ad

The current MESSAGE=LIST, FILTER, and TEMPLATE can be MODIFIED,
 with the MODIFY command), and REMEMBERED under a name during a
 working session. The SURVEY-TEMPLATE and the SWITCHES can be

Mailsys Summary

MODIFIED but not named. The STREAM can be changed (with the USE command), but cannot be named. 1ae

[Not yet implemented is a PROFILE feature which will allow named filters and templates, the SURVEY-TEMPLATE, and SWITCHES to be saved across sessions.] 1af

Normally, Mailsys uses the standard INBOX file, which is MESSAGE.TXT in your logged-in file directory. Incoming messages are always placed there by the system. To examine or process messages stored in other files, you must first change your current INBOX: 1ag

INPUT <file-name>(CR)
Ex.: INPUT SAVED,MESSAGES 1ah

You can get the same effect with USE INBOX <file-name> (CR), (CR). 1ai

(2) MESSAGE-CREATING COMMANDS 1aj

Commands TO, CC, BCC, FCC, SUBJECT, FROM, IN-REPLY-TO, REFERENCES, KEYWORDS, PRECEDENCE, MESSAGE-CLASS and SPECIAL-HANDLING create message header fields. One of the first four is needed to send a message; all others are optional. 1ak

TO, CC, and BCC address your message, take an <addressee list> with format as in SNDMSG: 1al

Ex.: TO: SMITH, JONES@BBND, DOE, GREEN@ISI(CR) 1am

Attention specs and distribution lists are also possible. DESCRIBE ADDRESS FIELDS for more detail. 1an

FCC <file-names> (CR) appends copies of message to designated Tenex files: 1ao

Ex.: FCC SAVED,MESSAGES, JOB,RECORD (CR) 1ap

Remaining header fields take arbitrary text: 1aq

Ex.: SUBJECT: This is arbitrary text, (CR) 1ar

Header fields can be continued past one line by preceding (CR) with comma: 1as

Ex.: KEYWORDS: ALPHA,(CR)
 BETA ,(CR)
 GAMMA (CR) 1at

Mailsys Summary

The command TEXT creates the text field of your message, takes arbitrary text up to <Control-Z>: 1au

TEXT: Now is the time
for all good men <Control-Z> 1av

Repeated use of any field command appends to the previous contents of the field (if any), 1aw

Other useful commands: 1ax

DISPLAY <field> or HEADER or ALL (CR) and
ERASE <field> or HEADER or ALL (CR) display or erase the specified field, all header fields, or all fields. Both default to ALL if nothing is specified. 1ay

Ex.: DISPLAY KEYWORDS (CR)
ERASE TEXT (CR) 1az

NETED <field> (CR), TECO <field> (CR) or XED <field> (CR) call your choice of text editor on the specified field. FORMAT <field> (CR) squares up text in the field. Type DESCRIBE FORMAT for conventions and subcommand options. NETED, TECO, XED and FORMAT default to TEXT if no field is specified. 1be

SEND (CR) queues your completed message for Mailer to transmit. SEND normally erases all message fields. Type DESCRIBE SEND for NO ERASE feature. 1ba

MAILER (CR) forces immediate transmission of all SENT messages. 1bb

SNDMSG (CR) invokes prompt-driven dialogue that elicits your input for the TO, CC, SUBJECT, and TEXT fields. You can return to command level to change or add to these fields before transmission. 1bc

FORWARD , (CR) and INCLUDE , (CR) copy the CURRENT MESSAGE into the TEXT field of the message being created (FORWARD sets up appropriate header fields); REPLY , (CR) sets up header fields for response to the CURRENT MESSAGE. Try them or use DESCRIBE for more detail. 1bd

MODIFY <command name> <switch settings> (CR) lets you change the default conditions for some MAILSYS commands. Type DESCRIBE SWITCHES for details. 1be

QUIT (CR) terminates MAILSYS session. 1bf

Mailsys Summary

(J33528) 22-SEP-75 16:02;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /SRI-ARC([INFO-ONLY]); Sub-Collections:
SRI-ARC; Clerk: RA3Y;

33528 Distribution

N. Dean Meyer, James E. (Jim) White, Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk H. Van Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson, Don I. Andrews, Susan K. Ocken, Raphael Rom, David C. Smith, Mary Ann Kellan, Buddie J. Pine, Andy Poggio, David L. Retz, Laura J. Metzger, Karolyn J. Martin, Jan A. Cornish, Larry L. Garlick, Priscilla A. Wold, Pamela K. Allen, Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log Augmentation, Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter, Robert Louis Belleville, Rene C. Ochoa, Ann Weinberg, Adrian C. McGinnis, Robert S. Ratner, David S. Maynard, Robert N. Lieberman, Sandy L. Johnson, James H. Bair, Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M. Beck, Marcia L. Keeney, Elizabeth K. Michael, Jonathan B. Postel, Elizabeth J. Feinler, Kirk E. Kelley

A conceptual introduction to Mailsys message-processing commands

Mailsys, known as XMail on USC-ISI machines, is BBN's new mail handling program.

A conceptual introduction to Mailsys message-processing commands

18-SEP-75 1804-EDT MOOERS at BBN-TENEXA; MSGGROUP# 165 A Conceptual Introduction to MAILSYS Message-processing commands

Distribution: MSGGROUP AT ISI

Received at: 18-SEP-75 15:18:12-PDT

Message-ID: <[BBN-TENEXA]18-SEP-75 18:04:23-EDT,MOOERS>

Bolt Beranek and Newman Inc, -- D R A F T -- Not for Publication

A CONCEPTUAL INTRODUCTION TO
MAILSYS MESSAGE-PROCESSING COMMANDS
9/18/75

I, MAILSYS TERMINOLOGY

A, OBJECTS IN THE MAILSYS ENVIRONMENT

CURRENT OBJECTS

If an OBJECT is defaulted, i.e., not specified by name, in a MAILSYS command, the system uses the current OBJECT,

- A.1. The current INBOX -- file containing messages.
- A.2. The current MESSAGE-LIST -- set of messages in INBOX.
- A.3. The CURRENT MESSAGE, symbolized by "." -- message just acted upon in some way. Initially set to "the one before the first" in MESSAGE-LIST.
- A.4. The current FILTER -- acts on a message-list; selects messages on the basis of information or status.
- A.5. The current TEMPLATE -- specifies parts of message to be output.
- A.6. The current SURVEY-TEMPLATE -- template used by SURVEY command.
- A.7. The current STREAM -- type of output: a printing device or file and a specification for separating messages.
- A.8. The current setting of the SWITCHES -- User's choices of switch settings for commands that have more than one default option.

PERMANENT OBJECTS

Named Message-lists: ALL, RECENT-MSGS, OLD-MSGS, NEW-MSGS, SEEN-MSGS, UNSEEN-MSGS, DELETED-MSGS, UNDELETED-MSGS

Named Filters: BLANK, SEEN, UNSEEN, DELETED, UNDELETED, TODAY, YESTERDAY

Named Templates: VERBATIM, ALL, BASIC-SURVEY, NULL

The SURVEY-TEMPLATE: The contents of the SURVEY-TEMPLATE can be changed, but only one SURVEY-TEMPLATE can exist.

A conceptual introduction to Mailsys message-processing commands

The SWITCHES: The settings of the switches can be changed, but only one group of settable SWITCHES can exist,	1j
OBJECTS CREATED AND STORED BY THE USER	1k
Named Message-Lists	
Named Filters	1l
Named Templates	
B. MESSAGE-HANDLING COMMANDS	1m
B.1. TRANSCRIBE: Copies messages to printing devices or files. (Special cases: READ, (LF), ^, PRINT, LINEPRINT, SURVEY.) Files are NOT MAILSYS-readable.	
B.2. FILE: Copies messages to MAILSYS-readable files.	
B.3. DELETE: Marks messages for deletion.	
B.4. UNDELETE: Removes DELETE markings.	
B.5. EXPUNGE: Physically removes DELETED messages from the files,	1n
C. OBJECT-HANDLING COMMANDS	1o
C.1. SHOW: Prints out any or all current objects, and any or all fields of the current message being built for sending. (Special case: STATUS = SHOW INBOX)	
C.2. USE: Creates a current object which is a copy of a literal value or a named object. USE takes current FILTER, INBOX, MESSAGE-LIST, STREAM, SURVEY-TEMPLATE, TEMPLATE or , (= CURRENT MESSAGE). (Special cases: INPUT, CONSIDER, JUMP-TO)	
MODIFY: Allows User to change the current object; calls a more or less elaborate editor, depending upon the object. MODIFY takes FILTER, INBOX, SURVEY-TEMPLATE, PROFILE [Not yet implemented,], SWITCHES, TEMPLATE.	
REMEMBER: Copies the current object into a named object. REMEMBER takes FILTER, MESSAGE-LIST, TEMPLATE.	
C.5. FORGET: Physically removes named object from the environment,	1p
D. THE USER'S PROFILE -- Characterization of the User Through Choice of Used-Defined Filters and Templates, the SURVEY-TEMPLATE, and Settings of Switches.	1q
D.1. RESET: Saves the PROFILE across sessions. [Not yet implemented,]	1r
II. THE MAILSYS COMMANDS	1s
A. CURRENT OBJECTS IN THE MAILSYS ENVIRONMENT	1t

A conceptual introduction to Mailsys message-processing commands

The MAILSYS current environment is composed of a number of OBJECTS. Only one of each of these objects can exist at any one time as the current OBJECT. There are eight current OBJECTS in all.

1u

A.1. The current INBOX, named "INBOX",

1v

An inbox is a file identified by a Tenex file name which contains MAILSYS-readable messages. Each message is identified by a message number.

1w

Initial setting: MESSAGE,TXT;1 in the User's directory,

1x

Any other file that contains MAILSYS-readable messages may be used as the INBOX.

1y

A.2. The current MESSAGE=LIST, named "MESSAGE=LIST",

1z

A message-list is an ordered set of message-groups, separated by commas:

1a@

<message-group>, <message-group>, ... , <message-group>

1aa

Initial setting: all RECENT messages in INBOX,

1ab

A message-group may be:

A literal message number or range of numbers:

Ex.: 3

1-4 = 1,2,3,4

9-5 = 9,8,7,6,5

A special symbol for a message number:

% = last message in INBOX

, = the CURRENT MESSAGE

A predefined, named list of messages:

ALL or * = all messages in INBOX

RECENT=MSGS = all messages that have arrived since you last looked at INBOX

OLD=MSGS = all messages that arrived before you last looked at INBOX

NEW=MSGS = messages that have arrive during this session, but that have not been looked at

SEEN=MSGS = all messages marked seen

UNSEEN=MSGS = all messages not marked seen

DELETED=MSGS = all messages marked deleted

UNDELETED=MSGS = all messages not marked deleted

A User-defined, named list of messages

A message-group acted upon by a filter:

<message-group>/<filter>

Ex.: ALL/FROM JONES

A conceptual introduction to Mailsys message-processing commands

DELETED=MSGs/SUBJECT BOSTON
23-%/UNSEEN

1ac

MESSAGE-LIST may set be to any combination of literal or named message-groups; it may be MODIFIED, named and REMEMBERED during a working session.

1ad

A.3. The CURRENT MESSAGE, symbolized by ".",

1ae

This is the message which has just been acted upon in some way.

1af

Initial setting: "the one before the first", that is, the position just before the first message on MESSAGE-LIST. This convention is adopted so that the command (LF) can be used to print out the first message.

1ag

(LF) changes the CURRENT MESSAGE to the next message on MESSAGE-LIST, and then prints the new CURRENT MESSAGE on the User's terminal.

^ changes the CURRENT MESSAGE to the previous message on MESSAGE-LIST, and then prints it.

1ah

The CURRENT MESSAGE can always be entered in a message-list symbolically as "."

1ai

SHOW . (CR) prints out the message number of the CURRENT MESSAGE on the User's terminal.

1aj

USE . <number> (CR) or JUMP-TO <number> (CR) changes the message number of the CURRENT MESSAGE to <number>.

1ak

A.4. The current FILTER, named "FILTER".

1al

A filter is a tool for selecting messages on the basis of their status or of the information they contain. It is always used in conjunction with a message-list.

1am

Initial setting: empty -- no current filter

1an

A filter may be:

A predefined, named filter:

BLANK -- passes any message

DELETED -- passes only messages marked deleted

UNDELETED -- passes only messages not marked deleted

SEEN -- passes only messages marked seen,

UNSEEN -- passes only messages not marked seen,

TODAY -- passes only messages received today,

YESTERDAY -- passes only messages received yesterday.

A conceptual introduction to Mailsys message-processing commands

A User-defined filter,

A one-time, throwaway literal filter:

BEFORE <date>

AFTER <date>

ONDATE <date>

where <date> may be entered as 4-JUL-76, 4 JUL 76, or 7/4/76, and the month may be abbreviated or spelled, upper- or lower-case,

<message header field> <text string>

where <text string> is a substring of the text of the header field.

Ex.: FROM SMITH

AFTER 12-MAR-75

SUBJECT PTERANODON

Note: SUBJECT PTER would also match.

1a0

The USE command sets FILTER to a copy of a named filter; it may be MODIFIED, and REMEMBERED under a User-assigned name during a working session. Named filters may be saved across sessions as part of the User's PROFILE.

1a1

A.5. The current TEMPLATE, named "TEMPLATE". A template is means of specifying the parts of the message that are to be output.

1a2

Initial setting: VERBATIM.

1a3

A template may be:

A predefined, named template:

VERBATIM -- includes all the message fields, the character count, and the message number; moves the message as a unit; cannot be MODIFIED.

ALL -- moves all the message, including all fields, the character count, and the message number. Rearranges the order of the fields if they are not in standard form. Can be MODIFIED to create a new template.

BASIC-SURVEY -- (used by SURVEY) moves a one-line summary of the message that includes message number, character count, date received, the FROM field, and as much of the SUBJECT field as will fit on the line. Can be MODIFIED to change the SURVEY-TEMPLATE.

NULL -- moves nothing. Can be MODIFIED to create a new template.

A User-defined, named TEMPLATE

1a5

The USE command sets TEMPLATE to a named template (e.g., BASIC-SURVEY or ALL); it may be MODIFIED, and REMEMBERED under a User-assigned name during a working session.

1a6

A conceptual introduction to Mailsys message-processing commands

A.6. The current SURVEY-TEMPLATE, named "SURVEY-TEMPLATE", is a special-purpose template used by the SURVEY command only. 1au

Initial setting: BASIC-SURVEY -- the message number, the character count, the date received, the FROM field, and as much of the SUBJECT field as will fit on one line. 1av

SURVEY-TEMPLATE may be set to a copy of a named template; it may be MODIFIED, but may not be named: its name is always SURVEY-TEMPLATE. Only one SURVEY-TEMPLATE can exist at a time. SURVEY-TEMPLATE may be saved across sessions as part of the User's PROFILE 1aw

A.7. The current STREAM, named "STREAM". 1ax

A stream defines the place and manner in which messages are output: 1ay

<stream> = <printing device or file> <separation> 1az

Initial setting: TTY: NOSEPARATE 1be

The <printing device or file> may be
 TTY: -- prints messages on the User's terminal,
 LPT: -- prints messages on the lineprinter,
 <file-name> -- copies messages into a file with the TENEX file designation <file name>. 1ba

The <separation> may be
 NOSEPARATE -- does not separate messages,
 SEPARATE -- inserts a formfeed after each message. 1bb

The USE command changes STREAM to a new literal value (e.g., to the lineprinter, or to a Tenex file); it may not be named. Only one STREAM can exist at a time. 1bc

A.8. The current SWITCHES or default settings, named "SWITCHES". 1bd

This feature allows the User to choose between the different default options associated with many of the MAILSYS commands. 1be

SWITCHES may be MODIFIED but may not be named. Only one SWITCHES can exist at one time. SWITCHES may be saved across sessions as part of the User's PROFILE. 1bf

B. COMMANDS THAT DO THINGS TO MESSAGES:
 TRANSCRIBE (Special Cases: READ, (LF), ^, PRINT, LINEPRINT, and SURVEY), FILE, DELETE, UNDELETE, EXPUNGE 1bg

A conceptual introduction to Mailsys message-processing commands

B.1. TRANSCRIBE -- The general command for processing messages, 1bh

TRANSCRIBE <message-list> <template> <stream> (CR) Also: T 1b1

TRANSCRIBE copies the messages listed in <message-list> according to <template> on the output <stream>. 1bj

The arguments default to the current value, so that: 1bk

TRANSCRIBE (CR) or T (CR) 1b1

means TRANSCRIBE MESSAGE=LIST according to TEMPLATE on STREAM. 1bm

Six special cases of TRANSCRIBE are given special names: READ, (LF), ^, PRINT, LIST and SURVEY. 1bn

READ <message-list> <template> (CR) Also: R
 = TRANSCRIBE <message-list> <template> <stream> (CR)
 prints out messages on <stream>.
 If <stream> is TTY:NOSEPARATE, The system pauses after each message. To READ the next message, type (LF). 1bo

(LF) cannot be mimicked directly by TRANSCRIBE. (LF) changes the CURRENT MESSAGE to the next message on MESSAGE=LIST and then prints out the new CURRENT MESSAGE on the User's terminal. 1bp

^ is the "reverse" of (LF). ^ changes the CURRENT MESSAGE to the preceding message on MESSAGE=LIST, and then prints it out.

PRINT <message-list> <template> (CR) Also: P
 = TRANSCRIBE <message-list> <template> TTY: NOSEPARATE (CR)
 prints out the messages on the User's terminal without pausing between messages. 1bg

LINEPRINT <message-list> <template> (CR) Also: L
 = TRANSCRIBE <message-list> <template> LPT: SEPARATE (CR)
 prints out the messages on the lineprinter. 1br

SURVEY <message-list> <printing device or file> (CR) Also: S
 = TRANSCRIBE <message-list> SURVEY-TEMPLATE <printing device or file> NOSEPARATE (CR)
 prints out messages according to SURVEY-TEMPLATE on the printing device or file. 1bs

B.2. FILE -- The command for filing messages. 1bt

FILE <message-list> <file-name> (CR) Also: F 1bu

A conceptual introduction to Mailsys message-processing commands

copies the messages listed in <message-list> into a file with the TENEX file designation indicated by <file-name>. 1bv

NOTE: FILE is the only command that produces files of MAILSYS-readable messages suitable for use as INBOXES. If a file-name is used as the <stream> in a TRANSCRIBE, READ, or SURVEY command, the resulting file will NOT be MAILSYS-readable, although it may be entered in a message field with one of the following commands: INCLUDE, ADD, Control-B, 1bw

B.3. DELETE -- The command for marking message for deletion. 1bx

DELETE <message-list> (CR) 1by

DELETE marks the messages on <message-list> for deletion. DELETED messages remain in the file, and may be selected with the DELETED-MSGs filter or TRANSCRIBED with the DELETED template. 1bz

B.4. UNDELETE -- The command for removing DELETE markings. 1c@

UNDELETE <message-list> (CR) 1ca

UNDELETE removes the DELETED markings from message that have been marked for deletion. There is no difference between messages that have been DELETED and then UNDELETED and messages that have never been DELETED. Messages that do not have DELETE marking may be selected with the UNDELETED-MSGs filter or TRANSCRIBED with the UNDELETED template. 1cb

B.5. EXPUNGE -- The Command that Removes Messages from the INBOX 1cc

EXPUNGE (CR) 1cd

EXPUNGE physically removes all DELETED messages from the INBOX so that they cannot be UNDELETED. The remaining messages are assigned new message numbers. The commands INPUT and QUIT ask the User: "May I clean up and renumber the INBOX?" If the User types (CR), the DELETED messages are EXPUNGED. If the User types, N(CR), the DELETED messages are not EXPUNGED. 1ce

C. COMMANDS THAT DO THINGS TO OBJECTS:
 SHOW (Special Case: STATUS),
 USE (Special Cases: INPUT, CONSIDER, JUMP-TO, and ^),
 MODIFY, REMEMBER, and FORGET 1cf

1. SHOW -- The command for printing out OBJECTS on the Users's terminal. 1cg

A conceptual introduction to Mailsys message-processing commands

SHOW <object> <name> (CR) 1ch

where <object> may be 1ci

INBOX -- does not take name; prints out the number of messages, the file-name of INBOX, the number of messages unseen and the number deleted,
 Ex.: 5 messages in file <JONES>MESSAGE.TXT;1 ; 1 unseen ; 2 deleted

MESSAGE-LIST -- takes name; prints out message numbers of the named message-list, with an indication of the CURRENT MESSAGE
 Ex.: SHOW ALL (CR)
 1-23, 24 (=.), 25-28

, -- does not take name; prints out message-number of the CURRENT MESSAGE

FILTER -- takes name; prints out contents of named filter

TEMPLATE -- takes name; prints out contents of named template

SURVEY-TEMPLATE -- does not take name; prints out contents of SURVEY-TEMPLATE

STREAM -- does not take name; prints out contents of STREAM

SWITCHES -- does not take name; prints out list of command names, attributes and attribute values

FIELD -- takes field name; prints out contents of current message being built for sending

ALL -- does not take name; prints out values of all objects 1cj

A special case of SHOW is:
 STATUS (CR) = SHOW INBOX (CR) 1ck

2. USE -- The command for changing the current OBJECTS in the ENVIRONMENT. 1cl

USE <object> <literal value or name> (CR) 1cm

where <object> may be

INBOX -- takes a <file-name> only

MESSAGE-LIST -- takes literal value or named list with FILTERS, in any combination

FILTER -- takes literal value or named filter

TEMPLATE -- takes named template only

STREAM -- takes literal value only

, -- takes a single message number

SURVEY-TEMPLATE -- takes named template only 1cn

USE creates a current OBJECT which is a copy of the literal value or named object. 1co

There are three special cases of USE: 1cp

A conceptual introduction to Mailsys message-processing commands

```

INPUT <file-name> (CR)                                Also: I
  = USE INBOX <file-name> (CR)
  changes the current INBOX                            1cq

CONSIDER <message-list> (CR)                          Also: C
  = USE MESSAGE-LIST <message-list> (CR)
  changes the current MESSAGE-LIST                    1cr

JUMP-TO <message-number> (CR)                         Also: J
  = USE , <message-number> (CR)                      1cs

C.3.  MODIFY -- The command for modifying OBJECTS.    1ct
      >MODIFY <object> <name> (CR)                    1cu

where <object> may be
MESSAGE-LIST -- takes named message-list
FILTER -- takes named filter
TEMPLATE -- takes named template
SURVEY-TEMPLATE -- does not take name
SWITCHES - does not take name                        1cv

Each OBJECT causes MODIFY to call an appropriate editor. The
system automatically goes into a subcommand mode, which is
signalled to the User by the subcommand prompt >> ,    1cw

The User can then edit a temporary copy of the OBJECT named
<name>. The subcommand options may be used any number of times
in any order. Typing (CR) to the >> prompt returns the User to
command level. The modified temporary object becomes the current
OBJECT,                                                1cx

C.3.1 THE MESSAGE-LIST EDITOR                        1cy
      >MODIFY MESSAGE-LIST <message-list:1> (CR)    1cz

where <message-list:1> may be any predefined or User-defined,
      named message-list,                            1de

Subcommand options:
>>DISPLAY -- prints the contents of the temporary
      message-list on the User's terminal; can be used at any
      time to check the temporary message-list as it is being
      MODIFIED.
>>SORT (by) <parameter> (CR) -- rearranges the messages on
      <message-list:1> according to <parameter> = Msg-number,
      Length (Character count), Date, Rcvd Date, From-Field,
      Message-ID, or Subject                          1da

```

A conceptual introduction to Mailsys message-processing commands

- >>ADD (messages) <message-list:2> (CR) -- creates a new temporary message-list consisting of the messages that appear on <message-list:1> plus the messages on <message-list:2>, with duplicate messages removed. This is the union of <message-list:1> and <message-list:2>, 1db
- >>REMOVE (messages) <message-list:2> (CR) -- creates a new temporary message-list by removing the messages on <message-list:2> from <message-list:1>. 1dc
- >>INTERSECT-WITH (messages) <message-list:2> (CR) -- creates a new temporary message-list which contains only the messages that appear on both <message-list:1> and <message-list:2>. This is the intersection of <message-list:1> and <message-list:2>. 1dd
- >>(CR) -- ends the subcommands and causes the temporary message-list to become the new current MESSAGE-LIST. 1de

C.3.2 THE TEMPLATE EDITOR 1df

- >MODIFY TEMPLATE <template> (CR) 1dg

where <template> may be any predefined or User-defined, named template. 1dh

or 1di

- >MODIFY SURVEY-TEMPLATE (CR) 1dj

Subcommand options: 1dk

- >>DISPLAY (line number) <line number> (CR)-- prints the contents of the line numbered <line number> on the User's terminal. If <line number> is omitted, defaults to the contents of the entire template.
- >>INSERT (on line number) <line number> (before item no. or items) <item number> <quoted text string> (CR) --
- >>DELETE (line number) <line number> (CR)
- >>PICTURE (line number) <line number> (CR)
- >>(CR) 1dl

C.3.3 THE FILTER EDITOR 1dm

- >MODIFY FILTER <filter> (CR) 1dn

where <filter> may be any predefined or Used-defined, named filter.

- >>DISPLAY (CR)

A conceptual introduction to MailSYS message-processing commands

```

>>REQUIRE <attribute> (CR)
>>REJECT <attribute> (CR)
>>IGNORE <attribute> (CR)
>>AFTER <date> (CR)
>>BEFORE <date> (CR)
>>ON <date> (CR)
>>(CR)

```

1do

C.3.4 THE SWITCHES EDITOR

1dp

```

>><command name> (value is) <switch setting> (CR)

```

1dq

where <command name> and <switch setting> may be determined by using ? at the appropriate places.

1dr

```

MODIFY SWITCHES (CR) >>DISPLAY (CR)

```

1ds

causes MAILSYS to printout a list of the settable commands, the attributes that can be set, and the current values of the settings.

1dt

C.4. REMEMBER == The command for naming OBJECTS within a working session.

1du

```

REMEMBER <object> (and name it) <name> (CR)

```

1dv

where <object> may be

```

MESSAGE-LIST
FILTER
TEMPLATE

```

1dw

and <name> is any string of characters that does not contain a <separator> (= space (SP) or escape (ESC)) or a <terminator> (= carriage return (CR) or linefeed (LF)).

1dx

REMEMBER copies the current MESSAGE-LIST, FILTER or TEMPLATE into a named object which persists for the duration of the current session. See PROFILE for instructions on saving filters or templates across sessions.

1dy

C.5. FORGET == The command for getting rid of OBJECTS.

1dz

```

FORGET <object> <name> (CR)

```

1e@

where <object> and <name> are the same as for REMEMBER.

1ea

FORGET physically removes the named object from the MAILSYS environment.

1eb

A conceptual introduction to Mailsys message-processing commands

D. PROFILE -- [Not yet implemented.]

lec

RA3Y 22-SEP-75 16:04 33529

A conceptual introduction to Mailsys message-processing commands

(J33529) 22-SEP-75 16:04;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /SRI-ARC([INFO-ONLY]); Sub-Collections:
SRI-ARC; Clerk: RA3Y;

33529 Distribution

N. Dean Meyer, James E. (Jim) White, Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk H. Van Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson, Don I. Andrews, Susan K. Ocken, Raphael Rom, David C. Smith, Mary Ann Kellan, Buddie J. Pine, Andy Poggio, David L. Retz, Laura J. Metzger, Karolyn J. Martin, Jan A. Cornish, Larry L. Garlick, Priscilla A. Wold, Pamela K. Allen, Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log Augmentation, Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter, Robert Louis Belleville, Rene C. Ochoa, Ann Weinberg, Adrian C. McGinnis, Robert S. Ratner, David S. Maynard, Robert N. Lieberman, Sandy L. Johnson, James H. Bair, Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M. Beck, Marcia L. Keeney, Elizabeth K. Michael, Jonathan B. Postel, Elizabeth J. Feinler, Kirk E. Kelley

Where to Find FORMATTER

This describes how to obtain the Subsystem FORMATTER, described in
(33523,)

Where to Find FORMATTER

For those of you who have used, or may in the future, the FORMATTER Subsystem....it is now located in the RADC directory. The RADC directory will be used in the future to store subsystems and special purpose programs peculiar to RADC. To obtain FORMATTER, type:

1

Execute Programs Load Program RADC,FORMAT...(type capital letters only)

1a

Then Execute or Goto Formatter to actually use it.

1a1

If you wish to have it loaded as one of the standard subsystem available as you enter NLS, then type:

1b

Execute Useroptions Include Program RADC,FORMAT

1b1

From then on the FORMATTER Subsystem will be loaded everytime you enter NLS. This eliminates the need to explicitly load it, as described above, when you want to use it.

1b1a

Where to Find FORMATTER

(J33530) 23-SEP-75 06:03;;; Title: Author(s): Duane L. Stone/DLS;
Distribution: /RADC([INFO-ONLY]) JHB([INFO-ONLY]);
Sub-Collections: RADC; Clerk: DLS;

33530 Distribution

Rocco F. Iuorno, Thomas J. Bucciero, Roger B. Panara, John L. McNamara, Joseph P. Cavano, Duane L. Stone, Marcelle D. Petell, Thomas F. Lawrence, James H. Bair, Wolf-Hasso Kaubisch, Kim Cynthia Carter, Samuel L. Ruple, Stephen P. Sutkowski, Richard Calicchia, William W. Patterson, Francis J. Hilbing, Robert K. Walker, Frank P. Sliwa, Joe F. Femia, Roger W. Weber, Melville J. Draper, Robert D. Krutz, James W. Hyde, David T. Craig, Fred N. Dimaggio, Robert E. Doane, Richard Nelson, William F. Stinson, Daniel R. Loreto, John B. McLean, Murray L. Kesselman, Edward F. LaForce, Agatha C. Deconde, Alan R. Barnum, Larry M. Lombardo, Roberta J. Carrier, Richard H. Thayer, Frank J. Tomaini, Mike A. Wingfield, Edmund J. Kennedy, Raymond A. Liuzzi, Donald VanAlstine, Deane F. Bergstrom, Frank S. LaMonica, William E. Rzepka

Trip Report, Visit to SRI, 15-19 Aug 75

NSW

1

Bill Carlson (ARPA) directed SRI to stop work on NSW protocols in Aug. His feeling was that the protocols were aimed at a more general problem than NSW and seemed to be too elaborate. As it became apparent that a disc would have to be ordered for the PDP-11 at Gunter and Gunter (Col. Baggiano) made strong statements about not ordering any more equipment, a decision was made to stop SRI's work and give MCA the protocol job.

1a

There were several negative aspects about this decision

1a1

The Steering committee should have been consulted on this decision, but only certain members were even aware it was happening.

1a1a

The DPS protocols developed by SRI were WORKING on the 10X, allowing the NLS frontend and backend to communicate.

1a1b

The MCA implementation appears that it will require coding for each tool/host pair. Whereas the SRI approach would only have to be done once for each host?

1a1c

The purpose of the meeting was to discuss SRI's role in the immediate future and in follow-on efforts for participation in the NSW project, in light of the cancellation of the protocol work.

1b

Bill Carlson feels that he cannot afford to commit more than \$200K this fiscal year and roughly the same next fiscal year to support SRI. He feels that ARPA support should be concentrated on the frontend work. It will be up to RADC, AFSDC and other sites interested in NLS to support needed work in the output processor, terminal and journal areas.

1c

The effect on the contract is to lengthen it from 9 months to 12 months, and delete SOW requirements for delivery of Protocol software. The wording of the SOW was revised jointly by Carlson, Watson, Postel and myself. The revised SOW is contained in (33501,).

1c1

This means that SRI will have to start tapering off soon, unless they can obtain support elsewhere. This seems a real possibility, but will require the divergence of some of their key people to proposal writing, etc.

1c2

ARPA (Carlson) wants to wind up its participation in the NSW development by the end of FY-77. This means that RADC will play an increasing role in the development, application (and consequently funding) of NSW. This seems reasonable thing to

Trip Report, Visit to SRI, 15-19 Aug 75

expect, since RADC should act as a middle man between basic research and application.

1d

WUS

2

The Workshop Utility Service contract with SRI was also discussed. SRI is still negotiating with BBNB for time on their machine to support the three additional slots (IBM, DMA, DSC). Another alternative being considered is to add core to the TYMSHARE machine, based on the success of the experimental effort at ISIC using 512K core. The addition of another 256K core would cost the equivalent of 3-4 slots, but is estimated to allow the support of 8-10 additional users. It therefore seems like the most economic way to go, but there is some question on the delivery time for more core, ie it may take too long to get it.

2a

SRI has received orders for terminal equipment for delivery to RADC and NAVCOSSACT. The others in the procurement mill, see (stone,wus,term:) have not reached SRI yet. They are willing to start shipment as soon as they get the request for quote from the government. Sufficient line processors are on hand. Delivery of Data medias only takes a month. Tektronix equipment takes more like 90 days.

2b

Martin Hardy is planning on installing 6 lineprocessors and Data Medias at RADC on the week of Oct 6th. Either that week, or the week after the KWAC meeting in Boston (two weeks later), a trainer will be available to instruct RADC users.

2b1

Alternatives for structuring the follow-on WUS contract were discussed. Unbundling of training, option for purchase of TYMSHARE machine at the end of the contract, L-10 training, pricing based on use, fixed price, etc. I will explore some of these options with our procurement to see which are possible, difficult etc.

2c

SRI applications group has not taken a position on whether or not they will support NLS running in the NSW environment. This could mean preparing documentation, training, etc for two versions of NLS; depending on when and if NLS 8.5 with graphics is brought up as the standard NLS running at Office-1.

2d

My feeling is that there should be no functional difference (from the user's viewpoint) between the service versions of NLS on Office-1 and NLS within NSW. The implementation, particularly the way the frontend talks with the back end may be different, but most training, documentation, etc should be able to be used on both types of NLS. There will, of course, be an experimental version of NLS...probably within the NSW

Trip Report, Visit to SRI, 15-19 Aug 75

environment for the next couple of years. The transition point between versions of NLS needs to be made more explicit, and criteria has to be established as to when a version is "completed", ie reaches the point where it is solid enough to be offered as a service and maintained by a minimal applications staff.

2d1

Other

2e

Doug Englebart is interested in setting up a meeting in the near future between key people in NSA, AMC WWMCCS and myself, to discuss the possibility of jointly supporting the development of a framework (Network Operating System + protocols) that will be more general in nature than the ones now being developed by MCA. This could take the form of continuation of the DPS work.

2e1

Dirk Vannouhuys gave me a draft proposal that will soon be circulated among the KWAC, to see if there is an interest in supporting a Document Production and Control System (DPCS) user group. The basic idea is that interested users would pool their resources (on the order of 10K each) to support someone at ARC who would be a focal point for development work and dissemination of information in the DPCS domain.

2e2

I received a number of think pieces and draft ideas for NLS development work, that SRI believes need to be worked on to bring NLS closer to a finished product. The principle areas are the Journal, Output processor and terminal configurations.

2e3

Dick Watson will be making a tour of the East coast in the next couple of weeks, to see where support might be forth-coming.

2e4

Trip Report, Visit to SRI, 15-19 Aug 75

(J33531) 23-SEP-75 06:12;;; Title: Author(s): Duane L. Stone/DLS;
Distribution: /JLM([INFO-ONLY]) FJT([INFO-ONLY]) RDK([INFO-ONLY]) ARB([INFO-ONLY]) FJH([INFO-ONLY]) MAW([INFO-ONLY]) ; Sub-Collections: RADC; Clerk: DLS;

33531 Distribution

John L. McNamara, Frank J. Tomaini, Robert D. Krutz, Alan R. Barnum,
Francis J. Hilbing, Mike A. Wingfield,

Re: Library space, (26522,) Jake

Jake, Something does need to be done about the shelves in the Conference rm, however, I don't want to store our documentation there. It is available to all, all the time outside J2096. Perhaps the excess Journal hardcopy (the oldest stuff) could be moved to the Parsely Rm where the bare shelves are unattractive. Jim

1

Re: Library space, (26522,) Jake

(J33532) 23-SEP-75 09:40;;; Title: Author(s): James H. Bair/JHB;
Distribution: /SRI-ARC([INFO-ONLY]); Sub-Collections: SRI-ARC;
Clerk: JHB;

33532 Distribution

Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk H. Van Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson, Don I. Andrews, Susan K. Ocken, Raphael Rom, David C. Smith, Buddie J. Pine, Andy Poggio, David L. Retz, Laura J. Metzger, Karolyn J. Martin, Jan A. Cornish, Larry L. Garlick, Priscilla A. Wold, Pamela K. Allen, Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log Augmentation, Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter, Robert Louis Belleville, Ann Weinberg, Adrian C. McGinnis, Robert S. Ratner, David S. Maynard, Robert N. Lieberman, Sandy L. Johnson, James H. Bair, Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M. Beck, Marcia L. Keeney, Elizabeth K. Michael, Jonathan B. Postel, Elizabeth J. Feinler, Kirk E. Kelley, N. Dean Meyer, James E. (Jim) White

Recovering the Hudson Terminals

If Hudson is not coming back, we should take steps to recover the display and TI 735 terminals we gave Rudy and are paying for ourselves. Also Rudy may want to sell his line processor. Since the display and 735 are expensive, we should recover them as soon as possible. What steps shall we take?

1

Recovering the Hudson Terminals

(J33533) 23-SEP-75 10:27;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /JCN([ACTION]) BJP([ACTION]) MEH([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: RA3Y;

33533 Distribution

James C. Norton, Buddie J. Pine, Martin E. Hardy,

What's Mits? cf (26528, 1j8)

Robert: Would be handy to find out what the "MITS" system is that beat us out with Major Borel -- cf your contract report on talk with Mel Draper, on 16 Sept 75 (HJOURNAL, 26528, 1j8). Would you look for a chance to learn, possibly get some documentation? Thanks, Doug

1

What's Mits? cf (26528, 1j8)

(J33534) 23-SEP-75 10:48;;; Title: Author(s): Douglas C.
Engelbart/DCE; Distribution: /RLL([ACTION]) JCN([INFO-ONLY])
RA3Y([INFO-ONLY] Ray: Like some grist for the C-7 work?) ;
Sub-Collections: SRI-ARC; Clerk: DCE;

33534 Distribution

Robert N. Lieberman, James C. Norton, Raymond R. Panko,

NMSG Manual

This is John Vittal's manual for NMSG, an experimental successor to MSG. It is currently available only at USC, but it may become more widely available later. This is a fairly rough manual, designed to serve as temporary documentation.

MSG MANUAL

John Vittal
USC Information Sciences Institute

MSG is a program for reading, writing, and subsectioning files which have a message file format. It is very simple and straightforward to use. Commands are initiated by typing one character, which causes the program to type out the rest of the command name and wait for input from you.

Before the commands are described, there are a few general statements about how MSG works and some conventions used in describing the commands that you should know about. The prompt characters letting you know that MSG is waiting for a command character to be typed are "<-". When MSG is started up (by typing MSG<return> to the EXEC) it will first try to read your MESSAGE.TXT;1 file in your directory. If this file does not exist MSG will say so. If you were not connected to your login directory, MSG will try to find a MESSAGE.TXT;1 there. If that also fails, it will say so and wait for a command to be typed. If you have a MESSAGE.TXT;1, it will scan it and type out the header information (i.e. the date, from, and subject fields) for each message since the file was last read, preceded by a message number sequentially assigned by MSG. These message numbers are used in association with the various commands.

However, if you started MSG by typing MSG<space> to the EXEC, it will ask you for a file to be read. Typing an escape as the first character will cause MESSAGE.TXT;1 to be typed out, and confirmation requested from the user to ensure that that was what was intended. Once a file name has been specified and positively acknowledged, then the same information as described in the previous paragraph will be output to your terminal.

When reading a message file in MSG, either when starting up MSG or with the Read command described below, the file must be in the so-called message file format. If MSG recognizes that the file does NOT conform to this format, you will be told so. However, you will be given the opportunity to keep everything that has been read so far, but NOT overwrite the 'bad' file. These two exceptional circumstances and some suggestions for getting around them are described at the end of this manual.

The following conventions and symbols are used in the command descriptions below. There are only five types of input MSG expects:

- (1) a MSG command (or sub-command) character
- (2) a message sequence specification
- (3) a TENEX file name

NMSG Manual

- (4) a confirmation character
- (5) a local user name or remote site name

To abort output to the terminal type ^O (control-O). If MSG does not understand your input, it will return to command input mode, or reprompt you. The following are symbols and their associated meanings used in the command descriptions:

<FILE-NAME>

Stands for any TENEX file descriptor, including TTY: or LPT:. If you are requested to input a file name, the appropriate TENEX confirm will be given (e.g. [Old version]).

<MSG-SEQUENCE>

This input is prompted by the string (message sequence) in verbose typeout mode. A sequence of message numbers has the following format.

- (1) Any single message number.
 - (2) Any two numbers separated by ">" or ":". This means message numbers delimited by the two outside numbers (e.g. 2>5 means messages 2,3,4, and 5 in that order). NOTE: if the first number is greater than the second number, it means the sequence in reverse order (e.g. 5>2 means messages 5,4,3, and 2).
 - (3) A pair of numbers separated by "-". This is so that the standard interpretation of the string "21-4" (that is not "21-24") means message numbers 21, 22, 23, and 24. Using this interpretation, the string and "24-1" is an error.
 - (4) Any sequence of the previous three types separated by commas. This is the way to group several non-adjacent messages together. For example: 1,3,5:7,10 means messages 1 and 3 and 5 through 7 and 10.
- <MSG-SEQUENCE> of the types described above are ALWAYS terminated by <return>.

(5) However, there are special types of message sequences. All are determined by the first character that you type in the <MSG-SEQUENCE> stream. The following are the twelve possibilities:

- a. <escape> is typed, which causes the current message number to be echoed to you and the relevant process performed on that message only.
- b. <control-I> is typed, which causes the previous completely specified <MSG-SEQUENCE> to be echoed and processing performed on that message stream.
- c. R which stands for "Recent messages" only.
- d. O standing for "Old messages" only.
- e. A standing for "All messages" and which is equivalent to 1:(last message number).
- f. D standing for "Deleted messages". This is valid ONLY in the context of the Headers, Undelete, and Delete commands. Everywhere else, the headers of the deleted messages will be printed. Of course, you can delete the

- typeout of those headers by typing control-0.
- g. U standing for "Undeleted messages".
- h. I standing for messages in inverse order. This is the opposite of the A (for all messages) sub-command.
- i. S for "Subject field search for string" which asks you to provide a string which will be used as a mask match on the subject field of the message headers.
- j. F for "From field search for string" which is like S but searches the Author field of the message headers instead. NOTE: the header command prints the initial part of the To: line of the message (if it exists) is the message was sent by the login-directory. Therefore, to search for messages sent by yourself, specify the string "To:" rather than the login directory name.
- k. E standing for "Examined messages", i.e. all messages which have been completely typed (with the T command) or listed (with the L command).
- l. N standing for "Not examined messages", which is the opposite of the E sub-command.

10

Types (i) and (j) require you to type a string terminated by <return>. Typing just a <return> (i.e. the null string) means that searching is not to be performed. Otherwise, the search will be performed on the string typed up to (not including) the <return>. The string you type must be an exact match to some substring of the appropriate field, but all alphabetic characters are treated as being upper case. (Note: carriage-returns in the subject field of the header listing are ignored.)

(6) If you type comma or "M" as the first character of the message sequence that you are specifying, you will be able to specify more than one of the options drawn from the first five items mentioned here. You will then be entered into a sub-command mode. Any of the standard message sequences are acceptable as input. To terminate the specification of the list of message sequences, just type a carriage return in response to the prompt. If you wish to abort the acquisition at any time, type "Q" (for Quit) or control-N (^N). To abort the acquisition of a single message sequence (like 3:14), type rubout. Typing rubout at the sub-command level (i.e. at the prompt without typing anything first) will have the same effect as typing control-N.

10a

The default message sequence is 'All messages'. Any message sequence specified causes an intersection to be taken between that single message sequence (like 'Examined'), and the previous total. For example, the sequence:

11

```
<- Headers ,
<<- Examined
```

```
<<- From string: VITTAL
<<-
```

11a

would cause only the headers corresponding to messages from VITTAL which have already been typed to get listed on your terminal.

12

If you just want to add a message sequence to the list, preface the actual message sequence with a "P" (for Plus) or "+". If you want to just subtract a message sequence from the list, preface the actual message sequence with an "M" (for Minus) or "-". For example,

13

```
<- Headers Multiple message sequences
<<- Examined
<<- From string: VITTAL
<<- Plus: Subject string: MSG
<<- Minus: Deleted
<<-
```

13a

will list the headers for all undeleted messages about MSG or which are examined messages from VITTAL. No further associations between msg-sequence specifications are currently allowed.

14

In the command format below, everything that the program types will be lower case and everything you type will be in UPPER CASE. This is not the case when using MSG, but is used here for clarity.

15

MSG COMMANDS

15a

Commands to Manipulate Message Headers

15b

```
<- Headers (message sequence) <MSG-SEQUENCE>
```

The headers for messages will be typed out for those messages defined by the message sequence typed. Headers corresponding to deleted messages have an asterisk printed before the header for that particular message. The headers for recent messages are preceded by a plus sign (+); messages which have not yet been typed are preceded by a minus sign (-), and deleted messages are preceded by an asterisk (*). If the message was sent by the user of the login directory, the initial part of the To: field of the message will be printed in the author field of the header, if the To: field exists in the message. In order to get the length of the message typed out along with the header, use the I command (which stands for Inclusion of length in header).

15c

```
<- Delete (message sequence) <MSG-SEQUENCE>
```

This command will indicate (by a preceding asterisk) in the header information for the messages specified by <MSG-SEQUENCE>

that those message are deleted. NOTE: This command marks each message in the actual message file indicating that it is deleted. If you reread the file for some reason, the messages will still be marked (and treated) as deleted (but not expunged). This command does however effect message numbers specified in later commands in the following way. If you have deleted message number 5 and then try to "Type" or "Put" message number 5 either directly or implied by the use of the ":" option, the deleted messages will NOT be included.

15d

<- Undelete (message sequence) <MSG-SEQUENCE>

Of course! If you can delete a message, you certainly ought to be able to undelete it. This command undoes the action of the Delete command for the messages specified by this <MSG-SEQUENCE>.

15e

Commands to See and Move Messages

15e1

<- Type (message sequence) <MSG-SEQUENCE>

This command will type on your terminal the messages specified by <MSG-SEQUENCE>. All messages which are completely typed are treated as having been 'examined'.

15f

<- Put (message sequence) <MSG-SEQUENCE>

into file name: <FILE-NAME>

This command will put the messages specified by <MSG-SEQUENCE> into the file specified by <FILE-NAME>. If the file does not exist, it will create that file and write the messages into it. If the file already exists, it will append the messages to the messages already in the file. This command is useful if you want to keep separate files containing messages concerning different topics.

15g

<- Move (message sequence) <MSG-SEQUENCE>

into file name: <FILE-NAME>

This command is a convenient combination of the Put and Delete commands. It will first put the messages into the file specified and then delete them from the header information. If any of the messages are already deleted, you will be informed.

15h

<- List (message sequence) <MSG-SEQUENCE>

on file: <FILE-NAME>

Lists all the specified messages on the file specified. All messages specified by the <MSG-SEQUENCE> are treated as having been examined (typed). If you are listing more than one message, there is a preface page with the headers for those messages, and you will be asked if you want each message on a separate page. All listed messages are marked as being 'examined'. The intention of this command is to allow a user to obtain a reasonable hard copy listing of some messages. (Note: the

preface page of headers will have the length of each message included depending on the setting by the I(nclusion of length in header) command.)

15i

Commands to Update Your Message Files

15i1

<- Overwrite old file <FILE-NAME> [confirm]

This command will overwrite the current file (specified by <FILE-NAME>) reflecting the fact that you have deleted messages. That is, if you delete message 2 and then "overwrite" your file, message 2 will disappear from that file. It also rereads your file, renumbering your messages. You are warned if any unexamined messages (which are also not deleted) exist in the file that you are overwriting.

15i2

<- Quit [confirm]

This command returns you to the TENEX EXEC without rewriting any file (almost equivalent to typing control-C). You are warned if any unexamined messages (which are also not deleted) exist in the current message file.

15i3

<- Exit and update old file <FILE-NAME> [confirm]

This command is another way to Overwrite your old message file, but instead of rereading the file it returns you to the TENEX EXEC. This is equivalent to doing an overwrite followed by a Quit, but without the overhead of rereading the file. You are warned if any unexamined messages (which are also not deleted) exist in the file that you are overwriting.

15i4

<- Write file <FILE-NAME> sorted by message arrival time

This is similar in nature to the Overwrite command, except that the messages are sorted into ascending sequence by their arrival time before the overwriting is attempted. The file is then rescanned. You are warned if any unexamined messages (which are also not deleted) exist in the file that you are sorting.

15i5

Commands to Read Other Message Files

15i5a

<- Read file name: <FILE-NAME>

You can use MSG on any file which has a message format. This means you can peruse or modify files created with the "put" or "move" commands. If, for example, you have a file containing messages pertaining to MSG problems, you can read it to make sure you've taken care of them. Read is the command which lets you read files other than MESSAGE.TXT;1. It also prints out the recent header information for that file. If that file has old messages which have not yet been

'examined', you will be informed. You will also be told if any of the old messages in the file are deleted. 1516

Commands to Sequence through the Messages 1516a

<- Current message is nn of mm messages.
in file: <FILE-NAME>

This command tells you (1) the number of the current message, (2) the total number of messages, and (3) the file name of the currently active file. The current message is either the last message typed on your terminal or, if you have not typed one yet, either after the last message if the file had no recent messages, or before the first recent message. This command will let you know where the Next and Backing up commands will start, i.e. the first message they will type if used. Finally, it will tell you what the currently active message file is. 151j

<- Go to message number: <NUMBER>

This will allow you to change the Current message number explicitly. If <NUMBER> is not in the range of acceptable numbers (i.e. it is less than 1 or greater than the number of messages in the file), or you did not type a number, you will be told and the Current message number will not be changed. However, there are several other options which are specified by the FIRST character typed:

- a. E for the end of messages (the last message)
 - b. L for the last message (same as E).
 - c. B for the beginning of messages (message number 1)
 - d. escape (alt-mode) for current message number
- 151j1

<- Next message is:

This command types the next message (Current message number + 1) if it is not a deleted message or you are not at the end of the list of messages. The Current message number is always incremented. 151j2

<- <line feed>

Same as Next. Types the message following the current message, and sets the current message to be that message. 151j3

<- Backing up -- previous message is:

This command always types the previous message (i.e., Current message number - 1). It is the inverse of the Next command. It always decrements Current message number. 151j4

<- ^

This is equivalent to the Back command. It types the previous message and sets the current message to be that message. 151j5

<- ^H

The <control-H> (or New-line) command is equivalent to the Back command. It types the previous message and sets the current message to be that message.

15j6

Other commands

15j6a

<- Verbose

This is a binary switch which causes the program to go into either 'Short timeout mode' or 'Long timeout mode', and tells you which is the setting that it changes to. The default is 'Short timeout mode'. Long timeout mode gives additional prompting regarding what is expected to be typed in.

15j7

<- Koncise

This is a binary switch which causes the program to go into either 'Concise timeout mode' or 'Short timeout mode' (the default), and tells you which is the setting that it changes to. Concise timeout mode shortens some of the timeout that MSG gives when it is interacting with the user. It is meant for 'advanced' users only.

15j8

<- Inclusion of length in header

This command is a binary switch which causes the program to go into a mode where header listings caused by the Header command will have the number of characters in the message included as part of the subject field. The default is that the length will not be included. Note that when you read a file initially, the length of 'recent' messages will always be included in the initial listing of recent headers.

15j9

<- *

The * command will ask you if you would like all messages sent by you with the answer command to also be stored on a file called SAVED.MESSAGES, if that file already exists in your directory. Answer with either 'Y' or 'N' for 'Yes' or 'No'.

15j10

<- Zap profile [Confirm]

The Zap profile command will allow you to set up a user profile file for yourself without having to know the format of such a file. For the time being, the profile information will be limited. Namely, you can preset your mode (Koncise, Verbose or Normal), you can default all Header listings to include (or exclude) the length of the message, and you can set a default which will cause all messages sent by you with the Answer command to be saved on a file called SAVED.MESSAGES. If that file doesn't exist when this command is executed, then you will be asked if you wish MSG to create the file for you. The answers you give

to the questions posed, in addition to creating the appropriate entry in the profile file, also changes (if necessary) those settings at the top level of MSG.

15j11

<- ; <COMMENT>

This command is mainly intended to allow you to talk with somebody over a link while you are in MSG. It eats all characters except <return> and control-Z (^Z), which return you to the command level of MSG. Two other characters have special effects. <delete> (<rub-out>) will type the string 'XXX' and is useful in indicating that the previous word (or phrase) should be ignored. <line-feed> will cause effectively a carriage return and tab sequence to be typed. This way you can type more than one line of text. NOTE: the standard TENEX editing characters (e.g. control-A) are treated as any other character and perform no special function.

15j12

Command to Run Other Programs

15j12a

<- Sndmsg [confirm]

This command will start up SNDMSG and give control of the terminal to it. When SNDMSG is finished (i.e. when you have sent the message), it will turn control back to MSG in the same state as it was before you sent the message. Control-N (^N) will ask if you wish to abort. If you provide a positive confirmation, then you will be returned to the top level of MSG. Otherwise, you will be returned to SNDMSG.

15j13

<- Answer message number: <MESSAGE-NUMBER>

Send response to: <ANSWER SUB-COMMAND>

This facility allows you to send a message to the sender of a message, and (at your discretion) those people to whom that message was sent, without having to type their addresses to Sndmsg.

15k

The <ANSWER SUB-COMMAND> can be any of the following: S --

Sender of the original message only

<RETURN> -- same as S.

O -- Original Sender with a cc: to <login directory> A -- All recipients of the original message (that is, the sender of the message and all addresses on the To: and cc: portions of the message)

R -- Respond to original sender only, but allow specification of additional cc: fields

E -- send the answer to Everybody who got the original, but allow specification of additional cc: fields. Typing anything else aborts the command.

15k1

The <MESSAGE-NUMBER> can be any argument that the Go command takes:

15k2

- a. a message number
- b. E for the end of messages
- c. L for the last message (same as E).
- d. B for the beginning of messages (message number 1)
- e. escape (alt-mode) for current message number
- f. <return> for current message number.

15k3

The header of the message specified is also typed so that you may be sure you are answering the correct message. In fact, the header is typed after you have specified the message number, but before you are asked to supply the sub-command.

15k4

If you respond with either of the R or E sub-commands, you will be permitted to specify an additional set of addresses which will be passed to SNDMSG as part of the cc: list. Some of the SNDMSG conventions are NOT implemented. These are the control-B feature which allows specification of a file, and the feature which allows you to specify a global host name (which spreads across several user names). Also, rubout aborts the Answer command! Local user names and remote site names are checked for validity.

15k5

An attempt is made to insure that all addresses are valid (i.e. all host names on remote addresses, and user names on local addresses), and that no duplications are present. If clarification is necessary from the user, you may be asked some questions. If these questions are posed, all type-ahead is deleted. If relevant, MSG will issue a warning if either the To: or cc: destination fields of the message have a destination list as part of the field (like LISP-USERS:). When control is given to you to type your answer, you will be typing to the message acquisition portion of SNDMSG (i.e. that part which normally would prompt you by typing "Message (? for help):"). Control-N (^N) will ask if you wish to abort. If you give positive confirmation, then you will be returned to the top level of MSG. Otherwise, you will be returned to SNDMSG.

15k6

If the response is going to anybody other than the original sender of the message, and there is a file named SAVED.MESSAGES in the login directory, you will be asked if you want your copy of the message to go to that file. If a positive response is given, then the login directory name will NOT appear in the destination lists.

15k7

<- Forward (message sequence) <MSG-SEQUENCE>

This facility will allow you to send copies of messages you have received to other people. First, you will be asked to provide the subject of this forwarded message. Then it will hand SNDMSG the subject and those messages you want forwarded, and leave you in SNDMSG in such a way that the message being forwarded can be edited, or your own comments added. You will be left in SNDMSG as though you had typed the forwarded message in yourself. When done, type a control-Z and then specify, in the standard way, to whom the mail is going. Once in SNDMSG, typing control-N (^N) will ask if you wish to abort. If you give a positive confirmation in the standard way, then you will be returned to the top level of MSG. Otherwise, you will be returned to SNDMSG.

15k8

<- Jump into lower fork running: <FILE-NAME>

This command is an escape in MSG in case you wish to run another program such as TECO, PUB, the EXEC, and so on. It searches directories to try to find the program you are asking it to run. The search list is, in order, <SUBSYS>, <SYSTEM>, your connected directory, and the login directory if different from the connected directory. This way, you can run EXEC without having to type the complete information (<SYSTEM>EXEC.SAV).

15k9

If you decide to leave the lower fork, but want to continue it at a later time, all you need do is type an escape as the first character of the file name you are requested to provide. This will cause the old file name (preceded by an appropriate message) to be printed, and then you will be asked to confirm in the standard way. If you provide a positive confirmation, you will be asked if you want to continue or start that program. Typing 'C' for continue will put you back in the lower fork at the place where you exited; typing 'S' for start will restart the program.

15k10

<- Xed (editor) [confirm]

This command will start up XED (a text editor written at ISI). It has the capability to give SNDMSG the text built while in the editor as the body of the message. When you quit XED you will return to MSG. Each additional time that you execute the XED command, you will be returned to the SAME copy of XED (with the old text buffers intact) as you left the last time.

15k11

<- ^Exec [confirm]

When you type control-E, the program will type "Exec" to you and ask for confirmation. This command is intended to give you a new copy of the EXEC with a minimum of hassles. To leave that EXEC and return to MSG, type Quit. If you decide that

NMSG Manual

you want a copy of the EXEC again, and you use this command, you will be given the same EXEC with all of your context intact.

15k12

This completes the list of MSG commands. There is only one item left to mention.

16

Receiving New Messages While Using MSG

16a

MSG, on typing a command or returning from the execution of a command, checks to see if your currently active message file, usually MESSAGE.TXT;1, has been written into. If it has, it prints out that fact and the headers for the new messages. It then executes your command or returns to command mode, accordingly.

16b

Command Summary

16b1

Cmdnd. Char.	Meaning
A	Answer message number: <MESSAGE-NUMBER>
Send response to S	-- Sender of current message only <return> -- same as S
O	-- Original sender, cc: to <login direct.> A -- All recipients of current message R -- Respond to sender, specify additional cc:
E	-- Everyone, specify accitional cc: B Backing up -- previous message is:
^	Same as Backing up
*H	Same as Backing up
C	Current message is nn of mm messages in file: <FILE-NAME>
D	Delete (message sequence) <MSG-SEQUENCE>
*E	Exec [confirm]
E	Exit and update old file <FILE-NAME> [confirm]
F	Forward (message sequence) <MSG-SEQUENCE>
G	Go to message number: <MESSAGE-NUMBER>
H	Headers (message sequence) <MSG-SEQUENCE>
I	Inclusion of length in header
J	Jump into lower fork running file: <program name> [confirm] K
K	Koncise -- provides shorter prompting
L	List (message sequence) <MSG-SEQUENCE> on file name: <FILE-NAME>
M	Move (message sequence) <MSG-SEQUENCE> into file name: <FILE-NAME>
N	Next message is:
<lf>	(line feed) same as Next message is:
O	Overwrite old file <FILE-NAME> [confirm]
P	Put (message sequence) <MSG-SEQUENCE> into file name: <FILE-NAME>
Q	Quit [confirm]

NMSG Manual

```

R  Read file name: <FILE-NAME>
S  Sndmsg [confirm]
T  Type (message sequence) <MSG-SEQUENCE>
U  Undelete (message sequence) <MSG-SEQUENCE>
V  Verbose -- provides more prompting
W  Write file <FILE-NAME> sorted by message arrival time
   [confirm] X  Xed [confirm]
Z  Zap profile [confirm]
*  * Will you always want to save your answers to messages on
   SAVED.MESSAGES? [Confirm]
?  ? Type command character for its description, ? for summary
;  ; Comment -- <return> or ^Z returns you to command level

```

16c

To abort a command on typein, type rubout (delete). Abort terminal output with ^O (control-O). Confirm with Y or <return>.

17

Errors While Reading a Message File

17a

When reading a file in MSG (either at startup or with the 'Read' command), the file MUST be in the so-called message file format. If MSG recognizes that the file does NOT conform to this format, you will be told so. The following are the circumstances which might cause the file to become unreadable, and some suggestions for getting around the problems.

17b

The file is a message file (that is, one or more valid messages have been read from it), but somewhere in the middle it does not conform to the message file format. It could be: (1) It has a hole in it. Read the file with a text editor to get rid of the hole, and write it back out, and reuse MSG. Try this first. If this doesn't work, MSG will give you an error at the same place. Then you can try the second suggestion: (2) If suggestion 1 didn't work, then the file has internal byte counts which do not match the actual file. Either you used a text editor on your message file changing the number of bytes but not the byte counts or your file was mysteriously altered. The date of a message could not be read. Either the byte count for the last message read was wrong, or there is junk between the last message read and the one with the error. Using some editor, find the last message read. The first line of that message contains a date-and-time followed by a byte count indicating how many characters are in the message body starting on the following line. Skip that many characters of the message body. You should be at the date-and-time line of the next message. If there is junk there, delete it. Otherwise, try to fix the count so it is pointing at the date-and-time of the next message.

17c

The beginning of the file does not conform to the message file format. It could be: (1) the file is not a message file -- sorry,

we can't help you there. (2) It is a message file with a bad first line -- probably a blank line. Read the file with a text editor. If the second line begins with a time and date then delete the first line and reuse MSG on the new file. (3) It is a message file with a hole at the beginning. Read it with a text editor to get rid of the hole, write it out and reuse MSG.

17d

Postscript

17d1

A history of MSG is now in order. It originated as a set of Teco macros written by Larry Roberts, the resultant system being called RD. What followed was a program by Barry Wessler called (I believe) NRD. Marty Yonke and I effectively rewrote it into a program called WRD. Marty then rewrote that effort, calling the result BANANARD. BANANARD was really the starting point for the MSG program, but the ideas really originated long ago. It is probably an understatement to say that without the efforts by Barry and Marty, MSG wouldn't exist today.

17e

NMSG Manual

(J33535) 23-SEP-75 10:50;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /SRI-ARC([INFO-ONLY]) AMC([INFO-ONLY])
; Sub-Collections: SRI-ARC AMC; Clerk: RA3Y; Origin: < PANKO,
NMSG-DOCUMENTATION.NLS;1, >, 23-SEP-75 10:43 RA3Y ;;;;####;

33535 Distribution

Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles H. Irby,
Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk H. Van
Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson, Don I. Andrews,
U.S. Army Materiel Command ,
Susan K. Ocken, Raphael Rom, David C. Smith, Buddie J. Pine, Andy
Poggio, David L. Retz, Laura J. Metzger, Karolyn J. Martin, Jan A.
Cornish, Larry L. Garlick, Priscilla A. Wold, Pamela K. Allen,
Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log Augmentation,
Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter, Robert Louis
Belleville, Ann Weinberg, Adrian C. McGinnis, Robert S. Ratner, David
S. Maynard, Robert N. Lieberman, Sandy L. Johnson, James H. Bair,
Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M. Beck, Marcia L.
Keeney, Elizabeth K. Michael, Jonathan B. Postel, Elizabeth J.
Feinler, Kirk E. Kelley, N. Dean Meyer, James E. (Jim) White

TWX from LHD to George Lalonde re VUcom for MIKE

● Hope this looks alright; also hope it doesn't take a month to get it.

TWX from LHD to George Lalonde re VUcom for MIKE

To: Mr. George Lalonde, CC Marketing
TWX No. 610 427 0421

From: Mr. L.H. Day, Staff Supervisor - Business Planning

Re: Request for installation of VUcom terminal in Mike Bedford's residence.

I would like you to arrange the installation of a VUcom terminal in Mike Bedford's residence.

Mike has been heavily involved in the Business Planning Group's use of NLS, a proto-type office automation system, and he finds it necessary to have a terminal at his home in order to maximize his efficiency with the system. Also, his experiences with working from home will be valuable to our evaluation of this type of system.

Mike is most familiar with the exact specifications required for the VUcom, and I will leave it in your hands to iron this out. Also, he will be able to specify what he needs in the way of modem equipment.

The address of the installation is below:

Michael T. Bedford
Boyd Settlement Road
Huntingdon County
Quebec

(Boyd Settlement Road is the second road south of the Chateauguay River at Dewittville, Quebec, on Route 138)

The modem is to be acoustically coupled to Mike's residence line: (514) 264-4317.

Since Mike is expecting a major project to get underway in early October, he would appreciate having the installation completed at the earliest date possible. Thank you for your cooperation in completing this order.

L.H. Day
Staff Supervisor - Business Planning

TWX from LHD to George Lalonde re VUcom for MIKE

(J33537) 23-SEP-75 13:38;;; Title: Author(s): Michael T.
Bedford/MIKE; Distribution: /LHD([INFO-ONLY]) ; Sub-Collections:
NIC; Clerk: MIKE;

33537 Distribution
Lawrence H. Day,

Bug in Load-File For Journal Items That Are Messages

Hi Pam (Sandy?). There's something that doesn't work right when you do a load file of a journal item that's a message, and you do it by number only. Like if there's a message and it's number is 12345, and i do "Load File 12345<CA>" it doesn't quite work right. NLS goes out and looks up which one of the journal's message files the item is in and loads it, but doesn't leave the CM at the right place. It leaves the CM at the top of the file instead of at the branch where the item in question lives. So if after it's finished doing the load, you do a "Print Branch<CA><CA>", you don't get the item, but rather statement 0 with all those stupid item numbers in it. I only know this happens for sure in TNLS, but I'll bet the same thing happens in DNLS as well. Thanx - Jeff

1

Bug in Load-File For Journal Items That Are Messages

(J33538) 23-SEP-75 14:49;;; Title: Author(s): Jeffrey C.
Peters/JCP; Distribution: /FEED([ACTION]) ; Sub-Collections:
SRI-ARC; Clerk: JCP;

33538 Distribution
Special Jhb Feedback,

Unlig and Network Delivery

● Can anybody give me some documentation on how network submission works?

Uhlig and Network Delivery

23-SEP-75 1512-PDT MEYER: Uhlig's Feeling about Journal
Distribution: PANKO, norton
Received at: 23-SEP-75 15:12:39-PDT

1

He finds it a pain in the neck to read his journal mail? We know that it takes people a while to be comfortable enough in NLS to wander around... How about switching his journal delivery to "Network", i.e. to deliver his citations to his message.txt file?

1a

Uhlig and Network Delivery

(J33539) 23-SEP-75 16:08;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /FEED([ACTION]) US([INFO-ONLY]) NDM([
INFO-ONLY]) ; Sub-Collections: SRI-ARC US; Clerk: RA3Y;

33539 Distribution

Special Jhb Feedback, Susan Gail Roetter, Priscilla A. Wold, Jeanne
M. Beck, Pamela K. Allen, Rita Hysmith, Sandy L. Johnson, N. Dean
Meyer,

The Support of User Programs: Need for person to coordinate

I noticed in (26478,) that Kirk "told JAC3 all about user programs", which inspired this.

User Programs come in three classes, briefly: A--fully supported, B--L-10 procedures guaranteed, C--not supported. Only Class a programs reside in <programs>. Class B and some Class C reside in <xprograms> (not clearly defined). Training, documentation and Help are provided for Class A (and the load program command defaults to <programs>).

1

Recently, some exciting programs have been written by, with and for clients, e.g. see (HJOURNAL, 33523, 0:w) by DLS. It takes significant personpower to support the software, write Help and documentation, and design training for Class A programs. However, I think we need a coordinator who can begin to review the various programs that will continue to be written by clients and work toward making them available in some coherent way. It is a basic service within the AKW concept.

2

The Support of User Programs: Need for person to coordinate

(J33540) 23-SEP-75 17:49;;; Title: Author(s): James H. Bair/JHB;
Distribution: /BJP([ACTION]) JCN([ACTION]) RWW([ACTION])
JAC3([ACTION]) SRI-ARC([INFO-ONLY]) ; Sub-Collections: SRI-ARC;
Clerk: JHB;

33540 Distribution

Kirk E. Kelley, N. Dean Meyer, James E. (Jim) White, Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk H. Van Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson, Don I. Andrews, Buddie J. Pine, James C. Norton, Richard W. Watson, Jan A. Cornish, Susan K. Ocken, Raphael Rom, David C. Smith, Buddie J. Pine, Andy Poggio, David L. Retz, Laura J. Metzger, Karolyn J. Martin, Jan A. Cornish, Larry L. Garlick, Priscilla A. Wold, Pamela K. Allen, Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log Augmentation, Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter, Robert Louis Belleville, Ann Weinberg, Adrian C. McGinnis, Robert S. Ratner, David S. Maynard, Robert N. Lieberman, Sandy L. Johnson, James H. Bair, Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M. Beck, Marcia L. Keeney, Elizabeth K. Michael, Jonathan B. Postel, Elizabeth J. Feinler

*Files to COM:

The file, Morgan.com;9, is on tape 0002. 2 sets of copyflo proofs will be made. One for sri-arc and one to go to Fortna at ETS. Will be mailed from DDSI on 9-25-75. Pam

*Files to COM:

(J33541) 23-SEP-75 18:12;;; Title: Author(s): Special Jhb
Feedback/FEED; Distribution: /DMB([ACTION] dpcs notebook please)
&DPCS([INFO-ONLY]) FEED([INFO-ONLY]) ; Sub-Collections: SRI-ARC
DPCS; Clerk: FEED;

33541 Distribution

Delorse M. Brooks, Documentation Production and Control System
Interest Group , Special Jhb Feedback,

File to COM

File, 110-guide.com;1, is on tape 0001. Will be ready on 9-25-75.
One set of copyflo proofs to be made. Pam

1

File to COM

(J33542) 23-SEP-75 18:16;;; Title: Author(s): Special Jhb
Feedback/FEED; Distribution: /DMB([ACTION] dpcs notebook please)
&DPCS([INFO-ONLY]) FEED([INFO-ONLY]) NDM([INFO-ONLY]);
Sub-Collections: SRI-ARC DPCS; Clerk: FEED;

33542 Distribution

Delorse M. Brooks, Documentation Production and Control System
Interest Group , Special Jhb Feedback, N. Dean Meyer,

New capability for checking page use of any directory

In response to requests by some of our clients, Jeff Peters has made one of his programs available at OFFICE-1, ISIC, and BBNB for anyone to use. This allows one to check the page allocation and page use of any directory, and may be especially beneficial for architects. At ISIC and OFFICE-1 you call it by typing "pages<cr>" at the exec. At BBNB type "<netsys>pages<cr>". After "DIR" is typed by the system one should type in the name of the directory that is to be checked followed by a space. The information will then be printed out. Any number of directory names may be typed in, one at a time. To get out type a <cr>. The total number of pages allocated and pages used of the directories checked will be given before one is returned to the exec. Please send comment or questions to Feedback, Pam

1

New capability for checking page use of any directory

(J33543) 24-SEP-75 00:44;;; Title: Author(s): Special Jhb
Feedback/FEED; Distribution: /KWAC([INFO-ONLY]) SRI-ARC([INFO-ONLY
]) FEED([INFO-ONLY]) ; Sub-Collections: SRI-ARC KWAC; Clerk: FEED;

33543 Distribution

Don I. Andrews, Special Jhb Feedback,
Pamela K. Allen, Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log
Augmentation, Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter,
Robert Louis Belleville, Ann Weinberg, Adrian C. McGinnis, Robert S.
Ratner, David S. Maynard, Robert N. Lieberman, Sandy L. Johnson,
James H. Bair, Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M.
Beck, Marcia L. Keeney, Elizabeth K. Michael, Jonathan B. Postel,
Elizabeth J. Feinler, Kirk E. Kelley, N. Dean Meyer, James E. (Jim)
White, Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles
H. Irby, Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk
H. Van Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson
Joseph L. Ehardt, Marilynne A. Sims, Elizabeth F. Finney, Lawrence A.
Crain, E. S. VonGehren, Glenn A. Sherwood, Kathy L. Mabrey, Jeanne
M. Beck, David A. Potter, Robert N. Lieberman, Terry H. Proch, Ronald
P. Uhlig, Susan Gail Roetter, Michael A. Placko, Stanley M. (Stan)
Taylor, Elizabeth J. Feinler, Rudy L. Ruggles, Frank G. Brignoli,
Robert M. Sheppard, Richard W. Watson, Douglas C. Engelbart, James C.
Norton, James H. Bair, Duane L. Stone, Inez M. Mattiuz, Connie K.
McLindon, Susan K. Ocken, Raphael Rom, David C. Smith, Buddie J.
Pine, Andy Poggio, David L. Retz, Laura J. Metzger, Karolyn J.
Martin, Jan A. Cornish, Larry L. Garlick, Priscilla A. Wold

KWAC Meeting, L-10 Training & Support

in response to (33536,)...some time should definately be set aside for discussion L-10 support, but I would not recommend that L-10 training/support be provided on the spot.

KWAC Meeting, L-10 Training & Support

Inez,

1

In answer to your questions:

1a

1. Yes we need L-10 support

1a1

2. Yes we have received some, but will need more

1a2

My understanding of L-10 support from SRI is as follows:

1b

Most of the L-10 programmers reside under Watson and not Norton. They are working on other jobs, and therefore not available on a consistent basis to subscribers of the Utility. Kirk Kelly, Dean Meyer (on a part time basis) and possibly Karolyn Martin are the only L-10 people, that I am aware of, that work for Norton.

1b1

L-10 support now comes under the general category of training. One can make private deals with Norton to swap L-10 training for NLS training, ie if you feel that you don't need the services of Susan and her group, then you might be able to have some of dean's time for example. This is strictly a gentlemen's agreement with Norton at this time.

1b2

In the contracts that we have had with SRI, there has been a gradual trend toward "pay-for-what-you-get". This can mean good or bad news depending on what you now get and what you now pay. In general, I would say that the organizations that are getting the most from the system are the ones that are yelling the loudest that they are not being supported adequately. This is to be expected. As organizations mature in their knowledge of NLS, there seems to be a natural progression toward wanting to learn L-10 and CML, in order to program special purpose applications at each site. We are trying to negotiate a contract for next year which has NLS training and L-10 support explicitly listed and costed seperately from computer time.

1b3

There are a number of organizations that have expressed a desire for more formal L-10 training...Bell, NSA, IBM, and RADC immediately come to mind. There seems to be sufficient interest and the need seems urgent, where we we should consider jointly funding a course. Follow-up day-to-day support for specific applications may have to be negotiated seperately for each site.

1b4

DLS 24-SEP-75 08:52 33544

KWAC Meeting, L-10 Training & Support

(J33544) 24-SEP-75 08:52;;; Title: Author(s): Duane L. Stone/DLS;
Distribution: /AID([INFO=ONLY]) ; Sub-Collections: RADC AID; Clerk:
DLS;

33544 Distribution

Frank G. Brignoli, Inez M. Mattiuz, Connie K. McLindon, Michael A. Placko, David A. Potter, Terry H. Proch, Rudy L. Ruggles, Robert M. Sheppard, Duane L. Stone, Stanley M. (Stan) Taylor, Ronald P. Uhlig,

LINKS&STUFF

Up to now, I have done the following:

Taken one of my existing files EFF-SYSANALSUPP and created a named branch B8660101 for my Job Order Number,

To this I have appended the directives that control the printing of the viewgraph file,

I have copied my viewgraph branch from the PETELL,AUGVG file to follow the named branch B8660101 one level down,

I have appended text in the form of a link <KENNEDY, EFF-SYSANALSUPP, B8660101: nWD> to the statement in the CARRIER,PRFILE that contains the content B8660101,

This provides the following capability,

You can load the file CARRIER,PRFILE and print it at your leisure. When you find you are interested in a particular JONO, you can, for example, use the command Jump to Content First T: B8660101. You can then use the command Jump to Link A: +e

Or if you already know the statement number that contains B8660101 you can use the command Jump to Link A: 5e4 +e

After getting to the link, your viewspecs are already set, all you need do is use the command Output to Terminal<cr> followed by yes,

What has to be looked at:

I have created the link <LAMONICA, EFF=TOOL-INTEGRATION-STUDIES, C0950101: nWD> for one of Frank LaMonica's Efforts,

The links can be created easily in Bobbie's file by copying most of the info, with no typing errors etc., thus retaining the exact numbers and names of the efforts.

What needs to be done is to create the appropriate files in each of the PE's directory.

One way is to create the files somewhere aand then have the PE copy it,

I want to think a bit about the problems of replacing branches between files and easy ways to do it,

The sheer mechanics of maintaining these files. This has two aspects:

LINKS&STUFF

First the technical capability to edit. Not easy on the IMLAC,
almost impossible on the TTY's.

3f1

Second the discipline and the rules and conventions. When two
versions of the viewgraphs differ which supersedes?

3f2

I'd like to have you and Stoney to think about this a little to see
if anything has been overlooked. Then, unless everything is quite
clearcut, we ought to get together.

4

LINKS&STUFF

(J33545) 24-SEP-75 11:25;;; Title: Author(s): Edmund J.
Kennedy/EJK; Distribution: /JLM([ACTION]) DLS([ACTION]) RJC([
INFO-ONLY]) ELF([INFO-ONLY]) MDP([INFO-ONLY]) TJB2([
INFO-ONLY]) ; Sub-Collections: RADC; Clerk: EJK;

33545 Distribution

John L. McNamara, Duane L. Stone, Roberta J. Carrier, Edward F.
LaForge, Marcelle D. Petell, Thomas J. Bucciero,

RA3Y 25-SEP-75 12:45 33547

Dialogue Support System Conference Transcript (LONG!)

This is the transcript of the Dialogue Support System conference to date. If you wish to keep abreast of things, the transcript is in my directory <panko,transcript,1:xy>.

Dialogue Support System Conference Transcript (LONG!)

INTRODUCTION

1

RA3Y 11-SEP-75 15:17 33459
 A Teleconference on a New Journal System
 Location: (JOURNAL, JRNL29, J33459:gw)
 *****Note: Author Copy*****

1a

Message:

1a1

Doug has asked me to coordinate the promotion of a new and more beautiful Journal system. I would appreciate you comments on what features the next-generation journal system should have. Please send them to my ident, RA3Y.

I will put all suggestions in a file in my directory: Panko,transcript. please feel free to read them and respond to any items by sending comments to me via Sendmail (no Sendmessages, please).

About titles, When you send something, make the title as specific as possible. For example, don't remark that it is for the Journal teleconference. If we keep titles specific, we may be able to search for them.

panko,transcript, also contains a number of "exhibits," i.e, old citations dealing with the journal system.

Thanks, Ra3y

1a1a

Dialogue Support System Conference Transcript (LONG!)

NEWS

2

Dialogue Support System Conference Transcript (LONG!)

EXHIBITS

3

KIRK 12-SEP=75 23:56 26464
 Readmail design document
 Location: (HJOURNAL, 26464, 1:w)

3a

Comments: Should be considered as an 'exhibit' for RA3Y's new Journal design. After this design was written, it was decided Tabs, Help, and Air Force Manual formatting had priority, so it has not been implemented. The design is written as a Userguide and a Help file. It contains suggestions received since the last time it was journalized.

3a1

RA3Y 11-SEP=75 20:49 33466
 Recorded Dialog: Section from the Final Report
 Location: (HJOURNAL, 33466, 1:w)
 *****Note: Author Copy*****

3b

Comments: This is Branch 13 of the Final Report on Project 1868; this report is scheduled to be COM'd soon. The enclosed branch is Jim White's chapter on Recorded Dialog.

3b1

JEW 23144
 Description of a Multi-Host Journal System
 Location: (MJOURNAL, 23144:dw)
 *****Note: Simulated Citation*****

3c

JAKE 19-AUG=75 03:02 26293
 Network Identification System - Comments and Suggestions
 Location: (HJOURNAL, 26293, 1:w)

3d

DCE 29-JUL=75 12:43 33076
 NLS TELECONFERENCING FEATURES: THE JOURNAL, AND SHARED-SCREEN
 TELEPHONING
 Location: (HJOURNAL, 33076, 1:w)

3e

Dialogue Support System Conference Transcript (LONG!)

Comments: This paper was submitted for presentation at COMPCON 75, to be held in Washington D.C., 9-11 Sep 75. They require a 'digest' for their proceedings, limited to a fixed maximum page space.

3e1

JBP 29-APR-75 01:52 25806

'MSG' mail reading and processing program documentation

Location: (GJOURNAL, 25806, 1:w)

*****Note: [INFO=ONLY] *****

3f

Comments: The MSG mail program is installed as a TENEX subsystem at BBNB.

3f1

RA3Y 4-SEP-75 09:19 33407

Annotated and Modified Message Service Group Transcript

Location: (HJOURNAL, 33407, 1:w)

*****Note: Author Copy*****

3g

Comments: Here is an NLS version of the Message Service Group's ongoing teleconference. I put it together on September 3, but it is not quite up to date. Output processor directives are included.

WARNING: ABOUT 200 PAGES! If you did an Output (to) Journal this morning, of course, you already know that.

3g1

The Message Service Group consists of people from ARC, BBN, USC-ISI, RAND, AMC and a scattering of other places who are chatting about how to design and integrate message services.

3g2

Dialogue Support System Conference Transcript (LONG!)

CONFERENCE RECORD

4

DVN 13-APR-73 08:58 15931
 The Journal is One Kind of Dialog Support
 Location: (JOURNAL,JRNL9,J15931:w)
 *****Note:
 (Secondary Distribution Copy from DVN)*****

4a

Message:

4a1

The Journal is a subset of the Dialog Support System. Shared images, for example are, another dialog support system within nls. But because the journal has been far and away the most prominent system, people have often confused the part with the whole; that is sloppy use of language.

4a1a

DVN 24-SEP-75 12:45 26551
 Comments To Individual Journal Recipients Should be Longer.
 Location: (JOURNAL, JRNL29, J26551:gw)

4b

Message:

4b1

I mean when you write in a special note to one person after an ident. I use this feature a lot but if I have more than about 4 lines it says "String too long". Often I have just typed some long, subtle implication, and I am really pissed off when that happens. It could be atleast 2000 characters. While I am on the subject, you should be able to make such comments to the recipients of forwards, and forwarding should add the item to appropriate subcollections.

4b1a

DCE 22-SEP-75 09:27 33518
 Note on the term Dialog Support
 Location: (HJOURNAL, 33518, 1:w)

4c

SGR 19-SEP-75 19:38 26514
 Dialog Support
 Location: (JOURNAL, JRNL29, J26514:gw)

Dialogue Support System Conference Transcript (LONG!)

*****Note: [ACTION] *****

4d

Message:

4d1

Just a note on (33510,) - A definition of Dialog Support. In the third TNLS course we spend some describing the concepts Journal and Dialog Support. A suggestion made by Jim Bair as to the content of this little discourse which I like goes as follows: Most professions have journals which are used to support dialogue between geographically distributed professionals. Every group of people with a common interest needs some method for "supporting" communications (dialogue), therefore we have a "journal" to aid our communications (support our dialogue). It just dawned on me that this is an analogy not a definition so it may not be of any real use to you but since I've taken time to type it I'll go ahead and send it.,.,

4d1a

RA3Y 12-SEP-75 08:06 33473
Using Keywords
Location: (JOURNAL, JRNL29, J33473:gw)
*****Note: Author Copy*****

4e

Comments: For the Journal conference

4e1

Message:

4e2

In your comments to the Journal conference (JOURNAL, JRNL29, J33459:gw), please include one or more of the following keywords. This will help readers browse through the conference record.

4e2a

Userware: Pertaining to what the user sees, especially commands and subsystems like Sendmail, Readmail, Teleconference, and Identification

4e2a1

Systemware: Pertaining to things the user does not see, especially the inner workings of the Journal system

Dialogue Support System Conference Transcript (LONG!)

Philosophy: Philosophical comments about Dialog Support Systems 4e2a2

Procedural: Procedural comment about the conference 4e2a3

Exhibit: to denote a substantive document 4e2a4

DVN 15-SEP-75 18:33 26472
 Command Words for Readmail
 Location: (JOURNAL, JRNL29, J26472:gw)
 *****Note: [ACTION] *****

4f

Comments: Comment on 26464. 4f1

Message: 4f2

In general I like the Readmail design. It would be nice to keep the verbal quality in initial NLS command words. Howabout "Scan" for "Brief" and "File" for "Catagory", 4f2a

RA3Y 12-SEP-75 07:00 33472
 Citation Format Suggestion: Show Number of Pages
 Location: (JOURNAL, JRNL29, J33472:gw)
 *****Note: Author Copy*****

4g

Message: 4g1

I would like to suggest that, in future journal citation formats, the number of pages in the submission be listed. This help users in deciding whether or not to print a citation. 4g1a

RA3Y 4-SEP-75 08:35 33410
 cost outlook for NLS
 Location: (HJOURNAL, 33410, 1:w)
 *****Note: Author Copy*****

4h

Dialogue Support System Conference Transcript (LONG!)

RA3Y 15=MAY-75 16:15 32519
 Contact Report: Bert Liffman
 Location: (GJOURNAL, 32519, 1:w)
 *****Note: Author Copy*****

41

Comments: This contact report was made for the SRI teleconferencing project. It details use of General Conferencing Systems Ltd.'s computer teleconferencing system by the Nonmedical Drugs Directorate of the Canadian Federal Government. Cost figures are included.

411

RA3Y 19=MAY-75 16:57 32549
 Teleconferencing and Computer Communication
 Location: (GJOURNAL, 32549, 1:w)
 *****Note: Author Copy*****

41

Comment: Draft paper surveying computer teleconferencing and general computer-based human communications.

411

Dialogue Support System Conference Transcript (LONG!)

REFERENCES

* indicates that references were brought online for the conference
(Well, we have asked Tymshare to bring them online, anyway). 5a

*(MULTI=HOST)Description of a Multi=Host Journal System
26135 MULTI=HOST 5b
(23144,1:wznC)

*(TEAM INTERACTION)Online Team Environment / Network Information
Center and Computer Augmented Team Interaction
26135 TEAM INTERACTION 5c
(13041,1:wznC)

*(JOURNAL)NLS TELECONFERENCING FEATURES: THE JOURNAL, AND
SHARED=SCREEN TELEPHONING 33076 JOURNAL 5d
(33076,1:wznC)

(DIALOG)First Half July Message Group Dialog
26135 Dialog 5e
(26135,1:wznC)

(JOURNAL)Journal mail - Fourth Quarter 1974
10 Jan 75 31617 Journal 5f
(31617,1:wznC)

(JOURNAL)Journal Mail - Third Quarter 1974
10 Jan 75 31616 Journal 5g
(31616,1:wznC)

(JOURNAL)Status of Suggestions Regarding Network Journal Delivery
9 Jan 75 31606 Journal 5h
(31606,1:wznC)

(JOURNAL)Journal Citations vs. ARPA Network Standards
24 Feb 75 25453 Journal 5i
(25453,1:wznC)

(JOURNAL)A Simple Commands Branch to Reject Journal Mail on

Dialogue Support System Conference Transcript (LONG!)

Specific Subjects (25042,1:wznc)	9 Jan 75	25042	Journal	5j
(JOURNAL)Hardcopy Journal 30 Jan 75 25254 Journal (25254,1:wznc)				5k
(JOURNAL)Comments on the 'final' journal citation 24 Feb 75 25452 Journal (25452,1:wznc)				5l
(DIALOG)Some Recent Dialog among the Message Service Committee 32806 Dialog (32806,1:wznc)				5m
(DIALOG)Recent Message Committee Dialog for Week through June 27 27 Jun 75 32840 Dialog (32840,1:wznc)				5n
(JOURNAL)Journal Mail - Second quarter 1974 23 Jul 74 30938 Journal (30938,1:wznc)				5o
*(JOURNAL)Network Journal Distribution 2 Dec 74 24647 Journal (24647,1:wznc)				5p
*(JOURNAL)NETWORK JOURNAL DELIVERY 2 Dec 74 24646 JOURNAL (24646,1:wznc)				5q
(JOURNAL)Network Journal Delivery 24645 Journal (24645,1:wznc)				5r
(JOURNAL)Journal citation dialogue: second go around; two alternatives; Call for a 37 24828 Journal (24828,1:wznc)				5s

Dialogue Support System Conference Transcript (LONG!)

*(JOURNAL)how to getwnls to print only the changed journal mail
 instead of eVerything 12 Aug 74 23759 journal 5t
 (23759,1:wznC)

(JOURNAL)Bug in journal citations
 26 Sep 74 24069 journal 5u
 (24069,1:wznC)

*(JOURNAL)NEW NLS Journal Delivery
 26 Sep 74 24067 Journal 5v
 (24067,1:wznC)

(JOURNAL)Output Journal bug
 19 Sep 74 24006 Journal 5w
 (24006,1:wznC)

(JOURNAL)Journal Indices
 19 Sep 74 23994 Journal 5x
 (23994,1:wznC)

(JOURNAL)Journal Confounds Bugs with Dreams
 24172 Journal 5y
 (24172,1:wznC)

(JOURNAL)BUG:journal system, pushing its way in.
 3 Oct 74 24157 journal 5z
 (24157,1:wznC)

(JOURNAL)SUBMISSION OF JOURNAL FROM THE NETWORK
 13 Oct 74 24210 JOURNAL 5ae
 (24210,1:wznC)

(JOURNAL)For A user otion to Turn off Journal notification
 23 Oct 74 24269 Journal 5aa
 (24269,1:wznC)

(JOURNAL)Comment on functional Documents & Journal Numbers
 28 Oct 74 24341 Journal 5ab
 (24341,1:wznC)

Dialogue Support System Conference Transcript (LONG!)

*(JOURNAL)journal citations again
 28 Oct 74 24337 journal 5ac
 (24337,1:wznc)

(JOURNAL)Opinion on Journal Citation Delivery Format
 24336 Journal 5ad
 (24336,1:wznc)

(JOURNAL)Functional Documents and Journal Numbers
 24325 Journal 5ae
 (24325,1:wznc)

(JOURNAL)Journal Citation Recommendations
 25 Oct 74 24324 Journal 5af
 (24324,1:wznc)

(JOURNAL)More On Journal Citations
 93 24318 Journal 5ag
 (24318,1:wznc)

(JOURNAL)journal citations revisited
 24315 journal 5ah
 (24315,1:wznc)

*(JOURNAL)journal headers
 24 Oct 74 24294 journal 5ai
 (24294,1:wznc)

*(JOURNAL)re J24269: Journal notification & info/action branches
 24 Oct 74 24291 Journal 5aj
 (24291,1:wznc)

(JOURNAL)Additional Thought on Input of Messages to the Journal
 5 Nov 74 24411 Journal 5ak
 (24411,1:wznc)

*(JOURNAL)SUMMARY OF ARC JOURNAL
 24621 JOURNAL 5al
 (24621,1:wznc)

Dialogue Support System Conference Transcript (LONG!)

(JOURNAL)One More thought about Journal Deliveryy
 7 Nov 74 24437 Journal 5am
 (24437,1:wznC)

*(JOURNAL)ARC Journal: ROUGH DRAFT of Report Chapter
 8 Nov 74 24448 Journal 5am
 (24448,1:wznC)

*(DIALOG)ARC Dialog Support: ROUGH DRAFT for a Report Chapter
 8 Nov 74 24450 Dialog 5ao
 (24450,1:wznC)

(DIALOGUE)Journal citation dialogue: second go around; two
 alternatives; call for a 37 24828 dialogue 5ap
 (24828,1:wznC)

(DIALOGUE)Kudlick's Jul 74 notes re, NIC Experience with Dialogue
 Support 13 Dec 74 24755 Dialogue 5aq
 (24755,1:wznC)

(DIALOGUE)My Thoughts about Recording Written Dialogue
 24431 Dialogue 5ar
 (24431,1:wznC)

(DIALOGUE)My thoughts about recording dialogue
 24430 dialogue 5as
 (24430,1:wznC)

(DIALOGUE)Recording our written dialogue
 3 Nov 74 24393 dialogue 5at
 (24393,1:wznC)

(DIALOG)Collection of Dialog on a Maling list Superintendent
 16 Sep 74 23976 Dialog 5au
 (23976,1:wznC)

(JOURNAL)Journal mail -Last quarter 1973
 6 Feb 74 30059 Journal 5av
 (30059,1:wznC)

Dialogue Support System Conference Transcript (LONG!)

(JOURNAL)On-line only Delivery of Journal Documents

6 Feb 74 30052 Journal
(30052,1:wznC)

5aw

(JOURNAL)Submitting Documents to the Journal

23 Jan 74 30002 Journal
(30002,1:wznC)

5ax

*(JOURNAL)Journal SUBCOLLECTIONS command

31 May 74 23174 Journal
(23174,1:wznC)

5ay

(JOURNAL)Network Journal Submission

1 May 74 22879 Journal
(22879,1:wznC)

5az

(JOURNAL)Journal Citations Reformatting Program

29 Apr 74 22853 Journal
(22853,1:wznC)

5be

(JOURNAL)some comments on the dual-journal system

9 Apr 74 22673 journal
(22673,1:wznC)

5ba

(JOURNAL)Response to (22598,) = On Journal Statistics

4 Apr 74 22625 Journal
(22625,1:wznC)

5bb

(JOURNAL)Comments on Journal Statistics

2 Apr 74 22598 Journal
(22598,1:wznC)

5bc

(JOURNAL)Yummy, Yummy! Journal Statistics are Neat!!!

1 Apr 74 22594 Journal
(22594,1:wznC)

5bd

*(JOURNAL)Preliminary Mapping of Journal Use for One Subject

1 Apr 74 22549 Journal
(22549,1:wznC)

5be

Dialogue Support System Conference Transcript (LONG!)

*(JOURNAL)Assorted Journal Statistics
 1 Apr 74 22547 Journal
 (22547,1:wznC) 5bf

(JOURNAL)The journal hardcopy command,
 28 Mar 74 22521 journal
 (22521,1:wznC) 5bq

*(JOURNAL)SCENARIO FOR USING THE NETWORK JOURNAL
 27 Mar 74 22507 JOURNAL
 (22507,1:wznC) 5bh

(JOURNAL)on-line journal indexes and other on-line files
 20 Mar 74 22426 journal
 (22426,1:wznC) 5bi

*(JOURNAL)Estimated Manpower Required to Implement the Multi-Host
 Journal System 14 Mar 74 22406 Journal
 (22406,1:wznC) 5bj

(JOURNAL)Journal delivery, hardcopy vs online
 4 Feb 74 21787 Journal
 (21787,1:wznC) 5bk

(JOURNAL)Scenarios for Sending to, and Retrieving From, Network
 Journal 30 Jan 74 21689 Journal
 (21689,1:wznC) 5bl

(JOURNAL)Sample of Network Journal Through FTP
 30 Jan 74 21682 Journal
 (21682,1:wznC) 5bm

(JOURNAL)Sample of Mail Sent Through TELNET to the Network Journal
 30 Jan 74 21681 Journal
 (21681,1:wznC) 5bn

(JOURNAL)Immediately Relevant Suggestions for the Journal System
 23 Jan 74 21595 Journal
 (21595,1:wznC) 5bo

Dialogue Support System Conference Transcript (LONG!)

(JOURNAL)Journal and the Utility, some feedback,
 23 Jan 74 21594 Journal 5bp
 (21594,1:wznc)

(JOURNAL)journal mail fourth quarter 1973
 21 Jan 74 21539 journal 5bq
 (21539,1:wznc)

(JOURNAL)journal mail fourth quarter 1973
 21 Jan 74 21538 journal 5br
 (21538,1:wznc)

(JOURNAL)Comments on proposed NIC identfile and journal changes
 16 Jan 74 21443 journal 5bs
 (21443,1:wznc)

(JOURNAL)Request for Means to Send Journal Items to Online
 Addressees Only 4 Jan 74 21294 Journal 5bt
 (21294,1:wznc)

(DIALOG)Dialog Support Ideas Solicited
 8 Jan 74 21332 Dialog 5bu
 (21332,1:wznc)

(DIALOG)Additional Comment on Your Proposal for Private Journal
 Dialog 29 Nov 73 20624 Dialog 5bv
 (20624,1:wznc)

(DIALOG)Reply to Jim White's Private Journal Dialog Proposal
 28 Nov 73 20601 Dialog 5bw
 (20601,1:wznc)

*(JOURNAL)DSS: New Journal Features under Consideration
 27 Dec 73 21224 Journal 5bx
 (21224,1:wznc)

(JOURNAL)Thoughts on Possible Exec-Level Journal Commands
 31 Jan 73 20933 Journal 5by
 (20933,1:wznc)

Dialogue Support System Conference Transcript (LONG!)

(JOURNAL)re JEW's 20543 Journal Privacy Proposal
 6 Dec 73 20750 Journal
 (20750,1:wznc) 5bz

(JOURNAL)Additional Comment on Your Proposal for Private Journal
 Dialog 29 Nov 73 20624 Journal
 (20624,1:wznc) 5cø

(JOURNAL)Reply to Jim White's Private Journal Dialog Proposal
 28 Nov 73 20601 Journal
 (20601,1:wznc) 5ca

*(JOURNAL)A Just-About-Final Journal Privacy Proposal
 26 Nov 73 20543 Journal
 (20543,1:wznc) 5cb

(JOURNAL)Suggestion for Change in Journal Hardcopy Format
 25 Nov 73 20492 Journal
 (20492,1:wznc) 5cc

(JOURNAL)Agreement that SNDMSG and Journal Should be more Closely
 Coupled 20 Nov 73 20392 Journal
 (20392,1:wznc) 5cd

(JOURNAL)Journal Distribution: Two Additions to be Implemented
 19 Nov 73 20380 Journal
 (20380,1:wznc) 5ce

(JOURNAL)Some answers to queries about the journal (20362,)
 19 Nov 73 20371 Journal
 (20371,1:wznc) 5cf

(JOURNAL)Regarding your Ident system and Journal Subcollection
 questions 12 Nov 73 20178 Journal
 (20178,1:wznc) 5cg

(JOURNAL)Suggestion for Initial File Journal Item Indexing
 4 Nov 73 20037 Journal
 (20037,1:wznc) 5ch

Dialogue Support System Conference Transcript (LONG!)

*(JOURNAL)What to do When Interogate Tells you a Journal File is
 Not On Line (reply to 2 Nov 73 20013 Journal 5ci
 (20013,1:wznc)

*(JOURNAL)Output Journal Mail Command
 1 Oct 73 19401 Journal 5cj
 (19401,1:wznc)

(JOURNAL)Test of Sndmsg vs. the Journal
 26 Sep 73 19309 Journal 5ck
 (19309,1:wznc)

(JOURNAL)A Comment on Journal Privacy
 13 Sep 73 19053 Journal 5cl
 (19053,1:wznc)

(JOURNAL)Questions About Private=Journal Catalogs and Speed of
 Proposed Implementation 11 Sep 73 18982 Journal 5cm
 (18982,1:wznc)

(JOURNAL)Response to 18069: Attaching Comments to a Journal Item
 31 Jul 73 18145 Journal 5cn
 (18145,1:wznc)

(JOURNAL)On Journal=Item Citation Naming; cf. (18011,), (18036,) and (18063,) 31 Jul 73 18132 Journal 5co
 (18132,1:wznc)

(JOURNAL)Answers to Questions About Net Journal Submission & Delivery 20 Jul 73 17964 Journal 5cp
 (17964,1:wznc)

(JOURNAL)a sample journal message
 18 Jul 73 17857 Journal 5cq
 (17857,1:wznc)

(JOURNAL)a sample journal session
 18 Jul 73 17856 Journal 5cr
 (17856,1:wznc)

Dialogue Support System Conference Transcript (LONG!)

- (JOURNAL)NWG/RFC 543: Network Journal Submission and Delivery
 16 Jul 73 17777 Journal
 (17777,1:wznC) 5cs

- *(JOURNAL)L10 Program to Tabulate Modes of Journal Delivery
 11 Jul 73 17746 Journal
 (17746,1:wznC) 5ct

- (JOURNAL)Network Journal Mail
 3 Jul 73 17628 Journal
 (17628,1:wznC) 5cu

- (JOURNAL)What to do If Your Journal Branch Is Destroyed
 2 Jul 73 17613 Journal
 (17613,1:wznC) 5cv

- (JUL)Journal Items received from Oct 72-Jul 73
 8 Nov 73 20118 Jul
 (20118,1:wznC) 5cw

- *(DIALOG)Summary of Dialog on Journal Headers
 7 May 73 16322 Dialog
 (16322,1:wznC) 5cx

- (DIALOG)Privacy in Journal Dialog == Initial Thoughts
 23 Apr 73 16052 Dialog
 (16052,1:wznC) 5cy

- *(DIALOG)The Journal is One Kind of Dialog Support
 13 Apr 73 15931 Dialog
 (15931,1:wznC) 5cz

- *(DIALOG)Question about Journal vs Dialog Support
 12 Apr 73 15756 Dialog
 (15756,1:wznC) 5d@

- (DIALOG)Dialog-Supported Debugging System
 15 Mar 73 15101 Dialog
 (15101,1:wznC) 5da

Dialogue Support System Conference Transcript (LONG!)

(ARC)* ARC JOURNAL INDEX BY TITLEWORD (PDP-10 entries only) to 28
 JAN 72 2 Feb 72 8715 ARC

To:
 (8715,1:wznC) 5db

(ARC)* ARC JOURNAL INDEX BY AUTHOR (PDP-10 entries only) to 28 JAN
 72 31 Jan 72 8714 ARC

To:
 (8714,1:wznC) 5dc

(ARC)* ARC JOURNAL INDEX BY NUMBER (PDP-10 entries only) to 28 JAN
 72 31 Jan 72 8713 ARC

To:
 (8713,1:wznC) 5dd

(ARC)* New ARC Journal Entries - Since 10 JAN 72 (issue
 28 Jan 72 8696 ARC

To:
 (8696,1:wznC) 5de

(Availability)* Reply to AAM on Status of Journal Availability
 23 Mar 72 9607 Availability

To: AAM RWW WSD JDH NICSTA
 (9607,1:wznC) 5df

(Background)* Program for determining proper startup of Journal
 Background Process 7 Feb 72 8799 Background

To: WRF JDH RWW JCN DCW KEV
 (8799,1:wznC) 5dg

(Bugs) Some Journal Bugs and Glitches
 22 Nov 71 8110 Bugs

To: WSD JDH
 (8110,1:wznC) 5dh

*(Catalog) Proposed New Journal Submission Features -
 Pre-specified Catalog Data 27 Jan 72 8672 Catalog

To: WSD DCE CHI
 (8672,1:wznC) 5di

(Catalog)* A Recommended Set of Conversion Conventions from the

Dialogue Support System Conference Transcript (LONG!)

Journal Catalog to the 27 Aug 71 7614 Catalog
To: WSD WLB JBN JCN (Jim this set of conventions
is more complete than a previous set you wsd and jbn had
discussed)
(7614,1:wznC) 5dj

(changes)* Proposed Journal changes and the new file system
11 Jan 72 8425 changes
To: JDH WSD WHP
(8425,1:wznC) 5dk

(Changes)* Proposed Journal Changes
7 Jan 72 8405 Changes
To: NPG JCN RWW
(8405,1:wznC) 5dl

Dialogue Support System Conference Transcript (LONG!)

(J33547) 25-SEP-75 12:45;;; Title: Author(s): Raymond R.
Panko/RA3Y; Distribution: /SRI=ARC([INFO-ONLY]) GCE([INFO-ONLY])
LHD([INFO-ONLY]) RWH([INFO-ONLY]); Sub=Collections: SRI=ARC;
Clerk: RA3Y; Origin: < PANKO, TRANSCRIPT,NLS;13, >, 25-SEP-75
12:38 RA3Y ;;; #####

33547 Distribution

Douglas C. Engelbart, Martin E. Hardy, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, James C. Norton, Jeffrey C. Peters, Dirk H. Van Nouhuys, Kenneth E. (Ken) Victor, Richard W. Watson, Don I. Andrews, Gwen C. Edwards, Lawrence H. Day, Roger W. Hough, Susan K. Ocken, Raphael Rom, David C. Smith, Buddie J. Pine, Andy Poggio, David L. Retz, Laura J. Metzger, Karolyn J. Martin, Jan A. Cornish, Larry L. Garlick, Priscilla A. Wold, Pamela K. Allen, Delorse M. Brooks, Beverly Boli, Rita Hysmith, Log Augmentation, Joseph L. Ehardt, Raymond R. Panko, Susan Gail Roetter, Robert Louis Belleville, Ann Weinberg, Adrian C. McGinnis, Robert S. Ratner, David S. Maynard, Robert N. Lieberman, Sandy L. Johnson, James H. Bair, Jeanne M. Leavitt, Rodney A. Bondurant, Jeanne M. Beck, Marcia L. Keeney, Elizabeth K. Michael, Jonathan B. Postel, Elizabeth J. Feinler, Kirk E. Kelley, N. Dean Meyer, James E. (Jim) White

Add (33523,...)User documentation on FORMATTER to DPC_S Subcollection DLS 25-SEP-75 13:36 33548

ref (26549,...), request to rejournal (33523,...)

Add (33523),,,,User Documentation on FORMATTER to DPCS Subcollection

Sure, go ahead and Rejournal (33523,). I didn't realize that forward would not add the item to a subcollection, but now that I think about it, why should it. My motivation in creating FORMATTER was to learn something about CML and its interaction with L=10. So I figured I might as well try for something which could potentially be useful. It has turned out that it is useful,,,the one flaw being the TYCOMs, which are forever failing for one reason or another. Several of the types of correspondance are now being accepted by the chain of command on lineprinter paper, so all is not lost,

1

Somehow I felt that a system with the word OFFICE=1 ought to have a couple of simple programs which had something to do with automating an office .

2

DLS 25-SEP-75 13:36 33548

Add (33523),...User Documentation on FORMATTER to DPcS Subcollection

(J33548) 25-SEP-75 13:36;;; Title: Author(s): Duane L. Stone/DLS;
Distribution: /DVN([INFO=ONLY]) ; Sub-Collections: RADC; Clerk:
DLS;

33548 Distribution
Dirk H. Van Nouhuys,