NIC 7591

Richard W. Watson
SRI-ARC
4 October 1971 1

TRANSMITTAL OF NETWORK INFORMATION CENTER USER GUIDE 2

Enclosed is a preliminary release of the Network Information Center User Guide which we will be using at the NIC course at MIT on Tuesday, October 12, 1971. Please skim through it before the course. 3

The course will begin at 9:00 A.M. in the second floor console area of 545 Technology Square. There will be signs, directing you to the room. The course will last until about 5:00 P.M. There will be twelve people attending, so the IMLAC consoles (at least 4) will have to be shared. 4

The goal of the course is to impart a basic understanding in the use of NLS and the Journal. John Melvin and myself will be doing the instruction. We are looking forward to both the course and this Network experiment. 5

NETWORK WORKING GROUP
Request for Comments #221
NIC 7612
Categories: A.5, O.7
Obsoletes: RFC 196, NIC 7141

Richard W. Watson
SRI-ARC
25 August 1971 1

A MAIL BOX PROTOCOL, VERSION-2 2

INTRODUCTION 3

Initial reaction to RFC 196, "A Mail Box Protocol", NIC(7141,) indicates general agreement on the need for such a mechanism. The conventions suggested in RFC 196 assumed only the use of the Data Transfer Protocol (in NIC 7104) in order to simplify an initial implementation. The valid argument, we believe, has been made that sites will also implement the File Transfer Protocol and that as much as possible the Mail Box Protocol should be a subset of it. This version is in answer to this suggestion. 3a

The purpose of a mail box protocol is to provide at each site a standard mechanism to receive sequential files for immediate or deferred printing or other uses. The files for deferred printing would probably be stored in intermediate disk files, although details of how a file is handled, stored, manipulated, or printed at a site are not the concern of this protocol. 3b

A mail box, as we see it, is simply a write only (from the Network) sequential file to which messages and documents are appended, separated by an appropriate site dependent code. 3c

It is also assumed that there would be a program at the sending site which sends the file in the format given below with the optional control codes when appropriate. This program could probably be accessed as a subcommand of the Telnet program. 3d

The motivation for developing this protocol is the Network Information Center's (NIC) need to be able to deliver messages and documents to remote sites, and to be able to receive documents for cataloging, redistribution, and other purposes from remote sites without having to know the details of path name conventions and file system commands at each site. Multiple mail boxes (256) are allowed at each site and are identified as described below. The default is mail box number 0 for use with the standard mail printer defined below. 3e

The only place where the Mail Box Protocol has a potential conflict with the File Transfer Protocol is in file naming conventions. The File Transfer Protocol assumes that the using site will use a filename which follows the access and file path name conventions of the serving site and that this information would be supplied by the user. In the Mail Box protocol we would like not to have to explicitly know the path name conventions at each site. 3f

In other words there is a need for a network virtual pathname convention. We did not want to solve this problem in general at this time and in RFC 196, NIC 7141, proposed the use of a separate socket for mail type delivery and the use of an integer 0-127 to specify the address of a specific file (Mail Box) to be appended to as the simplest form of network-wide standard file name convention for an initial implementation. 3g

To follow more closely the spirit of the File Transfer Protocol, I would now recommend the Append Request be specifically used and that the standard socket agreed on for use with the File Transfer Protocol also be used. Following the byte indicating an Append request, there would be a standard agreed-upon string of letters followed by a number, indicating that this is a mail box append request. A suggested name string would be NETMAIL#, where # is a byte interpreted as a mail box number 0-255. If the above suggested Mail Box file naming convention is unsuitable and some other network-wide standard mail box naming can be agreed on, then it can be used. Please let me know how you feel about this naming convention. 3h

Given agreement on a standard mail box pathname, then the Mail Box Protocol can utilize a subset of the File Transfer Protocol conventions to be given below. 3i

The other problem which was raised about the Mail Box Protocol was the possibility of someone accidentally or deliberately flooding the printer of a site with garbage, as there are no access or file size controls. Some thinking and discussions of this problem have yielded no simple satisfactory solutions. I would recommend initial implementations without standard special safeguards in this area. Safeguards would be a site-dependent option. Standard safeguards for the above problem can be easily added later if they really prove necessary and satisfactory ones can be agreed on. 3j

MAIL BOX PROTOCOL - VERSION 2 4

The Mail Box Protocol will use established network conventions, specifically the Network Control Program, Initial Connection Protocol, Data Transfer Protocol, and File Transfer Protocol (as described in Current Network Protocols, NIC 7104). 4a

The normal transmission for Mail Box 0 is to be Network ASCII. 4b

The standard receiving mail printer for mail box number 0 is assumed to have a print line 72 characters wide, and a page of 66 lines. The new line convention will be carriage return (Hex 'OD'), (Octal '015') followed by line feed (Hex'OA')(Octal 012') as per the Telnet Protocol, RFC 158, NIC 6768. The standard printer will accept form feed (Hex'OC')(Octal'014') as meaning move paper to the top of a new page. 4c

It is the sender's responsibility to control the length of the print line and page. If more than 72 characters per line are sent, or if more than 66 lines are sent without a form feed, then the receiving site can handle these situations as appropriate for them. These conventions can be changed by control codes as described below. 4d

At the head of the message or document sent to mail box number 0 there is to be an initial address string terminated by a form feed. This address string is to contain the sender's name and address, and the receiver's name and address formatted in some reasonable, easy-to-read form for a clerk to read and distribute. Comments could also be included in the address string. 4e

The format of information in mail boxes other than mail box number 0 is not explicitly defined by this protocol. 4f

Initial Connection 5

Initial Connection will be as per the Official Initial Connection Protocol, Document #2, NIC 7101, to the standard File Transfer socket not yet assigned. A candidate socket number, socket #3, has been suggested. 5a

File Transfer 6

The mail item (file) to be transferred would be transferred according to the File Transfer Protocol. 6a

As per the File Transfer Protocol, a file (mail item) can be

sent in more than one data transaction as defined in the Data Transfer Protocol. End of file is indicated by the file separator (as defined in Data Transfer Protocol) or by closing the connection. 6b

Order of Transactions 7

The only basic operation required is an append. 7a

Append Request 7b

(Mailer) User ----------> Server (Mail Box) 7c

<File - data> 7d

----------> 7e

End of File indication 7f

----------> 7g

Acknowledge 7h

<---------- 7i

The data type default is network ASCII. The standard line printer default is as defined above. Other control transactions can be used. 7j

CONTROL TRANSACTIONS TO BE USED 8

OP CODE

Hex	Octal	
00	000	Change data type identifier
09	011	Error or unsuccessful terminate
0A	012	Acknowledge or successful terminate
0B	013	Append request (add to existing file)
5A	132	Change printer control settings 8a

DATA TYPE CODES 9

All data types of the File Transfer Protocol can be used for special applications. For Mail Box 0, default is 8 bit bytes of Network ASCII characters. 9a

ERROR CODES 10

All error codes defined in the File Transfer Protocol could be returned. 10a

PRINTER CONTROL CODES 11

Hex	Octal		
01	321	Meaning:	Set line width to 72 characters
02	322	Meaning:	Use the full width of your printer
03	323	Meaning:	Set page size to 66 lines
04	324	Meaning:	Set page size to infinite 11a

Other virtual printer control codes can be added in the future. Other classes of control codes can be added as the need arises. 11b

<JOURNAL>7612.NLS;1, 27-AUG-71 10:41 RWW ; (Expedite) Title:
Author(s): Richard W. Watson/RWW; Distribution: SDC2 TFL JWM JFH REL
AOJO JEW AWH DLM PWF RAW HRVZ AAM RLS JMM JMW AKB PMK TNP ASL BMW JAM
EAF RTB JMP BDW JTM JCL AJB CDS RFH EMA;/NWG; Sub-Collections: NWG ARC
NIC; RFC# 221; Clerk: RWW;
Origin: <WATSON>MAIL.NLS;4, 27-AUG-71 9:51 RWW ;

RWW 9-SEP-71 12:08 7632

Requirements and Design Outline for a Stage 1 Master Catalog Entry and Production System

Please pick up figure 1 from Cindy

REQUIREMENTS AND DESIGN OUTLINE FOR A STAGE 1 MASTER CATALOG ENTRY AND NIC CATALOG PRODUCTION SYSTEM 1

Stage 1 is to function until: 1a

(1) A set system has been designed, implemented and demonstrated to work reasonably efficiently. 1a1

(2) The new data structure of NLS has been clearly specified and implemented. 1a2

(3) The Master Catalog fields have been evaluated and redesigned in light of (2) above. 1a3

(4) A new Master Catalog structure design making effective use of (2) above and (5) and (6) below has been accomplished. 1a4

(5) An archival file system has been designed, implemented and demonstrated to work operationally. 1a5

(6) We have a better feeling about what characteristics integrated data management facilities should be for formatted files in NLS. 1a6

(7) The Journal has been modified to work with the Master Catalog directly (i.e., the Journal Catalog is to be phased out of existence). 1a7

When the above are accomplished, we can design a Stage 2 Master Catalog System. That point in time is probably 6-9 months away, although individual tasks above will happen sooner. It may prove desireable to have some evolutionary stages between Stage 1 and Stage 2, but that remains to be considered as the above tasks are accomplished. 1b

The system is to be split as cleanly as possible into two parts 1c

(1) A Master Catalog entry and update phase. 1c1

(2) A subcollection listing and index production phase. 1c2

The system will have most of the features of a general purpose data management system. This potential generality should be made as explicit as possible in the final design of the system and its modules and their implementation so that other applications can use the work and experience gained here. The

main difference between what we are doing and a true, generalized data management system is that it allows a file data structure to be defined made up of basic entities and then carry, either in a directory or at the head of the file, a table of file structures and element names which is used by the generalized transaction, retrieval and formatting programs. What we have are programs for one structure, namely, the Master Catalog. There is a large bulk of applications that can use even the simple approach of the Master Catalog, and people should be strongly encouraged to do so, unless there are very good reasons otherwise, so their work can use our past efforts and their new contributions can be used more easily by others. We seem too frequently to reinvent the wheel. 1d

Just as the merge, sort, update functions are being implemented as low level primitives for L-10, higher level general functions should be created, and only at the highest level should NIC or Master Catalog specific records be frozen in the programs. 1e

Entry Requirements 1f

Refer to Figure 1. Programs needing building are shown as circles. 1f1

Entry to the Master Catalog is to come from two sources: 1f2

(1) The Journal 1f2a

(2) An ARC typist 1f2b

Later we may or may not deviate to allow others to enter records to the Master Catalog through an input program. 1f3

The Journal presently creates its own catalog in a different format from the Master Catalog. With the addition to the Journal of the ability to indicate documents updated or obsoleted and to handle Functional Documents, entries, probably in a separate branch, will be made indicating which other catalog records need to be updated. 1f4

For entry in the Master Catalog, a program is needed to convert entries and edit transactions in Journal Catalog format to those required for the Master Catalog. This program is only to convert those entries which have been added since the last time it has been run. This date

should probably be setable by a program or by hand, for proper interface to error recovery procedures. This program can insert full entries into, say, Branch 1 and edit transactions into, say, Branch 2, of a given destination file. 1f5

The entries coming from a typist need to be proofed, so we assume they go initially to a work file and then can be entered to appropriate branches of the destination file mentioned above. 1f6

The edit transactions need to be processed to create new entries. The Master Catalog allows records (statements) to have fields and one level of subfield. Fields and subfields are delimited with "field name" character strings. The basic functions required are: 1f7

(1) Delete catalog record 1f7a

(2) Add field or subfield 1f7b

(3) Delete field or subfield 1f7c

(4) Replace field or subfield 1f7d

The transaction processor should probably be implemented in two levels. One level as a primitive which can be called by L-10 like the merge, sort, and update functions and the second level for the application. 1f8

The primitives would take as an argument a statement ID and some number of strings which could be interpreted as field and subfield delimiters or names, as well as any other strings required, and perform the named basic operation. The problem is defining end of field. In the Master Catalog end of field is defined by the start of the next field. Maybe we need a standard end of field symbol such as EOL. 1f9

Question -- how does the above fit in with DEX? 1f10

In a way we can almost, except for the end of field, do the above now in TNLS, for example, replacing the #1 subfield of the *a1 field of record M5012 would be

```
r t .M5012 [*a1][#1] CA .M5012
[*a1][#1][<end of field>] CA y LIT CA
```

This is awkward for people, but reasonable for machines. 1f11

The transaction processor would assume each edit transaction is a statement in a named plex and copy the catalog entry to be edited to a given place and perform the edit. 1f12

Merging and updating back to the Master Catalog is most safely handled as a separate step. Entry deletion is an exception to the above. 1f13

A merge-update processor, with any presorting required, is then required to take the contents of a named file and enter them into the Master Catalog. Items which are updating entries in the catalog replace these entries based on signature date. In proofing and recovery purposes old-new update pairs are also added to a named file. 1f14

The above programs run, on some periodic basis, say, daily at night in the appropriate order automatically or manually under control of a sequencing program. After the Master Catalog has been updated, the input file version number is incremented to create a new blank file. 1f15

We need to take care in the final design to be sure we can back up and recover from various disasters to which we seem subject. The goal is a system secure enough so that once a person has entered a Journal item or a typist has entered a record they can know that the entry will reach the Master Catalog. 1f16

CATALOG PRODUCTION 2

There are initially two major subcollections: ARC and NIC. These are large enough so that the savings in hand proofing time and machine sorting and formatting time warrant building catalogs on an incremental basis. Two programs driven by the sequencer above will collect and merge or update new NIC or ARC items to files of incremental material. 2a

On a roughly monthly basis a Catalog Production program will produce a set of listings and indices. For details of present procedures, see (7263,). 2b

Roughly what is presently involved is: 2c

(1) Run the programs which get NIC subcollections to be indexed, presently NIC NWG/RFC and NGG into separate files. 2c1

(2) Build author, titleword-keyword, number, RFC, keys, and sort. 2c2

(3) Run formatter programs. 2c3

(4) Insert output processor directives. 2c4

(5) Run output Processor. 2c5

(6) Print. 2c6

The indices and listings will be proofed, using NLS. Errors will be corrected in the formatted files. Those in the Master Catalog will be fed back to a typist for reinput or edit input. 2d

The incremental listings and indices will then be merged, based on the full set of sort keys for the items, to form the latest version of a NIC catalog. 2e

With reasonable care, the Catalog Production program should be simply modified to handle a variety of subcollections, possibly with different key-making or formatting programs. 2f

INTRODUCTION 1

Although we have recently started using ARC's Baseline Planning System on a current task-by-task basis, we really need a more complete, "higher level" picture of what new ARC system developments (functions, features, stages..) we want and expect to see. Among other considerations, this brings up the subject of goals: what are we working toward? 1a

GOALS 2

Our overall goals have been the subject of many discussions in the past, usually with the agreement that "we should make ARC's goals (or goal structure) more clear" so we can better decide what to build, how to use our resources, and so on. 2a

Somehow our goals never seem to be made more clear, perhaps due to the complex relationships between various goal levels and the assortment of needs, possibilities, activities, tasks that exist...and that in most cases change constantly and in subtle ways ... or perhaps due to our not just putting down what we think they are and letting them be changed into more realistic terms by ideas from many people. 2b

It appears that the key goals we (ARC) are presently working toward are the following: 2c

1. We want to see and participate in continued (and better balanced) evolution of the ARC augmentation system -- tools, methods, languages, training -- all aspects. 2c1

2. We want increasingly more non-ARC users to integrate our augmentation system (or selected elements) into their daily work. 2c2

3. We want to establish relationships with other groups that will build improved augmentation systems or components of them, thereby increasing their rate of development and use. 2c3

These ARC goals fit into a set of even higher level goals as part of Doug Engelbart's developing picture of the needs and possibilities for augmentation systems and their ultimate meaning to man and his institutions. 2d

STRATEGIES 3

We will use many strategies in working toward the above goals. Some key ones are: 3a

Bootstrapping (an element in other strategies) 3a1

Integrating mainly system builders as the "next" users 3a2

Participation in the ARPA network 3a3

Modularizing NLS (and other sub-systems?) 3a4

Broadening support (even at some promotion cost) 3a5

a. To involve more people and organizations 3a5a

b. To ensure stable funding situation 3a5b

Using our resources more effectively. (system, people time and skills) through development and use of DEX, DPCS, PBMS etc. 3a6

Improving our management, information gathering, and documentation systems and practices 3a7

Learning more about how to provide augmentation system service 3a8

COMMENTS ON EMPHASIS 4

Up to this point, ARC has invested heavily in computer techniques and tools and we plan to continue to do so. However, the current proposed ARC activities (outlined below) require specific "total" augmentation sub-systems to be developed. 4a

We must aim toward plans for such a balanced and well documented development in each area that we can actually deliver marketable entities. 4a1

Accordingly, we will need to know how to assist other groups (through training and documentation) to easily replicate and use the functions we develop. 4a1a

Another underlying theme is our continuing effort on team augmentation. 4b

A major part of team augmentation is good communication: with useful dialog in face to face, on-line and hardcopy modes. 4b1

As we grow larger and as we interact more with people outside of ARC, the need for better documenting of our basic tools and how to use them in application-oriented tasks will increase. 4b2

We need to get in the habit of writing more, even though for some of us it seems difficult and unrewarding; this current planning effort provides a good starting base for pulling together outlines of what we do know about building augmentation systems for the proposed activities. 4b3

Until now, we have been dependent mainly upon one financial source. As a long term strategy, this is a risky situation even assuming continuing support and may not be the best strategy for us to meet our goals of actually seeing our developments being widely used in the world. 4c

Therefore, aiming toward the development of deliverable, integrated products as well as continuing technique and tool development while broadening our support base seems important to ARC's long range health. 4c1

PROPOSAL AS A START 5

Our recent proposal (7404,3a) outlines our basic approach for working toward our goals, where we see four main project activity areas. ARC non-project (overhead) activities constitute a fifth area. 5a

The current and proposed basic ARC activities are: 5a1

1. Develop new service functions 5a1a

   NIC (Network Information Center) 5a1a1

   DSS (Dialog Support System) 5a1a2

   DPCS (Documentation Production and Control System) 5a1a3

   SEAS (Software-Engineering Augmentation System) 5a1a4

   Handbook 5a1a5

   PBMS (Project Baseline Management System) 5a1a6

We assume that each activity plan will require many days of effort on the part of the pusher -- particularly with the balancing (adjusting) that may be needed. This may require as much as 1/3 time from some people over the next six weeks or so. 7b

This planning work deserves a high priority now. 7c

We are presented with many intriguing possibilities at this time -- ranging from possible NLS feature additions, to re-writing NLS in MPL, to Xerox, NASA Ames, RADC, and other external collaboration, to establishment of other new working relationships with organizations and people. 7c1

Until we decide more specifically what we want to do (build, etc), how we might use the resources we have ($, system, people), we cannot decide what we can or can't do with those resources or what added resources we should seek if we decide to proceed with added projects or activities. 7c2

So -- here's the list of activities and pushers we are starting with in requesting a "30 month" development plan -- first try. 7d

1. Service functions: Pusher: 7d1

   NIC .....Dick Watson 7d1a

   DSS .....Bill Duvall 7d1b

   DPCS .....Doug Engelbart 7d1c

   SEAS .....Charles Irby 7d1d

   Handbook .....Doug Engelbart 7d1e

   PBMS .....Jim Norton 7d1f

   CSES .....Bill Paxton 7d1g

   RINS .....Doug Engelbart 7d1h

Other plans that will be started shortly will cover: 7d2

2. Service Delivery Techniques .....Dick Watson 7d3

3. Service Marketing Techniques .....Jim Norton 7d4

4. Operational Services 7d5

Facility .....Ed Vanderiet 7d5a

NIC.....Dick Watson 7d5b

Other.....?? 7d5c

Plus activities not part of our proposal: 7d6

5. ARC overhead functions .....Jim Norton 7d7

6. BC* framework and kickoff .....Doug Engelbart 7d8

* BC = Bootstrap Community, as Doug has discussed (over many months) as being part of his next strategic push toward his longer term goals

7d8a

PLAN ELEMENTS 8

The following eight items are basic considerations each pusher should provide in his plan: 8a

1. Basic objectives of the activity as you see them. 8a1

What should it result in or produce? 8a1a

2. New or changed features that may be added...including descriptions of what they are, how they might work, what they mean to the system and/or the users 8a2

Use a separate branch (+dialog?) for each. 8a2a

These may be thought of either as separate tasks, or simply as "features" -- which might result from several tasks. 8a2b

3. All the non-machine methodology, procedures, and training that need development to really use the tools and features to produce useful total packages -- sub-systems. 8a3

4. Stages of development... logical combinations of features, procedures, training (not just points in time)...describing the "look" at significant points. 8a4

The stages should be activity-related -- not necessarily related to ARC overall stages (we can work those out later). 8a4a

Refer to stages by number: stage 1, stage 2, etc. 8a4b

Some activities will have less apparent need for showing stages of development than others. Still, it seems it is important to "partition" the future plan in some way, even if on an arbitrary, less meaningful basis. 8a4c

5. Relationships to other tasks or features needed 8a5

Where critical needs (for this activity) exist, they should be pointed out -- with some discussion of the situation. 8a5a

These should be discussed either with relevant feature descriptions, or as "item 5.", as a separate, more general topic for the activity as a whole. 8a5b

6. Effort needed to meet stages. 8a6

ROUGH estimates in man-weeks by feature or stage (+ types or who in mind if important? -- but we will do this later in most cases) 8a6a

7. Alternative possibilities for other features/stages/whatever 8a7

8. Implications on the staffing skills and levels required of ARC as a whole. 8a8

Dick Watson and I have done some preliminary looking at the whole picture -- features and people -- and have made rough notes (7653,) about each activity just to start some ideas flowing. We expect that each activity's pusher will greatly extend (and perhaps obliterate) our simple start. We will provide pushers with our notes though. 8b

We would rather have "too much" time spent on filling out the details of well constructed activity "framework pictures" than too little. 8c

Also, it would be worth reviewing our proposal (7404,1) and its related parts as a start. 8d

WHEN PLANS ARE NEEDED 9

It's hard to tell how much time is going to be needed for this full plan set to be made. We would like to have at least a draft outline from each pusher on: 9a

Friday, Sept. 17th (this week). 9a1

Then, we would like to have a more complete "first" plan from each pusher on 9b

Friday, Sept 24th (1 week later). 9b1

Then, we would like to have a reasonably complete "first" plan from each pusher on 9c

Friday, October 1st (2 weeks later). 9c1

If the Journal is up, please submit your plans that way (even in rough draft stage). If it is not up, then just give me hard copy and let me know the filename(s), with the assumption that they will go into the Journal when possible. 9d

NEXT STEPS IN THE PROCESS 10

After the initial plans are made, we will combine and "publish" a complete set, so all of ARC can see, consider, and comment. 10a

Then we will begin the process of adusting, changing (by pushers) all plans so that a realistic combined set (baseline) emerges. 10b

This continuing process will be part of our overall ARC Baseline Planning System. Changes to the plans will be recorded in the Baseline Record (in ways yet to be developed.) 10c

As activity stages come nearer in time, their task needs will become more clear through dialog, requirement specifications, designs, and documentation of other tasks. 10d

Therefore, new task agreements will be made, while other tasks finish, all within the Baseline Planning (and Record) system...even as the system itself evolves. 10d1

We are hoping, of course, that the current set of ongoing tasks is consistent with the Baseline picture we will develop. However, where task needs appear new, we can add -- where less need appears for current tasks (or effort levels) we can also adjust. 10e

If any of you have other ideas about this process -- we want them. Don't wait for Journal availability to tell. 11

(J7634) ; Title: Author(s): James C. Norton/JCN; Distribution: Douglas C. Engelbart, Richard W. Watson, William S. Duvall, William H. Paxton, Ed K. Van De Riet, Charles H. Irby/DCE RWW WSD WHP EKV CHI;
Sub-Collections: SRI-ARC; Clerk: JCN;
Origin: <NORTON>J7634.NLS;2, 18-OCT-71 8:37 JCN ; .RTJ=0; .LSP=0
18-OCT-71 8:48;

7636 1-OCT-71 ARC
INTRODUCTION

Section 1. INTRODUCTION

1

ARC DIALOG SUPPORT SYSTEM 1a

The Dialog Support System (DSS) is one of several key augmentation functions under development by the Augmentation Research Center. The DSS includes techniques to aid effective collaboration among distributed parties by means of inter-linked referencing between NLS files, particularly within the recorded-dialog medium of an NLS Journal. 1a1

The goal of DSS is the effective augmentation of collaborative teams by having an "intragroup documentation system," containing current and thoroughly used working records of the group's plans, designs, notes, etc. The Journal is the first step in an evolving intragroup documentation system for entering and managing those records. 1a2

Many DSS features will be made available to NIC users. This user guide is intended to help integrate the first of these DSS techniques into the NIC community. It includes discussions of three separate NLS subsystems of importance to the NIC user interested in using the Journal: the Journal System, the Identification System, and the Number System. 1a3

THE JOURNAL 1b

The Journal is an open-ended information storage and retrieval system intended to facilitate communication and the availability of documentation within the NLS user group. The Journal System consists of a set of procedures which enable the automatic cataloging and distribution of information items (messages and documents) within the group. 1b1

JOURNAL ENTRIES 1c

There are two types of entries that can be made to the Journal: messages and documents. From the sender's viewpoint, there is no distinction between the two; it is a distinction made by the system to facilitate processing these items. 1c1

A message can either be a single statement from an existing file, or literal text entered from the terminal when the Journal entry is created. This is the simplest form of Journal activity requiring no knowledge of NLS, only some basic Journal procedures. A sample session, showing how to send a message through the Journal is described below. 1c1a

A document may be a branch, group, or plex of an existing file, or an entire file. 1c1b

If the Journal entry is a document or an existing statement that will be used as a message, the file containing that entry must be loaded into NLS before the Journal procedure is begun by the user. 1c2

JOURNAL ENTRY REFERENCES 1d

One of the principal techniques utilized by the Journal is the cross-file referencing enabled through the use of "links". A link is a reference as well as an (on-line) vehicle to another file or Journal entry. The system uses links to find, deliver, and catalog Journal entries. The catalogs that are built from current Journal entries rely totally upon links for automatic management of cross-reference citations. Similarly, the user is encouraged to use links to pertinent documentation wherever possible in a Journal entry. For example, a Journal entry that references another Journal entry should include a link to the latter in order to facilitate on-line access to that document for other users. 1d1

Links to Journal entries usually take the following form: 1d2

(CATNUM,ADDRESS:VIEWSPECS) 1d2a

Where CATNUM = the catalog number assigned to the Journal entry. 1d3

ADDRESS = a specific location in the referenced Journal entry. (If not specified the beginning of the file is assumed.) 1d4

VIEWSPECS = an optional set of character codes that control the "view" of the entry. 1d5

Link specifications must be enclosed in parentheses, anglebrackets (<>) or preceded by two dashes (--) and terminated by a right parenthesis or a right anglebracket. 1d6

The link specification parameters are described in full detail in the NIC TNLS User Guide (see -- 7470,). 1d7

SAMPLE MESSAGE SENDING SESSION 1e

The following is a demonstration of how a message is submitted to the Journal by a Network user (including login, NLS access, and logout procedure). 1e1

```
login SP DOE SP DDD SP 1 CR   (a user named DOE logs in
                               to the system - his ID
                               is DDD)                    1e2

[JOB11 ON TTY14   3-AUG-71  17:11]                        1e3

@nls CR    (accesses NLS system)
[ID:] DDD CA
[device:]  N[et-tty]  ("N" signifies that the user DDD is
                       connected through the Network)     1e4

*e[xecute Journal]  (access Journal system)               1e5

[submit] m[essage]                                        1e6

THIS IS A SAMPLE MESSAGE CA  (contents of message)        1e7

[number] CA [3333]  (typing CA after request for number
                     causes system to assign DDD a unique
                     catalog number [3333] for the message) 1e8

& i[nterrogate] CA  (causes the system to prompt the user
                     for the correct information required
                     to send the item through the Journal) 1e9

&[title:] A SAMPLE JOURNAL SESSION CA                     1e10
```

```
&[distribution] XXX(for your action) YYY(for your
information) CA
            (XXX and YYY identify other persons known to
             the system; the text in parentheses are
             comments directed to them.)                    1e11

&[status] CA                                                1e12

  .
  .   (the system reiterates all information entered by the
  .    user)
  .                                                         1e13

&[go:] CD  (the user does not want to "go" now but add more
            information not requested by the Interrogate
            procedure)                                      1e14

&k[eywords:] test sample CA  (keywords provide input to an
                              index to all messages)        1e15

&c[omments:] isn't this fun  CA                             1e16

&g[o:] CA  (tells the system to begin Journal process.
            Note that as the author command was not
            used, DDD is assumed)                           1e17

[JOURNAL SYSTEM IN PROGRESS]  (the system is processing
                               the new Journal entry)       1e18

   NOTE: The time required to process a Journal entry
   ranges from a minimum of about 30 seconds to three
   minutes depending on the current system load.  During
   this time the system will periodically issue RUBOUT
   impulses (causing the terminal print head to "jump"
   without printing) until the Journal process is finished
   and the system prints the link created by the process
   and issues the NLS herald character "*".             1e18a

[Journal,JRN1,J333:gw]  (the system has created a link
                         to the new Journal entry)          1e19

*e[xecute] q[uit]  (the user has been returned to the NLS
                    command level as signified by the
                    system typing the herald character
                    "*"; the user then types the Execute
                    Quit command to return to the EXEC)     1e20

@logout  CR       (the user logs out)                       1e21
```

IDENTFILE 1f

Some of the key concepts of the Identification System are described here because they are necessary for initial understanding and use of Journal Entry procedures. 1f1

Each of the systems described in this user guide requires IDENT as its prime person identification parameter. IDENTs are processed in the same manner throughout NLS; therefore any procedures described for IDENT here are applicable to the Journal, Identification, and Number Systems. 1f2

When a new person is introduced to the system, an entry is made for him in a file (IDENTFILE) containing all necessary information about individual users. When this entry is made the user is assigned an identification word -- his IDENT. The IDENT must be used subsequently whenever he or another user wishes to identify him to the system. 1f3

The following are the names of the principal fields of information maintained about each person in the IDENTFILE. 1f4

IDENT - 1g

the primary means of user identification. It may consist of two to six alphanumeric characters for an individual and from two to twelve characters plus the characters "-" and "'" for a group (defined below). The first character of an IDENT must be alphabetic. The IDENT typically consists of the person's or group's initials. In the case of multiple persons or groups with the same initials, IDENT is appended by an integer number (not less than a value of 2). 1g1

IDENT may be used to designate an individual or a group. In the latter case IDENT may represent a set of other IDENTs (i.e. the IDENTs of the members of the group). 1g2

Whenever the syntax of the commands described under the Journal, Identification, and Number Systems requires IDENT, the user may respond with one of the following: 1g3

a literal which is interpreted as IDENT 1g3a

a CA which indicates the IDENT used to enter NLS 1g3b

a period followed by a text string (which causes the system to follow the procedure described below) 1g3c

a question mark (?) followed by a text string enclosed in square brackets or followed by a single alphabetic character (which causes the system to follow the procedure described later in this section under Journal System Conventions) 1g3d

Some of the commands allow multiple IDENTs or an IDENTLIST. The correct syntax for an IDENTLIST is a list of IDENTs separated by spaces (e.g.,JCN MFA HGL WSD). When the user enters an IDENT that the system does not recognize, the system prints a question mark after the terminating space and awaits a valid IDENT entry. Also, where comments are allowed in an IDENTLIST (see individual commands), the comment itself must be enclosed in parentheses and immediately follow the IDENT to which it applies. Thus, a question mark signifying that the system does not recognise an IDENT will not appear until the user types a space following the IDENT and comment. 1g4

There are places in the procedures described in this section where IDENT is a valid substitute for other parameters, e.g. IDENT may be used as ADDRESS for groups; the system actually supplies the necessary information. These cases are noted where appropriate. 1g5

In cases where a user is attempting to identify another person to the system, but does not know that person's valid IDENT, the procedures described under Journal System Conventions, the period (.) and question mark (?) characters, may be used. These procedures for user identification may be used whenever the IDENT parameter is required, e.g. for addressee, author, etc. 1g6

The default value for the IDENT parameter is, in all cases, the IDENT used by the current user when he entered NLS. Thus, the user may respond to a request for an IDENT simply by typing CA to indicate himself. 1g7

GROUP - 1h

the IDENT of a set of IDENTS. In most cases, the IDENT of a group may used be in the same way as the IDENT of an individual. However, the entry for a group in the IDENTFILE requires additional information not required for individuals. See the COORDINATOR identifier described below. 1h1

In cases where a user is attempting to identify a group to the system, but does not know that group's valid IDENT, the procedure described under Journal System Conventions, the question mark (?) character, may be used. 1h1a

Groups may contain other groups as well as individuals and as such may be nested to any depth. 1h2

ADDRESS - 1i

the address of the person or group represented by IDENT. For individual IDENTs, the system automatically supplies the mailing address of the user's professional affiliation (see the AFFILIATION field). However, if the user is un-affiliated or wishes to use an address other than that of his affiliation, his IDENTFILE record must contain an ADDRESS field, a conventional mailing address as it would appear on a mailing envelope (i.e., STREET NUMBER SP STREET NAME CR CITY comma STATE SP ZIP CODE). For groups, ADDRESS may be expressed as the IDENT of the group coordinator in which case the system supplies the address of the coordinator (see the COORDINATOR identifier). 1i1

AFFILIATION - 1j

the IDENT of the professional affiliation of the individual. AFFILIATION should indicate the professional entity with which the individual's activities are based, e.g., SRI-ARC and MIT-MULTICS. The IDENTs for affiliations are mnemonics that may be assigned by NIC only; users must use these identifiers exactly as they appear in the current list of affiliations in Appendix D of this user guide. For new affiliations see the procedure described below. 1j1

Where AFFILIATION is a required parameter and the user knows the proper IDENT, the user must type the entire mnemonic. 1j1a

In cases where a user is attempting to identify an affiliation to the system, but does not know that affiliation's valid IDENT, the procedure described under Journal System Conventions, question mark (?) character, may be used. 1j1b

If the affiliation is new, the user must type a slash character (/) and supply its proper name when asked for affiliation. The system then asks the user for the address of the affiliation and assigns a temporary IDENT. A permanent IDENT will be assigned to the affiliation later, when the NIC clerk at ARC verifies new IDENTFILE entries. The user who entered the new affiliation will be informed of the verified mnemonic by the NIC clerk at ARC. 1j1c

Every user must be associated with at least one affiliation. If the user does not specify a professional affiliation, the system assumes an affiliation of "IND"ic (independent). In cases of multiple affiliations, the first affiliation specified for a person is considered his primary affiliation. 1j2

AFFILIATION is used by default as ADDRESS for individuals. unless the individual's affiliation is IND in which case a mailing address is required. In cases where a person has multiple affiliations, for the primary affiliation and is used for his mailing address. 1j3

COORDINATOR - 1k

the IDENT of the person who is responsible for a group. COORDINATOR is used in a manner similar to that of AFFILIATION for individuals. 1k1

COORDINATOR is considered synonymous with the creator of a group unless specified otherwise. Only the coordinator of a group (or a user designated by him) has control over establishing or changing the membership of the group. 1k2

The coordinator's IDENT is used by default for the ADDRESS of a group. 1k3

JOURNAL SYSTEM CONVENTIONS 1l

These characters are used as follows in the Journal, Identification, and Number Systems: 1l1

Character	Meaning	1l2
CA	synonymous with typing "y[es]" in response to system queries.	1l3
SP	synonymous with typing "n[o]" in response to system queries.	1l4
.	(pertains to individual IDENTs only) when immediately followed by a text string where IDENT or IDENTLIST is required, causes the system to treat that string as a last name and search the IDENTFILE for all occurrences of that string. The system then prints a complete list of all persons of that name known to the IDENTFILE along with their IDENTs.	1l5

The user is then asked by the system whether a new IDENT is to be entered, i.e. the person that the user intended is not in the IDENTFILE. If the user responds by typing yes (or CA) he is placed in the Identification system wherein he may enter a new person to the system. After he completes the Identification system procedure the system automatically supplies the new IDENT as required when the user typed ".lastname". The user may then proceed to enter other IDENTs or terminate the command as appropriate. 1l6

If the person intended by the user does appear in the list the user should type no (or SP) whereby the system inquires whether the user wants to continue entering names in IDENTLIST. If yes (or CA) the system expects the user to enter another IDENT; if no, the IDENTLIST is considered terminated. 1l7

? (pertains to all IDENTs)
causes the system to print a complete list of identifiers of the type required by the field in which this character is typed. ? should be followed by a single alphabetic character or a literal enclosed in square brackets. The former for groups causes the system to search for all identifiers that begin with the letter specified; for individuals the system searches for all last names that begin with that character. 118

The latter causes the system to print all occurrences of the literal specified for that field. Thus, if the user typed "?[Washington]" in response to a system request for affiliation, the system would print a list of IDENTS to the system whose proper name contains the literal "Washington". Again, for individuals, the search for the text string is confined to last names. 119

The user is then asked by the system whether a new IDENT is to be entered, i.e. the person, group, or affiliation that the user intended is not in the IDENTFILE. If the user responds by typing yes (or CA) he is placed in the Identification system wherein he may enter a new person, group, or affiliation to the system. After he completes the Identification system procedure the system automatically supplies the new IDENT as required when the user tvped "?". The user may then proceed to enter other IDENTs or terminate the command as appropriate. 1110

If the person, group, or affiliation intended by the user does appear in the list the user should type no (or SP) whereby the system inquires whether the user wants to continue entering IDENTs. If yes or (CA) the system expects the user to enter another IDENT; if no, the IDENTLIST is considered terminated. 1111

RUBOUT terminates printing of the list caused by typing "?" as described above. 1112

BW Backspace word may be used as in NLS to delete the last word entered by the user. 1113

BC Backspace Character may be used as in NLS to delete the last character entered by the user. 1114

BS Backspace Statement may be used to delete the current entry and cause the system to generate a Carriage Return/Line Feed. This key is useful when the user is in the process of supplying several fields of information during the course of one command. The user may delete only the current field entry without jeopardizing other entries. 1115

CD Command Delete aborts the current command, deleting all information entered since the last herald character of the system in which the user was operating when he used CD. Thus, the user is always returned to the command level of the current system. 1116

NIC JOURNAL SYSTEM USER GUIDE

1

Augmentation Research Center
Stanford Research Institute
Menlo Park, California 94025

2

CONTENTS

Section 2. JOURNAL SYSTEM

1

CREATING JOURNAL ENTRIES 2

The user enters the Journal submode with the Execute Journal command: 2a

```
e[xecute] j[ournal  CR]
[submit] s[tatement] ADDRESS          CA [CR]
        b[ranch] ADDRESS
        g[roup] ADDRESS CA ADDRESS
        p[lex] ADDRESS
        f[ile]
        m[essage CR] LIT
        SP [CR] LIT
```

2a1

```
[number:] CA [CATNUM]
          CATNUM CA [(assigned to):] IDENT CA
             (if RFCNUM number associated with CATNUM)
             [rfc # RFCNUM]
          r[fc number:] RFCNUM CA [(assigned to):] IDENT CA
                        CA [rfc # RFCNUM CR]
                           [catalog # CATNUM]
```

2a2

NOTE: the process of obtaining a catalog (and RFC) number may take from 30 seconds to one minute depending on the system load. During this time the system will periodically issue RUBOUT impulses (causing the terminal print head to "jump" without printing) until a catalog (and RFC) number is assigned. 2a2a

where ADDRESS = any combination of address elements specifying a structure in a file. 2a3

LIT = any string of characters (message content). When entering LIT, the control characters for editing entries (i.e. BC, BW, etc.) may be used. Note: "m" and "SP" are equivalent parameters. 2a4

CATNUM = if supplied by the user, a pre-assigned Catalog number. (The process of obtaining pre-assigned Catalog numbers is discussed in Section 4, Number System.) If the user does not have a pre-assigned Catalog number he must type CA and the system will supply him with one. If the pre-assigned number is locked due to a system crash before a Journal process was completed using the same CATNUM, the system will print:

[CATNUM in use...proceed?]

to which the user may respond with typing "y[es]" causing the system to unlock the number and use it for the current Journal process. Alternatively, the user may type "n[o]" which causes the Journal processs to be aborted and the user returned to the NLS command level. 2a5

RFCNUM = if supplied by the user, a pre-assigned RFC number. RFC pertains to "Request for Comment" documents which are used by the Network Working Group. (The process of obtaining pre-assigned RFC numbers is discussed in Section 4, Number System.) If the user has a pre-assigned RFC number it is unnecessary to specify a Catalog number here also because when the RFC number was originally assigned the user was also assigned a Catalog number. (The system automatically uses this Catalog number for the Journal process when the user specifies his RFC number.) 2a6

If the pre-assigned number is locked due to a system crash before a Journal process was completed using the same RFCNUM, the system will print:

[RFCNUM in use...proceed?]

to which the user may respond with typing "y[es]" causing the system to unlock the number and use it for the current Journal process. Alternatively, the user may type "n[o]" which causes the Journal processs to be aborted and the user returned to the NLS command level.

2a7

If the user does not have a pre-assigned RFC number, he may obtain one by typing CA immediately after "r". The system responds by printing an RFC number followed by a Catalog number. 2a8

IDENT = the user to whom the RFC/Catalog number is assigned. 2a9

Note that "submit statement" is identical to "submit message" in terms of Journal processing. They differ only in that "submit statement" causes the system to go to the address specified for the text of the message whereas "submit message" requires the user to enter the text from the terminal at the time of submission. 2a10

Once this procedure is completed, the user is at the Journal submode command level indicated by the system typing the herald character "&". 2b

JOURNAL COMMANDS 3

In the Journal submode, the following commands are allowed: 3a

INTERROGATE 3b

This command causes the Journal system to enter a mode whereby the user is interrogated for information. 3b1

i[nterrogate] CA 3b1a

The information which will be requested by the system will be:

```
Title:
Distribution:
Status:
Go?
```

3b2

The user should respond to the requests as follows: 3b3

Title: LIT CA 3b4

Where LIT = any series of characters which constitute the title. 3b4a

This allows the user to enter a title which will be used in the hard copy printouts of the document or catalog listings and indexes. 3b4b

Note that the title will be overwritten by any existing HED directives within the text of the document. 3b4b1

Distribution: IDENTLIST CA 3b5

This command allows the user to specify a list of persons who will receive copies of the document being entered. 3b5a

IDENTLIST may consist of IDENTs of groups as well as individuals. 3b5a1

If the user attempts to enter an identifier that the system does not recognize, the system will print a question mark (?) at the terminal. 3b5a2

The user may insert notes for particular individuals in the list by enclosing a text string in parentheses immediately after (with no intervening blank spaces) the IDENT of the person to whom the comment is directed. (Comments may be addressed to the entire IDENTLIST by using the Comments field, described below.) 3b5b

There are two modes of distribution: expanded and unexpanded. Expanded distribution means that if there are group IDENTs in the IDENTLIST, each member of the group will receive a copy of the Journal item. Alternatively, distribution may be made unexpanded to the group membership as a whole, that is, one copy of the item is sent to the group mailing address. Initially, the default mode is "unexpanded". However, this default may be changed by use of the Expanded Group Reference subcommand of the Modify mode (discussed in Section 3, Identification System). Also, distribution may be controlled when submitting an item to the Journal by prefacing the group name with either of the following characters: 3b5c

$ - overides the default and causes unexpanded distribution 3b5c1

↑ - overides the default and causes expanded distribution 3b5c2

Status: CA 3b6

This causes the system to print the values for the items supplied by the user in response to the above and additionally, other fields that the user may specify using the commands described below. 3b6a

Document Number
RFC number
Author
Title
Distribution
Subcollections
Keywords
Clerk
Comments
Place Link

3b6a1

Go? y[es]
CA
n[0]
SP 3b7

Responding to this prompt with "yes" or CA causes the Journal entry process to begin. 3b7a

Reponding by "no" or SP returns the user to the Journal system command level wherein he may modify and/or add to information entered during the Interrogate session. 3b7b

If the user does not wish to respond to a cerain request during the Interrogate session he may use a Command Delete (CD) annd proceed to the next request. 3c

Instead of using the Interrogate command to cause the system to request the user for each entry item, the following commands may be used to enter each item independently, to change a field previously entered and/or to add additional information not requested by the Interrogate procedure. 3d

AUTHOR 3e

This allows the user to specify the contents of the author field in the Journal header statement and in catalog entries. 3e1

a[uthor(s):] CA
IDENTLIST CA 3e1a

The normal response is CA indicating the IDENT of the username used by the current user when he entered NLS. 3e2

If the author's name is known but not his IDENT, the user may cause the system to search for the correct IDENT by typing the last name of the author preceded by a period according to the procedure described in Section 1, Introduction. 3e2a

TITLE 3f

This command is identical to responding to the Title request under the interrogate command. 3f1

t[itle:] LIT CA 3f1a

Note that the title will be overwritten by any HED directives within the text of the document. 3f2

DISTRIBUTION 3g

This command is identical to responding to the Distribution request under the interrogate command. 3g1

d[istribution:] IDENTLIST CA 3g1a

KEYWORDS 3h

This command allows the user to specify key words which typify the nature of the document. 3h1

k[eywords:] WORDLIST CA 3h1a

Where WORDLIST = a series of text strings separated by spaces. 3h2

These words will be included in a special keyword field in the header which may be used for later content searches for documents. Keywords are also used for appropriate indices of the catalog. 3h3

COMMENTS 3i

This allows a user to enter a comment about the document being entered. The comment appears on the first page of the item sent to each member of the distribution list. 3i1

co[mments:] LIT CA 3i1a

Where LIT = any string of characters 3i2

The text of the comment will be the last thing in the header statement and will be preceded by two CRs. 3i3

Notes may be addressed to particular individuals in the distribution list by enclosing a comment in parentheses immediately after the IDENT of the person for whom the comment is intended. 3i3a

PLACE LINK 3j

This command causes the system to create and place a link to the Journal item in the file specified by ADDRESS. 3j1

p[lace link (if successful)?] CA
y[es]
n[o]
SP 3j1a

(if yes or CA)
[after location:] ADDRESS CA LEVADJ CA 3j1a1

Where ADDRESS = any valid combination of NLS address elements which indicates the location at which the link specification will be placed. Note that a link is a valid address specification which causes the system to place the link to the current Journal entry in the file indicated by the link. 3j2

LEVADJ = a level specification relative to ADDRESS. 3j3

STATUS 3k

As or after the user enters header information, he may review it by causing the system to type out the values currently in the header statement. 3k1

st[atus] CA 3k1a

This command causes the following fields of the header to be printed at the terminal along with their current values. 3k2

Document number
RFC number
Author
Title
Distribution
Subcollections
Keywords
Clerk
Comments
Place link 3k2a

Any fields which are empty are omitted from the printout. 3k3

At this time, the user may replace the items in this list by re-entering any command. 3k4

CLERK 3l

This allows the user to specify the identification of the person who is currently doing the entry of the document. It is assumed (by default) to be the IDENT of the current NLS user. 3l1

cl[erk:] CA
IDENT CA 3l1a

SUBCOLLECTION 3m

When an item is submitted to the Journal, its catalog entry may contain a field indicating a subcollection to which the item is assigned. The subcollection is determined ordinarily by the user's affiliation. For example, the subcollection ARC pertains to all users affiliated with ARC. (Network users have all items automatically entered into the NIC subcollection. RFC's are automatically entered into the NWG and NIC subcollections. Subcollections are also maintained for each group and an entry is made for that subcollection if a group IDENT is used in the Distribution IDENTLIST. However, a Network user may, for example, send a message to a person and automatically be entered in the broad NIC subcollection, but still want an entry in a subcollection.) This command enables the user to add to the default subcollection assignment. 3m1

su[bcollection(s):] IDENTLIST CA 3m1a

Where IDENTLIST = any number of subcollection names (normally group IDENTs). The system maintains a list of recognized subcollection names. Multiple names must be separated by a space. 3m2

Note: The function of this command is initialize the subcollection entry. Therefore, if the current entry is, for example, NIC, and the user wishes to redefine the entry to include other subcollection names, he must reenter "NIC" as a parameter. 3m3

If the user had attached an RFC number of a Journal item, the item will automatically be included in the NIC and NWG subcollections. 3m4

OBSOLETE DOCUMENT 3n

This command enables the user to indicate if there is any Journal item(s) obsoleted by the current Journal item. 3n1

ob[soletes document(s):] CATNUMLIST CA 3n1a

Where CATNUMLIST = one or more catalog numbers separated by spaces 3n2

UPDATE DOCUMENT 3o

This command enables the user to indicate if there is any Journal item(s) updated by the current Journal item. 3o1

u[pdates document(s):] CATNUMLIST CA 3o1a

Where CATNUMLIST = one or more catalog numbers separated by spaces

3o2

QUIT 3p

This command returns the user to the NLS command mode. It should not be necessary except in cases where there has been a system error. 3p1

q[uit] CA 3p1a

GO 3q

This causes the Journal entry process to begin after the user has specified any header information he wishes. 3r

g[o?] y[es]
CA
n[o]
SP 3r1

This command is identical to responding to the Go? request under the Interrogate command. 3s

The actual Journal process may take from 30 seconds to three minutes depending on the current system load. During this time the system will periodically issue RUBOUT impulses (causing the terminal print head to "jump" without printing) until the Journal process is completed and the system prints the link created by the process and issues the NLS herald character "*". 3t

USER STATUS ON LEAVING JOURNAL SYSTEM 3u

After the user has successfully entered an item into the Journal the user is returned to NLS. The current file open is the file being submitted. If the item was a document (i.e. not a message) the Journal entry process will have created a permanent file named CATNUM which may not be further changed; however, it may be viewed, referenced, etc. If the item was a message, it will be added to the Journal message file.

3v

INITIAL JOURNAL DISTRIBUTION 4

Journal messages and documents are currently sent in hardcopy form and, for NLS users, by direct online delivery through the addition of a link to the "Journal" branch in the addressee's NLS initial file. (This branch is created in a user's initial file automatically when any Journal item is sent to him.) 4a

Since these initial files are automatically presented to each user as he enters NLS, messages are easily available to users. New messages are added at the head of the message plex for quicker viewing of the most recent ones. 4b

When the user is through viewing a message, he may delete it (or the statement with the link to it) for it will also be on file under directory JOURNAL in a file with a name such as 7006 (if it is a document) or as a branch in a file named JRNL1, JRNL2, etc. (if it is a short message).

4c

SECONDARY JOURNAL DISTRIBUTION 5

The user may cause a document(s) that was previously entered into the Journal by himself or others to be distributed by using the NLS command Execute Secondary Distribution. 5a

```
e[xecute] se[condary distribution CR]
[document #] CATNUM CA
[to:] IDENTLIST CA
```

5a1

Where CATNUM = the Catalog number associated with the document to be distributed. 5b

When this command is executed, the document specified will be distributed to the persons indicated by IDENTLIST along with regular Journal delivery. 5c

If the user specifies an invalid number, the system prints the message: 5d

ILLEGAL NUMBER 5d1

If the user specifies a valid number for which there is no document, the system prints the message: 5e

NO SUCH DOCUMENT 5e1

It is possible to get the latter message if a document has been entered into the Journal relatively recently. 5f

If this happens, wait a while (at least 30 minutes) and try it again. 5f1

OUTPUT DIRECTIVES AND THE JOURNAL 6

There is a user hazard regarding directives and the Journal worth mentioning here: 6a

Although the Journal system does not tamper with any directives in your file, caution should be exercised that there are no print directives in a file which would contradict those added by the Journal in the origin statement. It is recommended that the user delete or edit them so as not to detract from the format of the hard-copy of the Journal document being entered. 6a1

A convenient way of dispensing with any directives past the origin statement in a file is to include a directive such as "IGDir=1" (ignore subsequent directives) in the comment field of the header (using the Comment statement). 6a2

Section 3. IDENTIFICATION SYSTEM

1

Some of the functions of the Identification System are to allow new users to be recognized by the system and to allow users to modify existing information about people already known to the system. The two basic functions discussed here are identifying persons/groups/affiliations to the system and modifying information about these. The Enter Identification command is used to initialize a record of information about new IDENTs. The Modify command enables the user to access a set of subcommands which are necessary not only to change information about IDENTs but also to add other (optional) information not explicitly requested by the identification procedure.

1a

ENTERING THE IDENTIFICATION SYSTEM FROM NLS

1b

The user may access the Identification System from NLS by using the following command:

1c

e[xecute] id[entification submode] CA

1c1

When this command is executed NLS prints the herald character ">"; the user may then use any valid Identification System command.

1d

Errors caused by the user and CDs (Command Deletes) do not cause the user to leave the Identification System; the user may return to NLS only by using the Quit command:

1e

q[uit] CA

1e1

ENTERING THE IDENTIFICATION SYSTEM FROM THE JOURNAL OR NUMBER SYSTEM

1f

The user may also enter the Identification System directly from the Journal or Number Systems when he uses the .LIT, ?LIT and ?character procedures described in Section 1 of this document.

1g

ENTERING INDIVIDUALS 1h

Individual users may be entered into the Identification System by the following command sequence: 1h1

```
e[nter identification for] i[ndividual] CA [CR]
[last name:] LIT1 CA CR]
    .
  [ALL MATCHING LIT1's in IDENTFILE]
    .
[name in this list?] y[es]
                     CA
                     n[o]
                     SP                                          1h1a

[first name:] LIT2 CA [CR]
[middle initial:] LIT3 CA [CR]
                  CA [CR]
[affiliation:] IDENT1 CA [CR]
              /[new]LIT4 CA[affiliation address?]LIT5 CA         1h1b

  (if affiliation new)
  [temporary id for affiliation: IDENT2 CR]                      1h1b1

[id:IDENT3 CR]
[name: NAME  CR]
[affiliation: IDENT2 CR]
[address: ADDRESS CR]
[ok?] y[es]
      CA
      n[o really?] CA
      SP [really?]  CA
                                                                 1h1c

(If yes or CA)                                                   1h1d

[modify?] y[es]
          CA
          n[o] CA
          SP CA                                                  1h1e
```

Where LIT1 = the last name of the person being entered into the system. When the user types CA, the system prints all occurrences of that last name in the IDENTFILE. The system then asks the user if the name of the person intended is included in the list (i.e., the person is already entered into the system. If so, the user should respond with "y"[es] or CA which will return him to the most recent command level. Otherwise, typing SP or "n[o]" causes the system to prompt the user for the first name, middle name, etc. of the new person. 1h2

LIT2 = the first name of the new user. 1h3

LIT3 = the middle initial, name, etc. of the new user. This parameter is optional. 1h4

IDENT1 = the IDENT of the affiliation of the new person. (A list of current affiliations and their IDENTs is included in this user guide.) The entire IDENT for the affiliation must be entered. 1h5

LIT4 = (if the affiliation is new, the user must type the character "/" and enter the complete proper name for the new affiliation, e.g. Augmentation Research Center. The name of the affiliation must be spelled out. If the affiliation is new, the system will ask the user to enter the address of the affiliation. This address will be verified by the NIC clerk when recent IDENTFILE entries are checked. If LIT is not specified, the affiliation "independent" is assumed. 1h6

LIT5 = mailing address of the new affiliation in the format:
STREET NUMBER SP STREET NAME CR
CITY comma STATE SP ZIP CODE 1h7

IDENT2 = the IDENT for affiliation. (This IDENT is considered temporary until verified by the NIC clerk at ARC.) 1h8

IDENT3 = the system assigned mnemonic for the user indicated by first LIT. After the system prints IDENT it prints the status of the entry just created for the new user. In response to "ok?" the current user may verify the information by typing "y[es]" or CA which causes the system to ask the user if he wants to modify the information. If "yes" or CA the user is placed in the modify command mode; typing "no" (or SP) confirmed by a CA causes all information entered by the user since the last command to be deleted and the user returned to the most recent command level. 1h9

Responding to "modify?" by "yes" or CA causes the system to place the user in the modify command mode (see the Modify command in this section); responding by "no" or SP causes the system to return the user to the Identification System command level. 1h10

ENTERING GROUPS 1i

Groups may be entered into the Identification System by a process similar to that for entering individuals: 1i1

```
e[nter identification for] g[roup] CA [CR]
[name:] LIT1 CA [CR]
[coordinator:] IDENT1 CA [CR]
[membership list:] IDENTLIST CA [CR]
[identification is: IDENT2 ok?] y[es]
                                CA
                                n[o]
                                SP                          1i1a

   (if no or SP)
   [assign id:] LIT2 CA                                     1i1a1

[id:IDENT3 CR]
[name:LIT1 CR]
[coordinator: IDENT1 CR]
[membership list: IDENTLIST CR]
[ok?] y[es]
      CA
      n[o really?] CA
      SP [really?] CA                                       1i1b

(If yes or CA)
[modify?] y[es]
          CA
          n[o] CA
          SP CA                                             1i1c
```

Where LIT1 = the proper name of the interest group being entered into the system (e.g. Network Graphics Group). 1i2

IDENT1 = the IDENT of the coordinator of the group. If no IDENT is specified, the IDENT of the current user is assumed. 1i3

IDENTLIST = the members of the group (including other groups) 1i4

IDENT2 = the system assigned IDENT for the group. This may be changed by typing "no" or . SP in response to [ok?] and entering LIT2. 1i5

LIT2 = the user assigned IDENT for the group (if the user is unsatisfied with the IDENT created by the system. If an identical ident already exists in the IDENTFILE, the system will prompt the user again with "assign id". 1i6

IDENT3 = the final IDENT for the group. After the system prints IDENT it prints the status of the entry just created for the new group. In response to "ok?" the user may verify the information by typing "y[es]" or CA which causes the system to ask the user if he wants to modify the information. If "yes" or CA the user is placed in the modify command mode; typing "no" (or SP) confirmed by a CA causes all information entered by the user since the last command to be deleted and the user returned to the most recent command level. 1i7

Responding to "modify?" by "yes" or CA causes the system to place the user in the modify command mode (see the Modify command in this section); responding by "no" or SP causes the system to return the user to the Identification System command level.

1i8

Example: When the user is operating in the Identification System, instances may arise where the user finds it necessary to define one set of information before he can define another. For example, in the case of entering new groups, it is necessary that the user identify the coordinator of the group after the group identification process is begun. If the coordinator himself is not known to the system, the user must interrupt identifying the group to identify the group coordinator. The Identification System has made provision for this eventuality by allowing the user to suspend one definition sequence, move to another, and return to where he left off originally. Cases of this type generally follow the same sequence as in the following example of defining a new group: 1i9

1. Enter identification system 1i9a
2. Enter identification for group 1i9b
   2a. Specify group name 1i9b1

3. Enter IDENT of group coordinator 1i9c
(if coodinator IDENT is unknown to the user) 1i9c1
3a. Type ".lastname" - to see if coordinator is known to the system and if so, his IDENT. If IDENT is found return to step 3. Otherwise respond yes or CA to enter new IDENT to identify coordinator to system. 1i9c1a
3a1. Enter identification for new person (coordinator) 1i9c1a1
3a2. Verify information for new person; if correct, return to step 3. 1i9c1a2
3a3. If incorrect, enter modify 1i9c1a3
3a3a. Modify information 1i9c1a3a
3a3b. Return to step 3a2. 1i9c1a3b
4. Enter group membership 1i9d
4a. Enter IDENTLIST for membership: IDENT1 SP IDENT2 SP IDENT3 SP ... 1i9d1
(next member's IDENT is unknown to the user) 1i9d2
4b. Type ".lastname" - to see if next member is known to the system and if so, his IDENT. If IDENT is found return to step 4. Otherwise respond "yes" or CA to enter new IDENT to identify the member to system. 1i9d3
4b1. Enter identification for new person. 1i9d3a
4b2. Verify information for new person; if correct, return to step 4. 1i9d3b
4b3. If incorrect, enter modify 1i9d3c
4ba. Modify information 1i9d3c1
4b3b. Return to step 4b2. 1i9d3c2
5. System prints a summary of the new identification record. 1i9e

ENTERING AFFILIATIONS 1j

Affiliations may be identified to the system by the following process. 1j1

```
e[nter identification for] a[ffiliation] CA [CR]
[name:] LIT1 CA [CR]
[mailing address] LIT2 CA [CR]
[temporary id for affiliation: IDENT CR]
[name: LIT1 CR]
[mailing address: LIT2 CR]
[ok?] y[es]
    CA
    n[o really?] CA
    SP [really?]  CA                                    1j1a

(if yes or CA)
[modify?] y[es]
          CA
          n[o] CA
          SP CA                                         1j1b
```

Where LIT1 = the proper name of the professional affiliation, e.g. Stanford Research Institute. 1j2

LIT2= mailing address of the new affiliation in the format:
STREET NUMBER SP STREET NAME CR
CITY comma STATE SP ZIP CODE 1j3

IDENT = IDENT for affiliation assigned by system. This IDENT is considered temporary until verified by the NIC clerk at ARC. After the system prints IDENT it prints the status of the entry just created for the new affiliation. In response to "ok?" the user may verify the information by typing "y[es]" or CA which causes the system to ask the user if he wants to modify the information. If "yes" or CA the user is placed in the modify command mode; typing "no" (or SP) confirmed by a CA causes all information entered by the user since the last command to be deleted and the user returned to the most recent command level. 1j4

7638 1-OCT-71 ARC
IDENTIFICATION SYSTEM

Responding to "modify?" by "yes" or CA causes the system to place the user in the modify command mode (see the Modify command in this section); responding by "no" or SP causes the system to return the user to the Identification System command level. 1j5

USER STATUS AFTER ENTER MODE 1k

After the user has completed the Identification procedure for an individual or group, he may enter the modify mode by responding to the system prompt "modify?" by typing "y" or CA. Then he may proceed with any of the commands described under the modify mode as though he himself had typed the modify command. (Note that whenever the user enters the Modify mode in this manner, all changes/additions made pertain only to the IDENT of the new person/group/affiliation.) 1k1

MODIFY COMMAND 1l

The Modify command enables the user to access a set of modification subcommands which change certain fields in the existing record of information for a person, group, or affiliation in the IDENTFILE and to add information not explicitly asked for by the Identification procedure. 1l1

Note that changes to the IDENTFILE record for an IDENT are not permanently made until the user issues the Update command, thus. if the user leaves the system before updating, all information generated for an IDENT while in Modify mode will be ignored. 1l2

m[odify record for] IDENT CA
CA 1l2a

Where IDENT = a valid user identification indicating the user whose record will be altered in the Modify command mode. 1l3

When this command is executed, the user may enter any of the following Modify subcommands: 1l4

Note that in each of the following subcommands, the system prints the current value of that field. The user may keep that value simply by typing CA, i.e., not changing the value for the field. 1l5

AFFILIATION 1l6

This command allows the user to add or delete affiliations, or to to redefine the entire affiliation field of the person represented by IDENT. 1l6a

af[filiation IDENTLIST1 CR]
[+] a[dd] IDENTLIST2 CA
d[elete] IDENTLIST3 CA
i[nitialize] IDENTLIST4 CA 1l6a1

Where IDENTLIST1 = current affiliations of IDENT (the first in the list being the primary) 1l6b

IDENTLIST2 = affiliations to be added to the current list. 1l6c

IDENTLIST3 = affiliations to be deleted from the current list. 1l6d

IDENTLIST4 = affiliations to replace the current list. 1l6e

This command may be used only for the IDENTS of individuals. Otherwise, the system will issue the message:
GROUPS & AFFILIATIONS ARE NOT AFFILIATED 1l6f

The affiliation field of a record may be altered only by an individual himself or by persons designated by him. 1l6g

COORDINATOR 1l7

This command enables the user to change the coordinator of a group. 1l7a

co[ordinator: CURRENT IDENT] IDENT CA 1l7a1

An attempt to use this command with an individual IDENT causes the system to print the error message:
INDIVIDUALS DO NOT HAVE COORDINATORS 1l7b

The coordinator field may be changed only by the coordinator of the same group or by persons designated by him. 1l7c

EXPAND GROUP REFERENCE 1l8

This command enables the user to control whether Journal entries are distributed to each individual in a group, or just to the address indicated for the group. The default value is expand. 1l8a

e[xpand normal references ?] y[es]
CA
n[o]
SP 1l8a1

Where yes = distribution will be made to each member of the group. This is equivalent to CA. 1l8b

CA = same as yes. 1l8c

no = distribution will be made to the group as a whole, i.e., to the group coordinator. 1l8d

SP = same as no. 1l8e

An attempt to use this command with an individual IDENT will cause the system to print the error message:
INDIVIDUAL IDENTS MAY NOT BE EXPANDED 1l8f

This field may be modified only by the coordinator of the group or by users designated by him. 1l8g

GROUP MEMBERSHIP 1l9

This command enables the user to add and delete members of a group, or to redefine the entire group membership. 1l9a

g[roup membership IDENTLIST CR]
[+] a[dd] IDENTLIST CA
d[elete] IDENTLIST
i[nitialize] IDENTLIST 1l9a1

Where first IDENTLIST = the names of all users currently belonging to the group. 1l9b

second, third,
and fourth IDENTLIST = additions, deletions, or new group names depending on the command (add, delete, initialize, respectively). 1l9c

An attempt to use this command with an individual IDENT will cause the system to print the error message:
INDIVIDUALS DO NOT HAVE MEMBERSHIP LISTS 1l9d

This field may be modified only by the coordinator or the group, or by persons given special permission by the coordinator. 1l9e

ADDRESS 1l10

This command changes the mailing list associated with IDENT. 1l10a

ma[iling address: CURRENT ADDRESS] LIT CA
IDENT 1l10a1

Where CURRENT ADDRESS = currrent address assignment. 1l10b

LIT = new mailing address in the format:
STREET NUMBER SP STREET NAME CR
CITY comma STATE SP ZIP CODE 1l10c

IDENT = IDENT of some recognized user or group or affiliate. 1l10d

This field may be modified only by the individual himself, the coordinator of a group, or persons given permission to do so by either. 1l10e

NAME 1l11

This command enables the user to change the name associated with IDENT. 1l11a

n[ame: CURRENT NAME] LIT1 CR
(for individual IDENTs only)
[first name:] LIT2 CA [CR]
[middle initial:] LIT3 CA [CR] 1l11a1

Where CURRENT NAME = current name associated with IDENT 1111b

LIT1 = new name (last name for individual IDENTs) 1111c

LIT2 = new first name 1111d

LIT3 = middle initial(s), name, etc. 1111e

This field may be modified only by the individual himself, the coordinator of a group, or persons given permission to do so by either.

1111f

PHONE 1112

This command enables the user to change or add the phone number associated with IDENT. 1112a

p[hone: CURRENT PHONE] LIT CA 1112a1

Where CURRENT PHONE = current phone number associated with IDENT. 1112b

LIT = new phone number (including area code). 1112c

This field may be changed only by the individual himself, by the coordinator of a group, or by persons designated by these.

1112d

FUNCTION 1113

This command enables the user to change or add the function (title) associated with IDENT. 1113a

f[unction: CURRENT FUNCTION] LIT CA 1113a1

Where CURRENT FUNCTION = current Network-related function associated with IDENT. 1113b

LIT = new function, e.g., Liaison, Agent, Manager, Principal Investigator. 1113c

This field may be changed only by the individual himself, by the coordinator of a group, or by persons designated by these. 1l13d

ACCESS 1l14

This command enables the user to change or add the persons who are permitted to modify his record in IDENTFILE. 1l14a

ac[cess: CURRENT IDENTLIST] IDENTLIST CA 1l14a1

Where CURRENT IDENTLIST = current list of users permitted to modify information about IDENT. 1l14b

IDENTLIST= new list of users permitted to modify information about IDENT. 1l14c

This field may be changed only by the individual himself or by the coordinator of a group. 1l14d

STATUS 1l15

This command causes the various fields in the identification record associated with IDENT that have been modified to be printed at the terminal. 1l15a

st[atus] CA 1l15a1

UPDATE 1l16

This command causes all changes made to an IDENT while in the Modify mode to be made permanent. 1l16a

up[date] CA 1l16a1

QUIT 1m

The command enables the user to return to the NLS command level. 1m1

q[uit] CA 1m1a

(If the user has not previously issued the Update command)
[update?] y[es]
CA
n[o]
SP 1m1b

Unless the user responds to the "update?" prompt by "yes" or CA, all the information entered in the Modify mode for IDENT will not be incorporated into the IDENTFILE. 1m2

7639 1-OCT-71 ARC
NUMBER SYSTEM

Section 4. NUMBER SYSTEM

1

INTRODUCTION 1a

The Number System enables users to obtain unique number identifiers for items submitted to the Journal and RFC's in advance of actually submitting these items to the Journal. These pre-assigned numbers should be obtained only when necessary. Numbers are generally obtained when the user is actually submitting a document for distribution through the Journal. However, pre-assigned numbers are necessary, for example, when a user references in a Journal document another document not yet entered in the Journal system and when a document is prepared offline. 1a1

The following NLS command places the user in the Number System mode wherein he may obtain Journal numbers, release Journal numbers, or obtain RFC numbers. The user remains in this submode until he uses the Quit command or a system error occurs. 1a2

e[xecute] ca[talog numbers] 1a2a

When this command is executed the Number System herald character "#" is printed and the following commands may be used. 1a3

PRE-ASSIGN CATALOG NUMBER COMMAND 1b

p[re-assign Journal catalog number(s) to user(s):]
IDENTLIST CA
[count =] EMPTY CA
NUMBER 1b1

Where EMPTY= one number will be assigned by default. 1b2

NUMBER = an integer value from 1 to 20 indicating the amount of pre-assigned catalog numbers to be obtained. If more than 20 numbers are requested, an error message is generated and the user is returned to the NLS command level. 1b3

When this command is executed, the system records the name of the user(s) specified by IDENTLIST and the current date and time with the catalog number(s) assigned by this operation, and prints the catalog number(s) at the terminal.

1b4

7639 1-OCT-71 ARC
NUMBER SYSTEM

RFC NUMBER COMMAND 1c

This command enables the user to obtain a single RFC number as well as a Catalog number. 1c1

```
r[fc number (pre-assigned)] CA [CR]
[title:] LIT CA [CR]
[online document?] y[es]
                   CA [yes]
                   n[o] [CR   distribute via nic?] n[o]
                   SP [no]                         y[es]
```
1c1a

```
(if distributed via NIC or online)
[distribution:] IDENTLIST CA [CR]
```
1c1a1

```
[author(s): IDENTLIST CR]
[title: LIT CR]
[OnLine Document or OffLine Document]
[distribution: IDENTLIST]
[go?] y[es]
      CA
      n[o] CA
      SP CA
```
1c1b

```
(if yes or CA)
[rfc # RFCNUM]
[catalog # CATNUM]
```
1c1c

Where IDENTLIST = any number of system-assigned user identifiers separated by commas. (Current identifications may be found in the IDENTFILE.) 1c2

LIT = any string of characters which constitute the tentative title. 1c3

yes = (to [on-line document]) causes the document to be submitted in the form of an on-line Journal document. No indicates hard-copy distribution. 1c4

yes = (to [distribute via NIC?]) causes the document to be distributed by NIC. No indicates that the author will distribute the document. 1c5

no = or SP to "go?" allows the user to change (using the commands described below) any information previously entered. 1c6

RFCNUM = the RFC number assigned by the system. 1c7

CATNUM = the catalog number assigned by the system. 1c8

The above information supplied by the user is considered tentative and can be changed when the item is actually submitted to the Journal. The information is needed at this time by NIC to keep track of pre-assigned numbers and documents. 1c9

After this command is executed, the user is returned to the Number System command level wherein he may use any of the following commands to change the information entered by the RFC command. 1c10

AUTHOR 1c11

This command is identical to the Author command in the Journal System. 1c11a

DISTRIBUTION 1c12

This command is identical to the Distribution command in the Journal System. 1c12a

TITLE 1c13

This command is identical to the Title command in the Journal System. 1c13a

STATUS 1c14

This command is identical to the Status command in the Journal System. 1c14a

ONLINE DOCUMENT 1c15

o[nline document?] y[es]
CA [yes]
n[o CR distribute via nic?] y[es]
SP [no] n[o] 1c15a

(if distribute via NIC or online)
[distribution:] IDENTLIST CA
1c15a1

GO 1c16

This command is identical to the Go command in the Journal System.
1c16a

QUIT COMMAND 1d

The Quit command enables the user to leave the Number System and return to the NLS command level. 1d1

q[uit] CA 1d1a

This command and system errors are the only means by which the user may leave the Number System. 1d2

7640 1-OCT-71 ARC
COMMAND SUMMARY

Appendix A. JOURNAL SYSTEM COMMAND SUMMARY

1

JOURNAL COMMANDS

1a

```
e[xecute] j[ournal  CR]
[submit] s[tatement] ADDRESS          CA [CR]
       b[ranch] ADDRESS
       g[roup] ADDRESS CA ADDRESS
       p[lex] ADDRESS
       f[ile]
       m[essage CR] LIT
       SP [CR] LIT
```

1a1

```
[number:] CA [CATNUM]
          CATNUM CA [(assigned to):] IDENT CA
             (if RFCNUM number associated with CATNUM)
             [rfc # RFCNUM]
          r[fc number:] RFCNUM CA [(assigned to):] IDENT CA
                        CA [rfc # RFCNUM CR]
                           [catalog # CATNUM]
```

1a2

```
a[uthor(s):] CA
             IDENTLIST CA
```

1a2a

```
cl[erk:] CA
           IDENT CA
```

1a2b

```
co[mments:]  LIT CA
```

1a2c

```
d[istribution:] IDENTLIST CA
```

1a2d

```
g[o?]  CA
       y[es]
       n[o]
       SP
```

1a2e

```
i[nterrogate] CA
                                                                    1a2f

k[eywords:] WORDLIST CA
                                                                    1a2g

o[bsoletes document(s):] CATNUMLIST CA
                                                                    1a2h

p[lace link (if successful)?] CA
                                y[es]
                                n[o]
                                SP
   (if yes or CA)
   [after location:] ADDRESS CA LEVADJ CA                           1a2i

q[uit]  CA
                                                                    1a2j

st[atus] CA
                                                                    1a2k

su[bcollection(s):] IDENTLIST CA
                                                                    1a2l

t[itle:]  LIT CA
                                                                    1a2m

u[pdates document(s):] CATNUMLIST CA
                                                                    1a2n

e[xecute] se[condary distribution CR]
[document #] CATNUM CA
[to:] IDENTLIST CA                                                  1a3
```

IDENTIFICATION SYSTEM COMMANDS

1b

e[xecute] id[entification submode] CA

1b1

e[nter identification for] i[ndividual] CA [CR]
[last name:] LIT1 CA CR]
•
[ALL MATCHING LIT1's in IDENTFILE]
•
[name in this list?] y[es]
CA
n[o]
SP 1b2

(If no or SP) 1b3

[first name:] LIT2 CA [CR]
[middle initial:] LIT3 CA [CR]
[affiliation:] IDENT1 CA [CR]
/[new] LIT4 CA [affiliation address?] LIT5 CA 1b4

(if affiliation new)
[temporary id for affiliation: IDENT2 CR] 1b4a

[id: IDENT3 CR]
[name: NAME CR]
[affiliation: IDENT2 CR]
[address: ADDRESS CR]
[ok?] y[es]
CA
n[o really?] CA
SP [really?] CA

1b5

(If yes or CA) 1b6

[modify?] y[es]
CA
n[o] CA
SP CA

1b7

```
e[nter identification for] g[roup] CA [CR]
[name:] LIT1 CA [CR]
[coordinator:] IDENT1 CA [CR]
[membership list:] IDENTLIST CA [CR]
[identification is: IDENT2 ok?] y[es]
                                CA
                                n[o]
                                SP                                  1b8

   (if no or SP)
   [assign id:] LIT2 CA                                             1b8a

[id: IDENT3 CR]
[name: LIT1 CR]
[coordinator: IDENT1 CR]
[membership list: IDENTLIST CR]
[ok?] y[es]
      CA
      n[o really?] CA
      SP [really?] CA                                               1b9

(If yes or CA)
[modify?] y[es]
         CA
          n[o] CA
          SP CA
                                                                    1b10

e[nter identification for] a[ffiliation] CA [CR]
[name:] LIT1 CA [CR]
[mailing address] LIT2 CA [CR]
[temporary id for affiliation: IDENT CR]
[name: LIT1 CR]
[mailing address: LIT2 CR]
[ok?] y[es]
    CA
    n[o really?] CA
    SP [really?]  CA                                                1b11

(if yes or CA)
[modify?] y[es]
          CA
          n[o] CA
          SP CA
                                                                    1b12
```

7640 1-OCT-71 ARC
COMMAND SUMMARY

```
m[odify record for] IDENT CA
                    CA
```
1b13

```
ac[cess: CURRENT IDENTLIST]  IDENTLIST CA
```
1b13a

```
af[filiation CURRENT IDENTLIST   CR]
[+] a[dd] IDENTLIST2 CA
    d[elete] IDENTLIST3 CA
    i[nitialize] IDENTLIST4 CA
```
1b13b

```
co[ordinator: CURRENT IDENT] IDENT CA
```
1b13c

```
e[xpand normal references ?] y[es]
                             CA
                             n[o]
                             SP
```
1b13d

```
f[unction: CURRENT FUNCTION]  LIT CA
```
1b13e

```
g[roup membership  IDENTLIST CR]
[+] a[dd ] IDENTLIST     CA
    d[elete] IDENTLIST
    i[nitialize] IDENTLIST
```
1b13f

```
ma[iling address: CURRENT ADDRESS] LIT   CA
                                   IDENT
```
1b13g

```
n[ame: CURRENT NAME] LIT1 CR
(for individual IDENTs only)
[first name:] LIT2 CA [CR]
[middle initial:] LIT3 CA
```
1b13h

```
p[hone: CURRENT PHONE]  LIT CA
```
1b13i

```
q[uit] CA
```
1b13j

st[atus] CA

1b13k

up[date] CA

1b13l

7640 1-OCT-71 ARC
COMMAND SUMMARY

NUMBER SYSTEM COMMANDS

1c

e[xecute] ca[talog numbers] 1c1

p[re-assign journal catalog numbers(s) to user(s):]
IDENTLIST CA
[count =] EMPTY CA
NUMBER
1c2

r[fc number (pre-assigned)] CA [CR]
[title:] LIT CA [CR]
[online document?] y[es]
CA [yes]
n[o] [CR distribute via nic?] n[o]
SP [no] y[es] 1c3

(if distributed via NIC or online)
[distribution:] IDENTLIST CA [CR] 1c3a

[author(s): IDENTLIST CR]
[title: LIT CR]
[OnLine Document or OffLine Document]
[distribution: IDENTLIST]
[go?] y[es]
CA
n[o] CA
SP CA 1c4

(if yes or CA)
[rfc # RFCNUM]
[catalog # CATNUM]
1c5

a[uthor(s):] IDENTLIST CA
1c5a

d[istribution:] IDENTLIST CA
1c5b

g[o?] y[es]
CA
n[o] CA
SP CA
1c5c

o[nline document?] y[es]
CA [yes]
n[o CR distribute via nic?] y[es]
SP [no] n[o] 1c5d

q[uit] CA 1c5e

s[tatus] CA
1c5f

t[itle:] LIT CA
1c5g

q[uit] CA 1c6

7641 1-OCT-71 ARC
INDIVIDUAL IDENTS

Appendix B. CURRENT LIST OF INDIVIDUAL IDENTS

1

IDENT	NAME	AFFILIATION	PHONE	
				1a
DIA	Andrews, Don I.	(SRI-ARC)	(707) 874-3692	1b
HCA	Andrews, Harry C.	(USC)	(213) 746-2578	1c
MFA	Auerbach, Marilyn F.	(SRI-ARC)	(415) 326-6200 x3722	1d
	Aupperle, E. M.	(MERIT)	(313) 764-9423	1e
JHB	Bair, James H.	(RADC)	(315) 330-7834	1f
TJB	Barcalow, Thomas J.	(LINC)	(617) 862-5500	1g
JPB	Barden, John P.	(CASE)	(216) 368-4467	1h
BAB	Barett, Barbara A.	(SU-AI)	(415) 321-2300 x2800	1i
WLB	Bass, Walter L.	(SRI-ARC)	(415) 326-6200 x4372	1j
RDB	Bates, R. D.	(SRI-ARC)	(415) 326-6200 x3614	1k
ICB	Beattie, Imogen C.	(UCLA-CCN)	(213) 825-7541	1l
AJB	Bernstein, Art J.	(SUNY)	(516) 246-4080	1m
AKB	Bhushan, Abhay K.	(MIT-DMCG)	(617) 864-6900 x1428	1n
DGB	Bobrow, Daniel G.	(BBN-TENEX)	(617) 491-1850 x330	1o
WJB	Bouknight, W. Jack	(ILL)	(217) 333-0395	1p
RTB	Braden, Robert T.	(UCLA-CCN)	(213) 825-7518	1q
WHB	Broadley, William H.	(CMU)	(412) 683-7000	1r

IDENT	NAME	AFFILIATION	PHONE	
				1s
DRB	Brown, David R.	(SRI-ARC)	(415) 326-6200 x2944	1t
NB	Brown, Nan	(ILL)	(217) 333-7161	1u
NFB	Bruderer, Nancy F	(UTAH)	(801) 322-8224	1v
RFB	Bryon, Roland F.	(UCSB)	(805) 961-3933	1w
VGC	Cerf, Vint G.	(UCLA-NMC)	(213) 825-4864	1x
TEC	Cheatham, Thomas E.	(HARV)	(617) 495-1000	1y
SGC	Chipman, Steve G.	(BBN-TENEX)	(617) 491-1850 x358	1z
MSC	Church, Mary S.	(SRI-ARC)	(415) 326-6200 x3001	1a*
DC	Cohen, Dan	(HARV)	(617) 495-3991	1aa
ALC	Coley, Anita L.	(UCLA-NMC)	(213) 825-4797	1ab
LMC	Connelly, Linda M.	(RAND)	(213) 393-0411 x635	1ac
MGC	Cornell, Madge G.	(MITRE)	(703) 893-3500 x2395	1ad
IWC	Cotton, Ira W.	(MITRE)	(704) 893-3500 x2887	1ae
	Cowan, Donald	(WATERU)	(519) 744-6111 x3292	1af
SDC2	Crocker, Steve D.	(ARPA)	(202) 694-5051 694-5922	1ag
WRC	Crowther, Will R.	(BBN-NET)	(617) 491-1850 x459	1ah
GD	Davis, Glenn	(UCSB)	(805) 961-2462	1ai

7641 1-OCT-71 ARC
INDIVIDUAL IDENTS

IDENT	NAME	AFFILIATION	PHONE	
				1aj
MRD	Davis, Mal R.	(RAND)	(213) 393-0411 x652	1ak
	Davies, Thomas	(AMES-CD)	(415) 961-1111 x2485	1al
LPD	Deutsch, L. Peter	(XEROX)	(415) 493-1600 x41	1am
TRD	Dines, Thomas R.	(AMES-CD)	(415) 961-1111 x2485	1an
BAD	Dolan, Bruce A.	(ARPA)	(202) 694-5921 694-5922	1ao
WSD	Duvall, William S.	(SRI-ARC)	(415) 326-6200 x3630	1ap
DCE	Engelbart, Douglas C.	(SRI-ARC)	(415) 326-6200 x2220	1aq
DCE2	Evans, David C.	(UTAH)	(801) 322-8224	1ar
JAF	Fadiman, J. A.	(SRI-ARC)	(415) 326-6200 x2984	1as
EAF	Feigenbaum, Edward A.	(SU-HP)	(415) 321-2300 x4878	1at
	Farber, David	(UCI)	(714) 833-6891	1au
JWF	Forgie, James W.	(LINC-67)	(617) 862-5500 x7173	
		(LINC-TX2)	(617) 862-5500 x7173	1av
PWF	Foulk, Patrick W.	(CASE)	(216) 368-4467 x2936	1aw
	Fredkin, Edward	(MAC)	(617) 864-6900 x5861	1ax
MAG	Ginsberg, Martha A.	(CCA)	(617) 491-3670	1ay

IDENT	NAME	AFFILIATION	PHONE	
				1az
ELG	Glaser, Edward L.	(CASE)	(216) 368-2808	1b*
MMG	Goering, Margaret M.	(ARPA)	(202) 694-5921 694-5922	1ba
SG	Golding, Stan	(AMES-CD)	(415) 961-1111 x2501	1bb
MLG	Graham, Marvin L.	(ILL)	(217) 333-8051 333-3743	1bc
CCG	Green, C. Cordell	(ARPA)	(202) 694-5921 694-5922	1bd
GRG	Grossman, Gary R.	(ILL)	(217) 333-4266	1be
DMG	Grothe, David M.	(ILL)	(217) 333-4266	1bf
BAH	Hardeman, B. A.	(SRI-ARC)	(415) 326-6200	1bg
MEH	Hardy, Martin E.	(SRI-ARC)	(415) 326-6200 x3921	1bh
RFH	Hargraves, Robert F	(DART)	(603) 646-2643 x3755	1bi
DOH	Harris, David O.	(UCSB)	(805) 961-2534 961-2931	1bj
EFH	Harslem, Eric F.	(RAND)	(213) 393-0411 x7320	1bk
AWH	Hathaway, A. Wayne	(AMES-CD)	(415) 961-1111 x2493 x2117	1bl
JFH	Heafner, John F.	(RAND)	(213) 393-0411 x7606	1bm
FEH	Heart, Frank E.	(BBN-NET)	(617) 491-1850 x470	1bn
FPH	Hocker, Fred P.	(SRI-ARC)	(415) 326-6200 x4057	1bo

7641 1-OCT-71 ARC
INDIVIDUAL IDENTS

IDENT	NAME	AFFILIATION	PHONE	
				1bp
JDH	Hopper, J. David	(SRI-ARC)	(415) 326-6200 x2352	1bq
CHI	Irby, Charles H.	(SRI-ARC)	(415) 326-6200 x4611	1br
CI	Iselin, Chuck	(CASE)	(216) 368-2918	1bs
MEJ	Jernigan, M. E.	(SRI-ARC)	(415) 326-6200 x3007	1bt
REK	Kahn, Robert E.	(BBN-TENEX)	(617) 491-1850	1bu
RBK	Kalin, Richard B.	(LINC-TX2)	(617) 862-5500	1bv
	Kantrowitz, William	(LINC-TX2)	(617) 862-5500 x7349	1bw
PMK	Karp, Peggy M.	(MITRE)	(703) 893-3500 x2524	1bx
ACK	Kay, Alan C.	(XEROX)	(415) 493-1600	1by
KCK	Kelley, Karl C.	(ILL)	(217) 333-8469	1bz
	Kirstein, Peter	(UKICS)		1c*
LK	Kleinrock, Leonard	(UCLA-NMC) (UCLA-CCN)	(213) 825-2543 (213) 825-2543	1ca
CSK	Kline, Chuck S.	(UCLA-NMC)	(213) 825-2756 825-2368	1cb
FYK	Knight, Frances Y.	(MAC)	(617) 864-6900 x1458	1cc
MCK	Krilanovich, M. C.	(UCSB)	(805)-961-2476	1cd
REK2	Kruger, Richard E.	(USC)	(213) 746-2580	1ce
	Kurtz, Tom	(DART)	(603) 646-2643 x3755	1cf

IDENT	NAME	AFFILIATION	PHONE	
				lcg
	Lampson, Butler	(XEROX)	(415) 493-1600	lch
ASL	Landsberg, Abe S.	(SDC)	(213) 393-9411 x454 x6119	lci
LLL	Lane, Linda L.	(SRI-ARC)	(415) 326-6200 x3007	lcj
TSL	Lawrence, Thomas S.	(RADC)	(315) 330-3857 330-7843	lck
TJL	Layman, Terry J.	(ILL)	(217) 333-8051	lcl
JCL	LeGates, John C.	(EDUCOM)	(617) 227-1805	lcm
HGL	Lehtman, Harvey G.	(SRI-ARC)	(415) 326-6200 x4174	lcn
JCRL	Licklider, J. C. R.	(MIT-DMCG)	(617) 864-6900 x7705	lco
	Liddle, Dave E.	(OWENS)	(419) 242-6543	lcp
CML	Lisle, Carolyn M.	(CMU)	(412) 683-7000 x269	lcq
CRL	Loepkey, Charles R.	(UCSB)	(805) 961-2261	lcr
REL	Long, Robert E.	(SDC)	(213) 393-9411	lcs
SKL	Lynch, Steve K.	(UCSB)	(805) 961-3664	lct
JMM	Madden, James M.	(ILL)	(217) 333-7162	lcu
SLM	Mantiply, Stan L.	(AMES-ILLIAC)	(415) 961-1111 x2449	lcv
TMM	Marill, Thomas M.	(CCA)	(617) 491-3670	lcw

7641 1-OCT-71 ARC
INDIVIDUAL IDENTS

IDENT	NAME	AFFILIATION	PHONE	
				1cx
JM	McCarthy, John	(SU-AI)	(415) 321-2300 x4430	1cy
JWM	McConnell, John W.	(AMES-ILLIAC)	(415) 961-1111 x2449	1cz
DEM	McIntyre, David E.	(ILL)	(217) 333-6453	1d*
	McKay, Douglas B.	(IBM)	(914) 945-1159	1da
AAM	McKenzie, Alex A.	(BBN-NET)	(617) 491-1850 x441	1db
MM	McKinley, Mac	(BBN-TENEX)	(617) 491-1850	1dc
GHM	Mealy, George H.	(HARV)	(617) 495-3998	1dd
JTM	Melvin, John T.	(SRI-ARC)	(415) 326-6200 x4328	1de
RMM	Metcalfe, Robert M.	(MIT-DMCG)	(617) 864-6900 x1429 x1458	1df
EWM	Meyer, Edwin W., Jr.	(MIT-MULTICS)	(617) 864-6900 x6006	1dg
JCM	Michener, James C.	(MIT-DMCG)	(617) 864-6900 x1430	1dh
JGM	Mitchell, James G.	(XEROX)	(415) 493-1600 x42	1di
JAM	Moorer, James A.	(SU-AI)	(415) 321-2300 x4971	1dj
CJM	Mostrom, Carol J.	(LINC-67)	(617) 862-5500 x7177	1dk
DLM	Murphy, Dan L.	(BBN-TENEX)	(617) 491-1850 x351	1dl

IDENT	NAME	AFFILIATION	PHONE	
				1dm
LCN	Nelson, Lou C.	(UCLA-NMC)	(213) 825-4733 825-2368	1dn
JBN	North, Jeanne B.	(SRI-ARC)	(415) 326-6200 x4119	1do
JCN	Norton, James C.	(SRI-ARC)	(415) 326-6200 x2124	1dp
DRO	Oestreicher, Don R.	(UTAH)	(801) 581-8418 581-6802	1dq
AOJO	Ollikainen, Ari O.	(UCLA-NMC)	(213) 825-2381 581-2368	1dr
	O'Sullivan, Thomas	(RAY)	(617) 762-6700 x2120	1ds
MAP	Padlipsky, Michael A.	(MIT-MULTICS)	(617) 864-6900 x6007	1dt
CXP	Page, Cindy	(SRI-ARC)	(415) 326-6200 x3007	1du
WJP	Page, William J.	(CASE)	(216) 368-2984	1dv
BLP	Parsley, Bruce L.	(SRI-ARC)	(415) 326-6200 x3080	1dw
WHP	Paxton, William H.	(SRI-ARC)	(415) 326-6200 x4111	1dx
	Pepin,James	(USC)	(213) 746-2240	1dy
	Pepper, Marianne	(WASHU)	(314) 361-7356	1dz
AJP	Perlis, Alan J.	(CMU)	(412) 621-2600	1e*
MDP	Petell, Marcelle D.	(RADC)	(315) 330-4254 330-4230	1ea
JCP	Peters, Jeff C.	(SRI-ARC)	(415) 326-6200 x3002	1eb

7641 1-OCT-71 ARC
INDIVIDUAL IDENTS

IDENT	NAME	AFFILIATION	PHONE	
				lec
KP	Pogran, Ken	(MIT-MULTICS)	(617) 864-6900 x6019	led
JBP	Postel, Jon B.	(UCLA-NMC)	(213) 825-4797	lee
	Powell, Jerry J.	(MITRE)	(703) 893-3500 x2391	lef
WKP	Pratt, William K.	(USC)	(213) 746-2694	leg
EWP	Pughe, E. Wells	(RAY)	(617) 762-6700 x147	leh
TNP	Pyke, Thomas N., Jr.	(NBS)	(301) 921-2601	lei
GR	Raichelson, Gene	(MITRE)	(703) 893-3500	lej
BR	Raphael, Bertram	(SRI-AI)	(415) 326-6200 x2209	lek
JR	Ratliff, Jake	(SRI-ARC)	(415) 326-6200 x4057	lel
BAR	Reussow, Bradley A.	(HARV)	(617) 495-4147 x2871	lem
RJR	Reynolds, Rilla J.	(SRI-AI)	(415) 326-6200 x4618	len
LGR	Roberts, Lawrence G.	(ARPA)	(202) 694-5921 694-5922	leo
BER	Row, Barbara E.	(SRI-ARC)	(415) 326-6200 x2469	lep
CDR	Rosewall, Connie	(UCSB)	(805) 961-3221	leq
	Saylor, Jean M.	(SDC)	(213) 393-9411 x7010	ler
	Schipper, Tjaart	(NETH)		les
CDS	Shepard, C. D.	(CCCTF)	(613) 992-9238	let

IDENT	NAME	AFFILIATION	PHONE	
				1eu
MSS	Sher, Michael S.	(ILL)	(217) 333-2979	1ev
AS	Shoshani, Arie	(SDC)	(213) 393-9411	1ew
DLS	Slotnick, Daniel L.	(ILL)	(217) 333-0925	1ex
TAS	Standish, Thomas A.	(HARV)	(617) 495-3991	1ey
JRS	Stellato, Josephine R.	(RADC)	(315) 330-3827 (315) 330-2672	1ez
DLS2	Stone, Duane L.	(RADC)	(315) 330-3857 (315) 330-7834	1f*
RMS	Stoughton, Ron M.	(UCSB)	(805) 961-3793	1fa
RLS	Sundberg, Robert L.	(HARV)	(617) 495-4147	1fb
	Thomas, Robert H.	(BBN-TENEX)	(617) 491-1850 x351	1fc
JMT	Torsen, James M.	(CASE)	(216) 368-2918	1fd
JWT	Troxel, Janet W.	(SDC)	(213) 393-9411 x495 x534	1fe
KWU	Uncapher, Keith, W.	(RAND)	(213) 393-0411 x433	1ff
EKV	Van De Riet, Ed K.	(SRI-ARC)	(415) 326-6200 x2425	1fg
DVN	van Nouhuys, Dirk H.	(SRI-ARC)	(415) 326-6200 x3370	1fh
HRVZ	Van Zoeren, Harold R.	(CMU)	(412) 683-7000 x267	1fi
AV	Vezza, Albert	(MIT-DMCG)	(617) 864-6900 x1450	1fj

7641 1-OCT-71 ARC
INDIVIDUAL IDENTS

IDENT	NAME	AFFILIATION	PHONE	
AHV	Vorhaus, Alfred H.	(MITRE)	(703) 893-3500 x2511	1fl
RKW	Walker, Robert K.	(RADC)	(315) 330-2501	1fm
DCW	Wallace, Don C.	(SRI-ARC)	(415) 326-6200 x4580	1fn
SWW	Watkins, Shirley W.	(NBS)	(301) 921-2601	1fo
RWW	Watson, Richard W.	(SRI-ARC)	(415) 326-6200 x2013	1fp
LMW	Webster, Linda M.	(USC)	(213) 746-6379	1fq
RVW	Weeks, Rollin V.	(UCLA-NMC)	(213) 825-2225	1fr
BW	Wegbreit, Ben	(HARV)	(617) 495-3751	1fs
CW	Weissman, Clark	(SDC)	(213) 393-9411 x533	1ft
BDW	Wessler, Barry D.	(UTAH)	(801) 322-8378	1fu
JEW	White, James E.	(UCSB)	(805) 961-3454	1fv
	White, Laura	(RAY)	(617) 762-6700	1fw
BMW	Wilber, B. Michael	(SRI-AI)	(415) 326-6200 x4593	1fx
CJW	Wilkinson, Carol J.	(SU-HP)	(415) 321-2300 x4878	1fy
	Williams, Tom G.	(SDC)	(213) 393-9411 x427	1fz
JMW	Winett, Joel M.	(LINC-67)	(617) 862-5500 x7474	1g*
RAW	Winter, Richard A.	(CCA)	(617) 491-3670	1ga

IDENT	NAME	AFFILIATION	PHONE	1gb
SMW	Wolfe, Steve M.	(UCLA-CCN)	(213) 825-7424	1gc
DCW2	Wood, David C.	(MITRE)	(703) 891-3500	1gd
LRZ	Zar, Leon R.	(ILL)	(217) 333-8051	1ge

7642 1-OCT-71 ARC
GROUP IDENTS

Appendix C. CURRENT GROUP IDENTS

IDENT	NAME & COORDINATOR
NWGSC	Network Working Group Steering Committee Coordinator: Steve D. Crocker
NWG	Network Working Group Coordinator: Steve D. Crocker
NFG	Network Facilitators Group Coordinator: Steve D. Crocker
PI	Principal Investigators Coordinator: Larry G. Roberts
NLG	Network Liaison Group Coordinator: Steve D. Crocker
NGG	Network Graphics Group Coordinator: Al Vezza
IIG	IMLAC Interest Group Coordinator: Charles H. Irby
NSAG	Network Station Agent Group Coordinator: Jeanne B. North
TUG	Tenex User Group Coordinator: Dan L. Murphy

7643 1-OCT-71 ARC
AFFILIATION IDENTS

Appendix D. CURRENT LIST OF AFFILIATION IDENTS

The following is a list of all currently known affiliations. Some of these affiliationa are also Sites in which case a phone number is also specified. NIC has arranged for Sites to have phone service to the Center with all tolls billed to NIC. To call NIC, a Site member should give his local operator the appropriate number. If the operator is unfamiliar with the arrangement, the caller should ask to speak to the phone Supervisor.

IDENT	AFFILIATION
AMES-CD	NASA Ames - CD Group dial direct, 329-0740
AMES-ILLIAC	NASA Ames - ILLIAC Group dial direct, 329-0740
ARPA	Advanced Research Projects Agency Ent 1-0740
BBN-NET	Bolt Beranek and Newman, Inc. - Network Group Ent 0740
BBN-TENEX	Bolt Beranek and Newman, Inc. - TENEX Group Ent 0740
CASE	Case Western Reserve University Ent 0740
CCA	Computer Corporation of America
CCCTF	Canadian Computer Communications Task Force
CMU	Carnegie-Mellon University Ent 9074
DART	Dartmouth University, Kiewit Computation Center
EDUCOM	EDUCOM

IDENT	AFFILIATION	
		15
HARV	Harvard University Ent 0740	16
IBM	IBM Watson Research Center	17
ILL	University of Illinois Ent 1074	18
LINC-67	M.I.T. Lincoln Lab - 67 Group Ent 0740	19
LINC-TX2	M.I.T. Lincoln Lab - TX-2 Group Ent 0740	20
MERIT	MERIT Computer Network	21
MIT-DMCG	Project MAC - DMCG Group Ent 0740	22
MIT-MULTICS	Project MAC - Multics Group Ent 0740	23
MITRE	MITRE Corporation Ent 1-0740	24
NETH	Tjaart Schipper	25
NPL	National Physical Laboratory	26
NBS	National Bureau of Standards	27
RADC	Rome Air Development Center (ISIM) Ent 0740	28
RAND	The Rand Corporation Zenith 9-0740	29
RAY	Raytheon Data Systems	30

7643 1-OCT-71 ARC
AFFILIATION IDENTS

IDENT	AFFILIATION	
		31
SDC	System Development Corporation Zenith 9-0740	32
SRI-AI	Stanford Research Institute - Artificial Intelligence Group	33
SRI-ARC	Stanford Research Institute - Augmentation Research Center	34
SU-AI	Stanford University - Artificial Intelligence Group dial direct 329-0740	35
SU-HP	Stanford University - Heuristic Programming Project dial direct 329-0740	36
SUNY	SUNY Stony Brook	37
UCI	University of California, Irvine	38
UCLA-CCN	UCLA - Campus Computing Network Zenith 9-0740	39
UCLA-NMC	UCLA - Network Measurement Center Zenith 9-0740	40
UCSB	University of California at Santa Barbara Zenith 9-0740	41
UKICS	University of London, Institute of Computer Science	42
USC	University of Southern California Zenith 9-0740	43
UTAH	University of Utah Zenith 9-0740	44
WASHU	Washington University, Computer Systems Lab	45
WATERU	University of Waterloo, Dept. of Computer Science	46

1

INDEX

2

NOTE: topics listed in this index are followed by links to Journal documents. This index may be used online by jumping to the appropriate link. For off-line use refer to the Journal document and statement number indicated. 3