RLL 17-DEC-74 21:08 24779

My comments on final report section (7a): Technology transfer

This is a fast comment, Rob

RLL 17-DEC-74 21:08 24779

My comments on final report section (7a): Technology transfer

< LIEBERMAN, REPORT,NLS;2, >, 17-DEC-74 21:06 RLL ;;;; 1

My comments and correctins are in ## signs. 1a

some of the items (e.g. numbered paragraphs 1),2),etc.) are not on the same NLS level, this may or may not be critical. 1b

##I supporse the loose ends (e.g. references) will be cleaned up. 1c

over all pretty good, hope to have more commetns if i have time on this eastern trip. Rob## 1d

Aspects of ARC's Technology Transfer Strategy RWW *PR (5) 1e

INTRODUCTION 1e1

The Augmentation Research Center (ARC) at Stanford Research Institute (SRI) has been actively involved in the development of a Knowledge Workshop System (Workshop) over the past ten-plus years [1]. During much of this period, ARC has been using the system called _, oNLine System (NLS), in its own work, and feeding this experience into the ongoing development process 1e1a

Three years ago with the connection of ARC computer system to the ARPANET and the establishment at ARC of the Network Information Center (NIC), we began to actively plan for and carry out an explicit technology transfer strategy. Previous experience had indicated traditional approaches to technology transfer - publishing papers and reports, giving demonstrations at conferences and at SRI, making movies, and giving slide shows and talks, while useful, were not enough to achieve technology transfer at the rate desired. Additional mechanisms were needed, particularly, hands-on experience by target groups. This chapter outlines some of the additional mechanisms being used and considerations for their selection. 1e1b

%% what rate? why a rate?? 1e1b1

Let us now lok ##look## at the technology transfer issues that lead ##led## ##could be either present or past tense## to these steps: 1e2

1) There are two approaches 1e2a

Need or demand pull, and 1e2a1

technology push 1e2a2

My comments on final report section (7a): Technology transfer

We feel that successful transfer takes place only when a real need is met. Just to have a clever new toy is not sufficient for a technology to stick. It must meet a real need at a cost appropriate to the user's value in order for transfer to be realized. 1e2b

This need to understand real needs in the outside world and to try to determine how well we are providing value leads to the creation of an analysis function to study needs and analyze how well we are meeting them. We brought in an experienced operations research person with little interest in the technology as a thing in itself to provide this perspective. ##Is this needed?? If so, I think a rewording would be needed ## ## I would prefer deleting it ##

Because of changes in funding levels and pressing needs for trained personnel on the applications side, ## on the development side?? ## we have temporarily halted the Analysis function. Recruiting people with the appropriate interests, training, experience and motivation for the important Analysis function is a difficult task. It is a highly interdisciplinary function and is not easily filled by the present orientations of academic computer science, operations research, or psychology departments. 1e2c

1e2d

Second##comma needed here ## usage by real users with work and applications to do other than build our system are providing us with the feedback and contact with real needs that we feel are necessary to help us operate more on the need pull side of the technology pull-push spectrum. 1e2e

2) One has to know where one is with respect to the two questions: 1e3

Is one trying to ## delete one space##show something is feasible?

Is one trying to show something meets a need and should be continued?

We feel the former was accomplished and that we are in the latter area thus requiring a shift in emphasis from technology push to need pull. 1e3a

3) The ease of technology transfer is proportional to the risk and cost to the user in terms of total system, organization, work habit, otr ##should this be "and"?? ## training

reorganization he has to undergo to adopt the new technology. Technology transfer has been described as more of a battle than just a matter of communicating an idea. Our experience confirms this view. 1e4

To meet this issue we are: 1e4a

a) Asking user groups to have their architect introduce it in areas of least disruption and in a staged way.

b) Asking user organizations not to try and adopt it on a broad scale, but to find a subgroup to try it first, learn the advantages, problems, and create people trained in its use and introduction to take the next steps. 1e4a1

have user grups nd architect been disucssed??? If not maybe a footnote or brief sentence on this; or at least a refernce to another paragraph where it is explained.### 1e4a1a

4) Technology transfer of our type of technology is most successful by transfer of people. Studies at MIT of developments done at MIT and their transfer to industry found that on the order of 90% of the successful transplants were found by students or faculty going to work for the organization, obtaining an understanding of the organization's problems and then bringing in the technology he was familiar with. 1e5

Industrial firms transfer many of their people periodically for just these reasons.

It is not easy to transfer people from SRI to outside groups nor do we have enough people to do that. This problem, when coupled with the motivation of the Gatekeeper concept, has led us to the workshop architect role. 1e5a

In the future when we have our experiment off the ground, we may try to transfer ARC people to user groups for six months to one year, and vice versa. For the past three months and for the coming months, we have stationed one ARC person in Washington, D>C>, ## periods ## where a number of NLS user organizations presently are clustered to provide an approximation to such a role. We have found this close contact very useful and important to the transfer process.

We would like internally to move our people between the Development, Analysis, and Application areas to help them

obtain several points of view, as our technology transfer effort matures. 1e5b

5) To transfer a system such as ours and even many of its ideas requires much more than publishing papers and reports. One needs a gut feeling that only a demonstration or, better yet, hands-on experience can give. This has led us to encourage visitors to ARC and to set up the Utility. One problem we have faced is the task of finding suitable low cost commercially available display terminals for use. Thus, most outside users to date have ##delete one space##had to use the typewriter version which has quite different user characteristics and feel from the display version they see on use at ARC. To make the display version more widely available we have developed a special microcomputer based box for use with commercially available alphanumeric terminals that enables them to be used without modifications as true two-dimensional display NLS workstations. 1e5b1

CONCLUSIONS 1e5b2

Experience to date indicates that the elements of a technology transfer strategy above have put us on the right track, although there is much yet to be learned about the process. It has shown us that technology transfer can be made an explicit, conscious process and that the efficiency and effectiveness of technology transfer can be improved as a result.

1e5b2a

REFERENCES 1e5b3

1) 14724
2) 12445
3) Allen references
4) Andrews Line Processor Paper
5) Norton what we are about paper 1e5b3a

At the heart ## is this to be integrated into the above sectin?? it seems to need a header or something. ## of our views on technology transfer is the belief and experience that the type of information system we are developing can only be developed and evolve in an environment with real users doing their everyday work on the system. We at ARC have been the prime users of the system over the bulk of the project lifetime , but in the last three years have begun to seriously enlist outside users from a variety of organizations. The importance

of obtaining the views and feedback from the users with a variety of needs from many organizational environments is vital to the ongoing healthy evolution of a flexible and general purpose knowledge workshop. Based on this premise we have taken the following steps: 1e6

1) We have organized our internal activities into three areas that we call
Analysis
Development
Applications 1e7

The functioning of these three parts as a harmonious whole constitutes our research process. Development creates new user features, system organizations, and usage methodolgy based on past experience and anticipated needs. Applications provides computer and other services such as training to real users, both internally within the project and to outside groups. Analysis studies, at many levels, system ongoing evolutionary process. 1e7a

2) We have set up an ARPANET connected facility managed by Tymshare ## insert "Inc." to clearly show it is a corporation## at their Cupertino, California, computer Center to serve as a reliable utility for delivery of workshop computer services developed at ARC and elsewhere. The present PDP-10 system is called OFFICE - 1 ## deleete spaces on either side of dash## and is accesible by the ARPANET and directly through low- or high-speed phone lines. As part of the delivery problem we have also developed a low-cost unit called a Line Processor (now commercially available) to support the display version of NLS from low-cost commercially available alphanumeric CRTs. 1e8

The ability to offer reliable computer service is crucial to the development - ##delete sace before this dash##application-analysis strategy. Staff and facilities who have the know-how and motivation to create such a facility are not easily maintained by a highly development and application oriented organization such as ARC. Therefore, an important decision was made two years ago to subcontract computer facility management to a corporation like Tymshare, that has the staff and physical facilities oriented toward providing the needed services. Tymshare is responsible for hardware and operating system reliability. ARC is responsible for all services at higher levels. 1e8a

This has been a valuable and trend-setting move within the ARPA R&D computer community. 1e8b

My comments on final report section (7a): Technology transfer

3) We are asking each subscribing organization to provide what we are calling a "workshop architect", whose prime loyalty is to the using organization (preferably a person from the using organization, although we will provide a person for that role if necessary) to plan a staged introduction (an appropriate profile of services) of the technology and training appropriate to his organization. 1e9

The importance to successful technology transfer of having a person within the target organization who is familiar with his organization's needs and the outside technology has been clearly demonstrated in the works of Allen [XX]. Allen has called such a person a gatekeeper, and has shown that most technology transfer occurs through such people, usually operating on an informal basis. We are trying to formalize and make explicit this role. 1e9a

On the ARC side we have defined a role called a client liaison, whose function is also to be knowledgeable about ARC technology and user needs. It is across the overlapping liaison and workshop architect roles that we hope to achieve effective transfer. These roles are supported by other technical, analysis, and training roles as well. 1e9b

4) The technology was originally developed on the assumption that it would be used as the everyday working environment of its users and that therefore the users would quickly be of the expert category. Experience has shown that --, a) it will probably be some time before this is the case, and b) even where it becomes the case, there is a critical transfer phase. Therefore, we have begun during the past year to pay much more attention to levels of documentation, usage scenarios, help, novice language features, etc., to provide a spectrum of functions from new to experienced users. 1e10

Our experience indicates that concious ## spelling: conscious## attention to technology transfer by an R&D group affect: 1e11

1) Its organizational structure. 1e11a

2) The types of skills and roles needed. 1e11b

3) Its R&D strategy. 1e11c

RLL 17-DEC-74 21:08 24779

My comments on final report section (7a): Technology transfer

(J24779) 17-DEC-74 21:08;;;; Title: Author(s): Robert N. Lieberman/RLL; Distribution: /RWW([ACTION]) ; Sub-Collections: SRI-ARC; Clerk: RLL;

KIRK 17-DEC-74 21:11 24780

Where is the list of completed bug fixes and implemented features?

The ARC feedback process documented in <FEEDBACK,MANUAL,> had simple painless procedures for announcing what bugs had been fixed and what features added. Since the feedback process has been taken over by Applications, I have not been notified of any bug fixes or feature implementations and I do not know where the list is to add to it or document from it. I have the feeling this important information is being lost. If so, I request that the old procedures be followed. If not, please tell me where I can find the list. 1

KIRK 17-DEC-74 21:11 24780

Where is the list of completed bug fixes and implemented features?

(J24780) 17-DEC-74 21:11;;;; Title: Author(s): Kirk E. Kelley/KIRK; Distribution: /FEED([ACTION]) JHB([INFO-ONLY]) RLL([INFO-ONLY]) JCN([INFO-ONLY]) DCE([INFO-ONLY]) RWW([INFO-ONLY]) NPG([INFO-ONLY]) DIRT([INFO-ONLY]) ; Sub-Collections: SRI-ARC NPG DIRT; Clerk: KIRK;

DVN 17-DEC-74 22:06 24781

Request for Help from Lineprocssor Users

I am revising the Lineprocessor User's Guide to accomodate changes in the TIP. 1

I realize that may seem to some of you a little like saying I'm revising the Unicorn. 1a

Nevertheless I wonder if you could help by telling me if the writeup in the next branch jibes with your recent experience. 1b

When the Host Crashes 2

If the host crashes, the TIP will send you the message "Host not responding". 2a

The TIP will keep your connection open for you, but will not tell you when the host coms up. If you want to stay at the workstation in hopes of continuing your work, you must hit <CTRL-C> from time to time as a test. When the Host comes up, it wll respond to <CTRL-C> with the TENEX Login message. When it responds, go back to Step 8 above and continue from there.[Step 8 describes the Terminal Type Lineprocessor Command] < documentation,novguide,2h> 2b

DVN 17-DEC-74 22:06 24781

Request for Help from Lineprocssor Users

(J24781) 17-DEC-74 22:06;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /JOAN([ACTION] dirt notebook please) RLL([ACTION]) JHB([ACTION]) SRL([ACTION]) MAP2([ACTION]) DLS([ACTION]) MEH([INFO-ONLY]) &DIRT([INFO-ONLY]) ; Sub-Collections: SRI-ARC DIRT; Clerk: DVN;

JGR 18-DEC-74 10:03 24783

Search Modification: Request from ISI

Copy of message sent by Jeff Rothenberg to HGL.

JGR 18-DEC-74 10:03 24783

Search Modification: Request from ISI

16-DEC-74 1922-PST ROTHENBERG at USC-ISIB: Search modification
Distribution: NLS-HELP:
Received at: 16-DEC-74 19:23:23 1

The case stuff we need in the Search routine simply requires that case be ignored when searching for some string. That is, the strings "xyz", "Xyz", "XYZ", "xYz", etc. must all be matched by the search command "SEARCH-FOR("xyz"). 1a

We also need a form that allows searching exactly (the current way), so we would have to provide two versions of the appropriate routine with different names, or else put a switch in the calling sequence, as one of the arguments, to specify whether the search should be case-independent or case-dependent. 1b

If this isn't enough detail, let me know. 1c

Jeff 1d

JGR 18-DEC-74 10:03 24783

Search Modification: Request from ISI

(J24783) 18-DEC-74 10:03;;;; Title: Author(s): Jeff Rothenberg/JGR;
Distribution: /RWW([INFO-ONLY]) CHI([INFO-ONLY]) JGR([
INFO-ONLY]) HGL([INFO-ONLY]) EKM([INFO-ONLY]) RLB2([
INFO-ONLY]) DSM([INFO-ONLY]) ; Sub-Collections: NIC; Clerk: HGL;

DVN 19-DEC-74 09:15 24784

For Xmax to Control Page Width

In (24773,) Jon Postel points out an inconsistency in the way the present Output Prcessor Directives control page layout and suggests how it might be solved. There should be an Xmax dirctive like Ymax that would set an absolute page width. I agree with him. It is particularlly important for users away from ARC to be able to control simply page size because they may be printing with devices that have a variety of page images. 1

DVN 19-DEC-74 09:15 24784

For Xmax to Control Page Width

(J24784) 19-DEC-74 09:15;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /FDBK([ACTION]) JOAN([ACTION] dpcs notebook please) DPCS([INFO-ONLY]) ; Sub-Collections: SRI-ARC FDBK DPCS; Clerk: DVN;

APPLICATION'S RECOMMENDATIONS FOR USER PROGRAMS 1

After numerous interactions with the persons concerned, Applications endorses the following scheme for User Programs, class 1, for implementation at Office-1 and documentation in Help. Further developments and modifications should be reviewed by Applications and announced by User Development at the request of the implementors of the changes. 2

These are intended to be coherent and conceptually sensible from the user's point of view. A special emphasis is placed on satisfying user requests and feedback. 3

Subsystems: (* notes those user-progs for which a new CML was required) (In some cases, the command phrase is suggested.) 4

Modify 4a

DELCOL 4a1

Delete Column (beginning at for plex) DESTINATION OK 4a1a

Delete Visible (beginning in column for plex) DESTINATION OK 4a1b

ADDTEXT 4a2

Insert Front/Back STRUCTURE (at) DESTINATION (the text) CONTENT OK/<Filtered: VIEWSPECS OK> 4a2a

DELSP* 4a3

Delete Leading (spaces in) STRUCTURE 4a3a

LOWERCASE* 4a4

Force (Sentence case for:) STRUCTURE ... 4a4a

DELNAME* 4a5

Delete Names (from statements in:) STRUCTURE 4a5a

APPEND 4a6

Append Group (at) DESTINATION (through) DESTINATION (join with) CONTENT OK 4a6a

Format 4b

DELDIR 4b1

APPLICATION'S RECOMMENDATIONS FOR USER PROGRAMS

Delete (Directives in) STRUCTURE (at) DESTINATION OK 4b1a

Set Directive (filter) OK 4b2

Reset Directive (filter) OK 4b3

Insert Directives 4b4

Insert Directives (for File at) DESTINATION (using Format# show list first?) Y/N (format #:) CONTENT (Title:) CONTENT (Author Ident(s):) CONTENT (Journal Number:) CONTENT (Formatting File) 4b4a

Title page 4b4b

(basically SRI format where user specifies his organization, etc) 4b4b1

Generate title page,,% implementation deferred until resources available% 4b4b2

Message 4c

send (Message structure:)* 4c1

(to:) *DIRECTORYNAMES separated by comma OR Idents 4c1a

(subject:) CONTENT 4c1b

(message/structure:) Message/STRUCTURE (B/T:) 4c1c

(type CA to send the message, n to add to list) 4c1d

Move Message 4c2

Copy Message 4c3

<>Sort Message 4c4

ADDRESS* 4c5

Insert (address to follow) STRING DESTINATION OK (Input ident) CONTENT OK 4c5a

JFORM3* 4c6

Reformat (mail plex at:) %Implementation still questionable% 4c6a

publish 4d

APPLICATION'S RECOMMENDATIONS FOR USER PROGRAMS

INDEX 4d1

Index STRUCTURE 4d1a

TOC 4d2

Generate Table (of contents for file at:) 4d2a

WORDCOUNT 4d3

Count (visables in:) STRUCTURE 4d3a

MAKEREF 4d4

Generate References (at:) STRUCTURE 4d4a

Sort Keys: 4e

sortnocase 4e1

sortrev 4e2

sortnum 4e3

sortnmskip 4e4

Content Analyzers 4f

LETTER** %not to be made into a subsystem now by Development-RWW; part of NSW project% 4f1

Generate Letter (for file at:) (using format #:)... 4f1a

Catalog (suggestion for this subsystem to be released later by NDM; alternative subsystem name to "Bibliography". 4g

Insert Journal (catalog)... 4g1

APPLICATION'S RECOMMENDATIONS FOR USER PROGRAMS

(J24785) 19-DEC-74 09:32;;;; Title: Author(s): James H. Bair/JHB;
Distribution: /RWW([ACTION]) KIRK([ACTION]) SRI-ARC([INFO-ONLY
]) ; Sub-Collections: SRI-ARC; Clerk: JHB; Origin: < BAIR,
PROGS,NLS;6, >, 19-DEC-74 08:33 JHB ;;;;####;

DVN 19-DEC-74 09:38 24786

Response to Online Review

Thanks for your response about reviewing Technology Transfer. I glanced at <lieberman,report,> and your comments seemed helpful and sufficient. Dick has been home sick this week and hasn't seen it, but he may have more thoughts when he does. 1

DVN 19-DEC-74 09:38 24786

Response to Online Review

(J24786) 19-DEC-74 09:38;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /JOAN([ACTION] DPPCS notebook please) RWW([INFO-ONLY]) RLL([INFO-ONLY]) ; Sub-Collections: SRI-ARC DPCS; Clerk: DVN;

DVN 19-DEC-74 12:45 24787

Contact with Norn Nielson and Tom Humphry about trip to Gunter.

I'v infomred Norm that the idea is on again, and he seemed pleased. I chatted with Tom who thought time could be made some where in his schedule, but it may be hard when he comes to doing it. I didn't think I could go further until I talked with Dick at greater lenght. 1

DVN 19-DEC-74 12:45 24787

Contact with Norn Nielson and Tom Humphry about trip to Gunter.

(J24787) 19-DEC-74 12:45;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /DCE([INFO-ONLY]) RWW([INFO-ONLY]) ; Sub-Collections: DPCS SRI-ARC; Clerk: DVN;

KIRK 20-DEC-74 01:31 24788

Users' suggestions left out of NLS-8

Jim or Ron, here are the users' suggestions from <arc,nls,mods,> that almost got implemented. The proper place for them now I guess is the requested features branch of the feedback file. They came from the last (and first?) big pow-wow last spring that accepted and rejected suggestions sent to feedback. These are the suggestions that got accepted and are ordered by desireability and difficulty to implement. I would still like to help get something constructive done with the backlog of suggestions in FDBK. There should be a way with our augmentation tools to analyze all of these data and come up with a plan for future development most responsive to users. Lets discuss this when you get back.

KIRK 20-DEC-74 01:31 24788

Users' suggestions left out of NLS-8

EKM 1-MAY-74 MISSING COMMANDS 1

The following commands fell through a crack: 1a

Show Selections 1a1

Show Control Mark 1a2

Show Statement #'s in leveladjust 1a3

Show Upper Case 1a4

Set Tabs 1a5

Show Tabs 1a6

They should be implemented both in the SET and SHOW Editor commands and in the User Options subsystem. 1b

(3a3) -027- Valid alternatives should be available in Help exactly as stated in the response to questionmark, especially such things as <tab>, <insert>, etc., as well as ANY other response, (feedback,fdbk,03191) 2

(We acknowledge that this may require changes to the Help/Query Command recognition algorithm.) Not to be done now 2a

------- Mods above this line are the only ones development has committed itself to implement before October 1. ----- 3

KEV 9-OCT-74 09:22 24171
things i didn't get doen for nls-8
Location: (JJOURNAL, 24171, 1:w)
*****Note: [INFO-ONLY] *****

4

the following is a list of things that were (are) on my todo list for nls-8. due to time pressures and relative priorities they did not get done. i am distributing this in the hopes that maybe others might have some time in the future to implement them. 4a

provide a meta-language to implement an interactive process commands ability 4a1

implement what i call mouse button macro strings: 4a2

this associates a string (kept in user-options) for each of the keyset chords associated with mouse case-shift 101. thus when a user hits a chord on the keyset while mouse

buttons 101 are down, it would be equivalent to entering the string associated with that chord. 4a2a

fix the bug associated with improper prompting if both level-adjust and optional prompting are turned off. 4a3

this bug should go away when we rewrite the interpretter. 4a3a

implement the following commands: 4a4

subsystem XXX: Detach 4a4a

subsystem BASE: 4a4b

Undelete Modifications 4a4b1

" and linefeed in DNLS 4a4b2

Copy, Show, and Expunge Archive Directory 4a4b3

Retreive, Delete, and Undelete Archive File 4a4b4

Show Who is on 4a4b5

Show Where is user 4a4b6

(3c1) -0103- Let words containing hyphens break at the hyphen when it is at the end of a line. (feedback,fdbk,02775) 5

(and commas, and periods?) 5a

From time to time I have had such a long link e. g. (xjournal,12345,1ab) that the disply won't make a reasonable line because it breaks only at invisisbles. If the display looked for ', as well it would save us an occasional really ugly screen. 5b

You might or might not want long numbers with commas to be broken. If you did not want them broken it would be easey to check if a comma were in a number. (21714,) (DVN) 5b1

(3c1) -039- [dsm] Setting name delimiters should change the Statement Signature. (feedback,fdbk,02717) 6

Setting name delimiters does not change the Statement Signature. (22059,) (KIRK) 6a

(3c1) -041- Since a person is told a file is bad, he should also be told it is good. "file verify in progress" is an o.k. message, akin to "output quickprint in progress". (feedback,fdbk,02764) 7

KIRK 20-DEC-74 01:31 24788

Users' suggestions left out of NLS-8

(3c1) -047- <tab> is listed as an alternative in response to questionmark, but if not typed in the proper context it's responded to by "Illegal Search Type", (feedback,fdbk,02651) This err msg should be changed to "<tab> valid only to repeat a previous search" 8

Is it reasonable to get an error message when an alternative is typed, i.e. Illegal search type is displayed when a user types <tab>. (MEJ) 8a

(3c1) -050- The validity of the characters used for name delimiters should be checked as they are typed in, (feedback,fdbk,02718) 9

Check validity of name deliminator when being set in Set command (22802,) (RLL) 9a

(3c1) -056- The herald should be settable as an option to zero length leaving just a prompt, (feedback,fdbk,02907) 10

RLL 15-MAY-74 19:20 23011
herald length zero, why not?
Message: Why can't tthe herald be set to length zero? The prompt (if at least partially on) will be sufficient to key the user, The terse option could be defaoult to be length zero if any prompt is on (partial or full), otherwise leave terse as is when no prompt is on,
*****Note: [ACTION] *****
10a

(3c1) -058- Put the "Process Commands Branch" command in the Editor subsystem, 11

(3c1) -092- Allow viewspec "o" and "p" to be set before the completion of freeze and release commands and let "release all" OPTIONALLy result in viewspec "p", (feedback,fdbk,03249) 12

(4b2) Make a new viewspec which would turn all indenting off putting all text left justified regardless of structure, 13

(6b3) -069- CONFIRM should work for recognition so that <sp> or <esc> is not required when an entire command has been typed, (feedback,fdbk,02709) i.e. have CA as a right delimiter but not swallowed by the CML, 14

(6c2) -0111- Since ;filter; is defined as a viewspec in a link it should be a valid viewspec whenever the prompt V: appears, (feedback,fdbk,02741) 15

(6c2) -059- Resolve the present bug in newnls that makes it impossible to "jump to name" in the identfile when the name is

enclosed in single quotes, i.e. implement by removing the first single quote ' from the last names in the ident file. 16

(6c2) Review TNLS CALCULATOR and DNLS CALCULATOR 17

(6c2) in the output to terminal command add an output to file option which outputs to a sequential file from the output processor. 18

(6c3) -023- Rather than having left-over prompts at the top of the screen, such as "Replace Text at through by through", display the actual text typed, following and on the same line as the respective prompts. Need an appropriate symbol for a bug mark (possibly the word or character if text; the statement number, if structure; or a symbol such as "<bug>"). This would more closely approximate TNLS. (feedback,fdbk,01927), (feedback,fdbk,03236) Development feels that the current implementation of noisewords () is OK, we may display "<bug>" for bug selections. 19

in inset and substitue commands where multiple words or characters can be input the text (s) should appear after word or character (Ithought we had decided to do that may months ago, why didn't we?) (22940,) (RWW) 19a

(9c3) -065- After typing a space and one character, a backspace character should result in your being able to type another second-level command. (feedback,fdbk,03151) 20

(9c3) -070- Have a new user option to set the escape charater (and its echo) to be other than "<sp>". 21

Setting expert second level recognition key from space to period. (22061,) (KIRK) 21a

For the default in TNLS, don't repeat prompts, E.G. Update C: File OK:/C C: Compact OK:. Both C:'s aren't needed (22964,) (RLL) 22

KIRK 20-DEC-74 01:31 24788

Users' suggestions left out of NLS-8

(J24788) 20-DEC-74 01:31;;;; Title: Author(s): Kirk E. Kelley/KIRK; Distribution: /FEED([ACTION]) ; Sub-Collections: SRI-ARC; Clerk: KIRK;

KIRK 20-DEC-74 01:35 24789

exceed capacity bug with sendmail forms

Long comments don't fit in the sendmail form though they are acceptable via the comment command. The comment I was trying to send was less than 800 characters. 1

KIRK 20-DEC-74 01:35 24789

exceed capacity bug with sendmail forms

(J24789) 20-DEC-74 01:35;;;; Title: Author(s): Kirk E. Kelley/KIRK;
Distribution: /FEED([ACTION]) KLM([INFO-ONLY]) ;
Sub-Collections: SRI-ARC; Clerk: KIRK;

DSM 19-DEC-74 21:28 24790

NLS-MAE File Transfer: A Case Study

This document decsribes the most pragmatic method of accomplishing a desired task under the given time contraints, and is not meant to be a general model for file transfer between the two systems. We realize that the general problem is an important one and needs further study from both ends. 1

The need has arisen for a file transfer mechanism between NLS and the MAE text editing system. The immediate need is to transfer a set of NLS files which represent one half of a report (DEIS Report) into MAE where the other half of the report lives for final editing and printing. 2

Transferring Files from NLS to MAE: 3

The first problem was to choose a physical medium for the information transfer which is compatible with both systems. The available mediums on each system are: 3a

MAE I/O devices: 3a1

9-track tape , DEC tape, terminal, and cassette. 3a1a

NLS I/O devices: 3a2

7-track tape , DEC tape, terminal, network, and cassette. 3a2a

DEC tape appears to be the most appropiate medium, however it has the unfortunate problem that it has different formats (including different directory formats) on the PDP-10 and the PDP-11. This problem is solved by the existence of a program called ELEVEN which runs on the PDP-10 and is capable of reading and writing DEC tapes in the PDP-11 DEC tape format. This gives us the ability to write TENEX disk files on the PDP-10 onto a DEC tape which can be read by the PDP-11, or to copy a file from a PDP-11 format DEC tape to a TENEX sequentail disk file on the PDP-10. 3b

The program ELEVEN was tested and used to successfully transfer text between the two systems. It lives in the <subsys> directory at SRI-ARC. Documentation can be obtained by running the program and typing "/H". However we recieved the following warning from the SRI-AI group which uses the program. Do not perform any directory commands on the tape from the PDP-10. That is the PDP-10 can write a file to the tape or read a file from the tape but all other commands such as deleting a file or zeroing the directory should be performed only on the PDP-11. 3c

Examples of using ELEVEN: 3c1

The first step is to load a PDP-11 DEC tape on the PDP-10 DEC tape unit called DTA1: . Then give the TENEX Command: 3c1a

MOUNT DTA1: 3c1a1

Next run the program by giving the TENEX Command: 3c1b

ELEVEN,SAV 3c1b1

You are now taking to the program ELEVEN, Suppose you have a tenex file called TEST,TXT which you want to write to the tape as TAPETEST,TXT the proper command would be: 3c1c

DTA1:TAPETEST,TXT/V/A_TEST,TXT 3c1c1

Suppose there is a file on the tape called MAEFILE,TXT which was written on the PDP-11 and you want to copy it to a TENEX file called MYMAE,TXT the proper command would be: 3c1d

MYMAE,TXT _ DTA1:MAEFILE,TXT/V/A 3c1d1

Notice that the program always assumes your connected directory, 3c1e

Sequentializing the NLS file: 3d

NLS files must be made into sequential files before they can be put onto DEC tape and be subsequently read on the ELEVEN, I will try to sumarize the available methods of performing this task and give my recommendations for the proper method, 3d1

NLS Sequential Output Modes: 3d2

Output Sequential File: 3d2a

This command is similiar to Output Quickprint, Statemnts are broken into lines and blanks are inserted for indention, The output is not passed through the Output Processor, Therefore any output Processor Commands remain in the file as text, Lines are delimited by <CR> <LF>, 3d2a1

Output Printer File: 3d2b

The output is passed through the Output processor and stored on a file, Lines are terminated by a <^S> (This is avoid extra line feeds which are generated by our printer whenever it notices that there has been 55 <LF>'s printed without an intervening <^L> or form feed), 3d2b1

Output Terminal File: 3d2c

The output is passed through the Output Processor and stored on a file. Lines are terminated by <CR><LF>. Page Breaks are contained in the file as text (several blank lines followed by a page number followed by a line of text "----------". 3d2c1

The question of which kind of output file to generate is really a question of how you want to split the task of editing the file between the two systems. One possibility would be to do all the editing using NLS and the Output processor and hope the file did not require any additional editing on the MAE system. The other extreme would be to simply load the text of an NLS file somehow into MAE and use MAE to massage the file into final form. The problem with this approach is that NLS does not currently contain a command which will turn the file into a sequential file without doing some kind of editing, for example inserting line breaks and adding blank padding for indentation. As a result the MAE system must process the file to remove these unwanted additions. It would be fairly easy to add the appropiate output sequential command to NLS and perhaps this possibility should be investigated further. 3d3

The best approach using only the existing capabilities appears to be the following. 3d4

Use the NLS Output processor directives to format the file as closely as possible to the final desired output from MAE. Some suggested directives would be: 3d4a

IREST=0 to turn off indenting of successive lines of each statement 3d4a1

LBS=1 to insure that there is at least one blank line between statements so that the MAE system can delimit statements easily. 3d4a2

Use the Command OUTPUT PRINTER FILE to produce the sequential file. Such a sequential file has <"S>'s instead of <CR><LF> to indicate line breaks. This will make it easier for the MAE system to preprocess the text to assemble the lines back into the MAE equivalent of statements, called blocks. <CR> and <LF> are special control characters in MAE which would complicate this assembly process. 3d4b

There remians a question as to whether or not it would be better to have the page headings and footers be produced by the Output processor. If they are produced by the Output

processor we have the problem of aligning the NLS pages with the MAE pages, and also making sure that the headers are identical to the headers which already exist in the MAE half of the file. If headers are not generated by the Output Processor they will have to be inserted by hand in the MAE system. Further study by the people who will be responsible for preparing the final document is needed here to determine which way minimizes the amount of hand editing required. 3d4c

DSM 19-DEC-74 21:28 24790

NLS-MAE File Transfer: A Case Study

(J24790) 19-DEC-74 21:28;;;; Title: Author(s): David S. Maynard/DSM; Distribution: /MAP2([ACTION]) RWW([INFO-ONLY]) JCN([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: DSM;

garage

Concrete Floor	1
Exisiting floor has cracks, would cover with new concrete.	1a
Extend Wiring	2
Existing wall sockets are mid-wall height, would place at near floor level.	2a
Attach Sink to existing sewer	3
Put windows in garage door and wall	4
To alllow more natural light in the graage.	4a
Insulate walls and ceiling	5
To conserve heat.	5a
Cover walls with sheet rock	6
Cover ceiling with ?	7

garage

(J24791) 20-DEC-74 08:57;;;; Title: Author(s): Jonathan B. Postel/JBP; Sub-Collections: SRI-ARC; Clerk: JBP; Origin: < POSTEL, GARAGE.NLS;2, >, 20-DEC-74 08:56 JBP ;;;;####;

Please review and comment. This document describes the interconnection between the user and system code within a PCP (Tenex) process. Holes? Blunders? Also, how well does this strategy map over to the ELF implementation?

PREFACE 1

The Procedure Call Protocol (PCP) is an inter-process and/or inter-host protocol that permits a collection of processes within one or more ARPANET hosts to communicate at the procedure call level. In effect, it makes the component procedures of remote software systems as accessible to the programmer as those within his own system. PCP specifies both a virtual programming environment (VPE) in which remote procedures may be assumed to operate, as well as the inter-process exchanges that implement it. 1a

The Multi-Process Software System (MPSS) whose construction PCP makes practical and of which the NSW is an example, consists of collections of "procedures" and "data stores" called "packages", in one or more "processes", interconnected in a tree structure by "physical channels". Procedures within a process have free access to the procedures (and data stores) of each process adjacent to it in the tree structure, and may call upon them as if they were local subroutines. Superimposed upon the tree structure is a more general set of interconnections which give non-adjacent processes in the tree the same kind of access to one another. 1b

The MPSS is implemented by: 1c

1) low-level protocols which provide the basic, inter-process communicaton (IPC) facilities by which channels are implemented: an inter-host IPC protocol (PCPHST), an inter-Tenex-fork IPC protocol (PCPFRK), and data structure format specifications for both connection types (PCPFMT). 1c1

2) PCP proper, which largely defines the VPE (especially, the procedure call and return mechanism) and specifies the inter-process control exchanges required to implement it. 1c2

3) a set of system packages, implemented within each process, which augment PCP proper by providing mechanisms by which user procedures can: call remote procedures (implemented by the Procedure Interface Package, PIP), manipulate remote data stores (implemented by the PCP Support Package, PSP), and interconnect processes (implemented by the Process Management Package, PMP). 1c3

4) user packages in each process. 1c4

INTRODUCTION 2

This document defines the internal structure of a PCP process implemented to run on Tenex, and as such serves as a process implementer's guide. It describes the process' fork structure, the role and composition of each fork, and the manner in which the various forks interact with one another; indicates which components are supplied with PCP and which are the responsibility of the process implementer; and describes the manner in which the components are assembled at load time. 2a

FORK STRUCTURE OF THE PROCESS 3

A PCP process implemented on Tenex is a multi-fork structure consisting of a top-level fork called the "controlling fork" (CF), and one or more directly inferior forks called "processor forks" (PFs). Each PF is connected to the CF via a physical channel which the CF creates and manipulates using its IPC (PCPFRK) procedures. 3a

No ambiguity results from the fact that the IPC procedures are used both to create and manipulate the process, and to create and manipulate forks within the process, since each knows the context in which it is created and the role it is to play. 3a1

The CF serves as the process' point of connection to the MPSS tree, and manages the physical channels which connect the process to its direct superior, its direct inferiors, and any processes to which it is introduced. In this capacity, it queues and routes the PCP messages output by its PFs, and processes PCP and IPC messages sent to the process by neighboring processes, assigning PFs to procedure call requests, and interrupting and aborting their execution. 3b

While it is running, a PF is effectively a PCP processor. The CF assigns it a procedure to execute (via a CALPRO message); it executes the procedure, and signals its completion via a RTNPRO message. A PF is only serially reentered by the CF in this way; it is never called upon to execute more than one procedure at a time. While a PF is NOT running, either because it has made a temporary return or because it has been interrupted by the CF in response to an INTPRO request, it serves not as a processor but as a container in which the state of the procedure call is preserved. 3c

COMPOSITION OF EACH FORK 4

Modules 4a

The CF and PFs, all of which are instances of a single SAV file, each contain the following modules: 4a1

System Modules 4a1a

These modules are implemented in L10, the current "PCP implementation language" (PIL), and are supplied by the PCP implementer as a single REL file. They include: 4a1a1

1) system packages (i.e. PIP, PSP, and PMP),

2) system subroutines which user modules may call,

3) the "processor fork environment" (PFE) which encapsulates within the PF both user and all other system modules, and which receives control from the CF when the PF is created, and

4) the "controlling fork environment" (CFE) which encapsulates within the CF both user and all other system modules, and which receives control from the process' superior when the local process (i.e. the CF) is created.

User Modules 4a1b

These modules are implemented in an arbitrary programming language, and are supplied by the process implementer as one or more REL files. They include: 4a1b1

1) user packages, and

2) user subroutines which system modules may call,

Address Space Composition 4b

The SAV file which implements the process, behaving either as the CF, under control of the CFE, or as a PF, under control of the PFE, contains the following in its address space: 4b1

1) Executable code and static tables 4b1a

   One or more READ/COPY-ON-WRITE/EXECUTE pages of executable code and static tables, of which the CF and all PFs share a single copy. These pages are initialized by the local process' superior via the GET JSYS, as prescribed by PCPFRK. 4b1a1

2) Dynamic Tables 4b1b

   Zero or more READ/WRITE/EXECUTE pages of dynamic tables within the CF which are shared by each PF. The CF initializes these pages when the process is created and terminates the process on the basis of the state information they contain, when the process is deleted. The CF causes the PFs to share these pages of their address spaces, and the CF and PFs modify them throughout the process' lifetime. 4b1b1

3) Fork state information 4b1c

   One or more READ/WRITE/EXECUTE pages of fork-local storage, an individual copy of which exists for the CF and each PF. The CF or PF initializes these pages as required at run-time. 4b1c1

4) IPC (PCPFRK) windows 4b1d

   One or more READ/WRITE/EXECUTE pages shared between the CF and another fork (either a PF, the superior process, or an inferior or introduced process). The CF establishes these windows (via the IPC CRTPRC and CRTCHNEND procedures) as required at run-time. 4b1d1

OPERATION OF THE CONTROLLING FORK 5

Process Initialization and Termination 5a

The CF initializes the process' dynamic tables when the process is created, and by appropriately modifying the PF's page table, causes them to be shared by each new PF that it creates. The CF terminates the process on the basis of the state information contained in its dynamic tables, when the process is deleted by its superior. 5a1

Since the CF assumes responsibility for overall process initialization and termination, the PF need only initialize when created by the CF, any fork state information it requires, and terminate itself on the basis of that state information when deleted by the CF. 5a2

Inter-Process Communication 5b

The CF manages all of the physical channels by which the process is connected to the surrounding MPSS, in addition of course to those which connect the CF to its PFs. Included are the physical channels to the process' direct superior, the process' direct inferiors (if any), and all of the processes to which the local process has been introduced (if any). 5b1

Assuring that all of these physical channels are attached to the CF, rather than to the PF to which execution of the PMP procedure which creates the channel happens to be assigned, requires the cooperation of PMP. Specifically, it demands that whenever PMP requires one of the procedures in the IPC package, it use the CF's copy, rather than its own. A PF addresses the CF using the pseudo-PH -2, and its IPC package using the PKH -2; the CALPRO message which it issues to invoke the IPC procedure must have the priviledged bit set. 5b2

The CF implements the PCP message relaying facility required by PMP. 5b3

Processor Assignment and Release 5c

The CF monitors each of the physical channels attached to it -- those to its processors and those to other processes. Whenever it receives a PCP CALPRO message whose PH field contains SELF, it selects an "idle" PF, previously created, initialized, and attached to the CF via a physical channel. The CF associates the selected PF with the CH specified in the CALPRO message, marks the PF "running", and forwards the message to the PF for processing. The CF will not assign a second CALPRO message to that PF until it signals a permanent return. 5c1

When the processor signals a temporary return, the CF simply marks the PF "suspended" and forwards the RTNPRO message; the PF remains exempt from reassignment. When the CF receives a RSMPRO message specifying the appropriate CH, it again marks the PF running, and forwards the RSMPRO message to it. 5c2

When the processor signals a permanent return, the CF terminates the association between PF and CH, marks the PF idle and available for reassignment, and forwards the RTNPRO message. 5c3

The PF thus serves two distinct purposes. While running, it represents a processor. While suspended, it serves as a container in which the state of the processor is maintained. The number of processors within the process is therefore an upper bound on the number of PFs that may be either running or suspended simultaneously. 5c4

Processor Interruption and Abortion 5d

The CF processes PCP INTPRO, RSMPRO, and ABRPRO messages. It freezes the appropriate PF in response to an INTPRO message, thaws it in response to RSMPRO, and aborts the ongoing procedure call in response to ABRPRO. 5d1

To avoid interrupting a procedure call (and thereby exposing it to abortion) while it is engaged in a crucial series of operations, the CF, before freezing the PF, exclusively locks the FRZLCK lock which it shares with the PF to which it assigned the procedure call (locks are discussed later). A procedure may thus assure the completion of a crucial section of code by locking the lock upon entry to it (and unlocking it upon exit). 5d2

Once the CF has succeeded in locking the lock, it freezes the PF and acknowledges the INTPRO request in accordance with PCP. If a RSMPRO request follows the interruption, the CF simply thaws the PF and unlocks the lock. If it receives an ABRPRO request instead, the CF thaws the PF and forwards to it the abort request (whose parameters the PF ignores). 5d3

When it completes execution of the procedure assigned to it, the processor sends a RTNPRO message to the CF, which forwards it to the calling process and marks the PF free. 5d4

USER- AND SYSTEM-SUPPLIED SUBROUTINES 6

Introduction 6a

User and system modules are interfaced to one another by means of "system-supplied subroutines" (SSSs) provided by the PCP implementer and accessible to user modules, and "user-supplied subroutines" (USSs) provided by the process implementer and accessible to system modules. The process of transferring control to a SSS or USS requires the saving and restoring of both user- and system-environment state information. 6a1

All of the currently-defined USSs and SSSs are described in this section. 6a2

General Calling Sequence 6a3

To invoke a USS from a system module, or a SSS from a user module, the caller constructs the required argument list, saves its environment's state, stores the address of the argument list in R0, the address of the subroutine in R1, and transfers control to the "user/system entry point" (UEP/SEP, described later) by means of a "JSP 2, ..." instruction. 6a3a

When it receives control, the entry-point code saves the subroutine's argument list and return addresses, loads its environment's state, calls the requested subroutine, and on the basis of its outcome constructs a parameter list of the following form: 6a3b

Word 0 Type (of return) 6a3b1
Word 1 Subtype (of return) 6a3b2
Word 2- Result list 6a3b3

It them saves its environment's state, stores the address of the parameter list in R0, and returns control to the address stored by the JSP instruction, by means of a JRST instruction. The caller then restores its environment's state and proceeds with its execution. 6a3c

Shortcut for Some User Modules 6a4

For those user modules implemented, like the system modules, in PIL, the general, low-level calling sequence described above can be skipped, and the user module may, if it wishes, call the appropriate high-level PIL routine directly (e.g. CALPRO, rather than SYCALL). 6a4a

System-Supplied Subroutines 6b

The System Descriptor 6b1

The addresses of all SSSs are made available to user modules by means of the "system descriptor", addressed by the external symbol "SySDSC". The system descriptor has the following format: 6b1a

Hword	0	SYVERS	PCP version number	6b1a1
Hword	1	SYLANG	System implementation language code	6b1a2
Hword	2	SYSEP	Addr of system entry point (SEP)	6b1a3
Hword	3	SYSGNL	Addr of signal subroutine	6b1a4
Hword	4	SYWAIT	Addr of wait subroutine	6b1a5
Hword	5	SYLOCK	Addr of lock subroutine	6b1a6
Hword	6	SYUNLK	Addr of unlock subroutine	6b1a7
Hword	7	SYCALL	Addr of call subroutine	6b1a8
Hword	8	SYRSM	Addr of resume subroutine	6b1a9
Hword	9	SYINT	Addr of interrupt subroutine	6b1a10
Hword	10	SYABR	Addr of abort subroutine	6b1a11

SYLANG contains a code for the system modules' implementation language. The subroutines currently addressed by the system descriptor are described below. 6b1b

PIP Interface 6b2

Introduction 6b2a

The subroutines described in this section provide an interface to PIP for those user modules that wish to call remote procedures and/or procedures within other packages in the local process. 6b2a1

Subroutines 6b2b

Call remote procedure 6b2b1

SYCALL (Ph, Pkh, Pname, Args, Arglmsk, Reslmsk, Mode, CmpleVnt => Type, Subtype, Results, Ch)

This subroutine is equivalent in function to PIP's CALPRO procedure.

Argument and result list formats:

Word 0 Ph
Word 1 Pkh
Word 2 Pname (ASCIZ string pointer)
Word 3 Args (addr of PCPB36 data structure)
Word 4 Arglmsk (addr of PCPB36 data structure)
Word 5 Reslmsk (addr of PCPB36 data structure)
Word 6 Mode [INLINE=1 / OUTOFLINE=0]
Word 7 CmpleVnt (addr of ECB or zero)

Word 0 Type
Word 1 Subtype (or zero)
Word 2 Results (addr of PCPB36 data structure)
Word 3 Ch (or zero)

Resume remote procedure 6b2b2

SYRSM (ch, args, mode, cmplevnt -> type, subtype, results)

This subroutine is equivalent in function to PIP's RSMPRO procedure.

Argument and result list formats:

Word 0 Ch
Word 1 Args (addr of PCPB36 data structure)
Word 2 Mode [INLINE=1 / OUTOFLINE=0]
Word 3 Cmplevnt (addr of ECB or zero)

Word 0 Type
Word 1 Subtype (or zero)
Word 2 Results (addr of PCPB36 data structure)

Interrupt remote procedure 6b2b3

SYINT (ch)

This subroutine is equivalent in function to PIP's INTPRO procedure.

Argument list format:

Word 0 Ch

Abort remote procedure 6b2b4

SYABR (ch)

This subroutine is equivalent in function to PIP's ABRPRO procedure.

Argument list format:

Word 0 Ch

Events 6b3

Introduction 6b3a

The subroutines described in this section permit components of the process to synchronize their execution. These subroutines are used extensively by system modules and may be used by user modules as well. 6b3a1

pIp's CALPRC procedure signals the completion of a remote procedure called out-of-line by means of the SYSGNL subroutine described below. SYSGNL is also used to signal the arrival of a PCP message on a physical channel. 6b3a2

Event Control Blocks 6b3a3

An event is represented within the process by a single-word "event control block" (ECB), which has the following format:

Hword 0	EVWFRK	Fork handle of waiting fork, as seen by the CF / zero
Hword 1	EVSTS	Status of event [COMPLETE=1 / INCOMPLETE=0]

To define and use an event, the user or system module allots space for the ECB, either at compile- or run-time, initializes it by setting it to zero, and uses its address as an argument to the signal subroutines described below.

If the ECB resides in one of the pages shared by the CF and all PFs, the event can be used to synchronize forks within the process; otherwise, it can only be employed to synchronize modules within a single fork.

Subroutines 6b3b

Signal event 6b3b1

SYSGNL (event)

This subroutine signals event EVENT. If SYSGNL is called again before EVENT has been detected by a call to SYWAIT, it is a NOP.

Argument list format:

```
Word 0  Event (addr of ECB)
```

Wait for event 6b3b2

SYWAIT (events -> event)

This subroutine waits the local fork until one of the events EVENTS, namely EVENT, is signalled. The CF calls this subroutine to await input from one of the physical channels it manages; user modules may employ it as they desire.

Argument list format:

```
Word 0  Event count
Word 1  Event (addr of ECB)
   ... (other ECB addresses)

Word 0  Event (addr of ECB)
```

Locks 6b4

Introduction 6b4a

The subroutines described in this section permit forks within the process to synchronize their manipulation of shared data bases (e.g. data stores). These subroutines are used extensively by system modules and may be used by user modules as well. 6b4a1

A fork may lock a lock for either read or write. In the former case, application of the lock assures the fork that no other fork has (or can) apply a write lock to it until it is unlocked. In the latter case, the fork is assured that no other fork has (or can) apply ANY lock to it until it is unlocked. 6b4a2

Lock Control Blocks 6b4a3

A lock is represented within the process by a "lock control block" (LCB), which has the following format:

```
Hword 0   LKTYP   Type of lock [READ=0 / WRITE=1]
Hword 1   LKSTS   Status of lock [UNLOCKED=-1 /
                   LOCKED=0 / ENQUEUED=1]
Hword 2   LKRLC   Read lock count (for CF)
Hword 3   LKWLC   Write lock count (for CF)
   ... (for each PF)
Word  2   LKWTCT  Number of forks awaiting lock
Hword 6           Fork handle of waiting fork
Hword 7           Requested lock type
   ... (for each additional fork awaiting the
        lock)
```

LKRLC and LKWLC record the depth to which (i.e. how many consecutive times) the corresponding fork has locked the lock in the indicated manner.

To define and use a lock, the user or system module allots space for the LCB, either at compile- or run-time, initializes it by setting it to zero (except LKSTS, which is set to -1), and uses its address as an argument to the lock subroutines described below.

LCBs must reside in one of the pages shared by the CF and all PFs for the locks to be of any practical utility to the process.

Subroutines 6b4b

Set inter-fork lock 6b4b1

SYLOCK (lock, type, cmplevnt)

This subroutine applies a lock of the type indicated by TYPE to LOCK. If CMPLEVNT is not specified, the subroutine will fail if the lock is currently locked by another fork in a way that precludes its being locked immediately by the caller. Otherwise, the subroutine will queue the request if necessary and signal event CMPLEVNT when the lock has been set.

Argument list format:

```
Word 0  Lock (addr of LCB)
Word 1  Type [READ=1 / WRITE=0]
Word 2  Cmplevnt (addr of ECB or zero)
```

Remove inter-fork lock 6b4b2

SYUNLK (lock)

This subroutine removes from LOCK the lock most recently applied by the calling fork.

Argument list format:

```
Word 0  Lock  (addr of LCB)
```

User-Supplied Subroutines 6c

The User Descriptor 6c1

The addresses of all USSs are made available to system modules by means of the "user descriptor", addressed by the external symbol "USRDSC". The user descriptor has the following format: 6c1a

Hword 0	USNAME	String pointer to ASCIZ process name	6c1a1
Hword 1	USUEP	Address of user entry point (UEP)	6c1a2
Hword 2	USPRCT	Number of processors supported by process	6c1a3
Hword 3	USBUCT	Number of bundles within the process	6c1a4
Hword 4		Addr of bundle descriptor	6c1a5
... (other bundle descriptor addresses)			

USNAME is the formal name of the process, returned by PMP's CRTPRC procedure. USBUCT is the number of bundles within the process, and therefore the number of half-word entries which complete the process descriptor. Bundles are explained in the section that follows. 6c1b

Bundles 6c2

Bundle Descriptors 6c2a

The system modules are implemented in a single programming language (PIL) which, in general, differs from that in which user modules are implemented. Furthermore, not all user modules need be implemented in the same language. 6c2a1

In general, a process is constructed by the loader from one or more REL files, each containing a "bundle" of modules, all of which are implemented in the same language. Associated with each bundle is a unique run-time environment whose state must be initialized with the process. 6c2a2

Each bundle within the process is described by a variable-length "bundle descriptor" of the following form: 6c2a3

Hword 0	BULANG	Bundle implementation language code
Hword 1	BUINTM	Addr of bundle initiator/terminator
Hword 2	BUPDSP	Addr of procedure dispatcher
Hword 3	BUDSMN	Addr of data store manipulator
Hword 4	BUPKCT	Number of packages within the bundle
Hword 5		Addr of package descriptor
... (other package descriptor addresses)		

BULANG contains acode for the bundle's implementation language. BUPKCT is the number of packages within the bundle, and therefore the number of half-word entries which complete the bundle descriptor. 6c2a4

Subroutines 6c2b

Initialize or terminate bundle 6c2b1

BUINTM (mode, frzlck)

This subroutine initializes or terminates the bundle, according to MODE, and will be called by a system module just after the CF or PF is created and again just before it is deleted. It should perform whatever fork (but not process) initialization or termination functions are required by the bundle. When MODE is INITIALIZE, FRZLCK specifies the address of the FRZLCK lock.

Argument list format:

```
Word 0  Mode [INITIALIZE=1 / TERMINATE=0]
Word 1  Frzlck (addr of FRZLCK lock) / zero
```

Call procedure within bundle 6c2b2

BUPDSP (pkcode, pname, arglist => reslist)

This subroutine effects a call with arguments ARGLIST to procedure PNAME in the package whose code PKCODE is specified, and returns the results RESLIST it generates. It is also responsible for establishing the bundle's run-time environment.

Argument and result list formats:

```
Word 0  Pkcode
Word 1  Pname (ASCIZ string pointer)
Word 2  Arglist (addr of PCPB36 data structure)

Word 0  Reslist (addr of PCPB36 data structure)
```

Resume last-called procedure within bundle 6c2b3

BUPRSM (arglist -> reslist)

This subroutine effects a resumption with new arguments ARGLIST of the last-called procedure within the bundle (assuming it returned temporarily), and returns the results RESLIST it generates. It is also responsible for establishing the bundle's run-time environment.

Argument and result list formats:

Word 0 Arglist (addr of PCPB36 data structure)

Word 0 Reslist (addr of PCPB36 data structure)

Read or write data store within bundle 6c2b4

PKDSMN (pkcode, dsname, mode, newvalue -> curvalue)

This subroutine effects a manipulation of data store DSNAME in the package whose code PKCODE is specified. If MODE is READ, it returns the value CURVALUE of the data store (NEWVALUE is ignored). If MODE is WRITE, it assigns the value NEWVALUE to the data store DSNAME (CURVALUE is ignored).

The PH and PKH fields of DSNAME are to be ignored by PKDSMN.

Argument and result list formats:

Word 0 Pkcode
Word 1 Dsname (addr of PCPB36 dselector*)
Word 2 Mode [READ=1 / WRITE=0]
Word 3 Newvalue (addr of PCPB36 data structure)
/ zero

Word 0 Curvalue (addr of PCPB36 data structure)
/ zero

Packages 6c3

Package Descriptors 6c3a

Each package within a bundle is described by a double-word "package descriptor", addressable by convention via an external symbol whose name is that of the package (truncated to six characters, if necessary), and having the following format: 6c3a1

Hword 0	PKNAME	String pointer to UC ASCIZ package name
Hword 1	PKINTM	Addr of package initiator/terminator
Hword 2	PKOPCL	Addr of package opener/closer
Hword 3	PKCODE	Package code

PKNAME is the formal name of the package, by which it is addressed in PSP's OPNPKS procedure. PKCODE is a code by which the package initiator/terminator and opener/closer can index to or otherwise locate its own internal tables for the package. 6c3a2

Subroutines 6c3b

Initialize or terminate package 6c3b1

PKINTM (pkcode, mode =>sharedpages)

This subroutine initializes or terminates, according to MODE, the package whose code PKCODE is specified. If MODE is INITIALIZE, the subroutine indicates to its caller the set of address space pages to be shared by the CF with each PF. PKINTM will be called by a system module just after the CF is created and just before it is deleted, and should perform whatever package initialization and termination functions are required for the process.

Argument list format:

Word 0 Pkcode
Word 1 Mode [INITIALIZE=1 / TERMINATE=0]

Word 0 Sharedpages (XWD firstpage,lastpage)

Open or close package 6c3b2

PKOPCL (pkcode, mode)

This subroutine opens or closes, according to MODE, the package whose code PKCODE is specified. It will be called by a system module whenever the package is opened or closed, and should perform whatever package initialization and termination functions are required by the process, and implement whatever access controls are to be applied to the package.

Argument list format:

Word 0 Pkcode
Word 1 Mode [INITIALZE=1 / TERMINATE=0]

JEW 20-DEC-74 11:04 24792

Tenex PCP Process Internal Structure
PCPTNXINT Version 2

19-DEC-74

James E. White
Augmentation Research Center

Stanford Research Institute
Menlo Park, California 94025

PCPTNXINT defines the internal structure of a PCP process implemented to run on Tenex, and as such serves as a process implementer's guide. The reader is assumed familiar with the Procedure Call Protocol (PCP -- 24459,).

DRAFT JEW 20 DEC 74 7:48PM Tenex PCP Process Internal Structure

(J24792) 20-DEC-74 11:04;;;; Title: Author(s): James E. (Jim) White/JEW; Distribution: /NPG([ACTION]) ; Sub-Collections: SRI-ARC NPG; Clerk: JEW; Origin: < WHITE, PCP-TNXINT,NLS;22, >, 20-DEC-74 10:51 JEW ;;;; ####;

Software consideration for electrostatic printers.

This is in response to DVN's question about font posting.

Software consideration for electrostatic printers.

Software implications of the inclusion of an electrostatic printer 1

The Gould, Versetec, and Varian Printer/Plotters operate in two modes. 1a

As printers, internal font sets are used to convert incomming ASCII characters to raster output. The user has no access to this font set. Essentially, the device replaces a standard impact printer. 1a1

As plotters, each line of styli is controlled directly by input from the interface computer. As a result character to raster conversion is completely under user control. 1a2

In our case, a PDP 11 will be connected between the source of information and the printer/plotter. This configuration insures that sufficient buffer space and computational power can be applied to the incomming information. 1b

In my view this device (printer and cmputer) should be used to form the basis of a virtual print device. 1c

In addition to the printer, interface to all COM devices should be required to use this virtual device, as should all graphics hardcopy output (for example mechanical plotters). 1c1

Software for the system will require the following: 1d

A definition of the virtual printer/plotter. 1d1

This definition would be a near variant of the current virtual COM device with the addition of improved page spacial concepts and inclusion of internal justification. (That is, the input would not be prejustified by the output processer for example.) 1d1a

Interface to the CMU font definition formats. 1d2

The CMU XGP code provides a font description format which we could adopt. This means that the virtual printer would accept its fonts in the CMU format and that font design routines would produce output in this format. It is reasonable to assume that at least several font sets now exist in this form. 1d2a

PDP 11 implementation of the virtual printer. 1d3

This is relatively straightforward code once the virtual printer specifications are formalized. The exact

implementation for random to raster conversion for line drawings is the most troublesome of any of the alied problems. 1d3a

Interface to existing programs. 1d4

Of course, existing routine would need to be connected to the virtual printer. This largely impacts only the output processor. For the most part, the new graphics code will follow this concept anyway. Since the virtual definition will probably default (in some sence) to a line printer, routines like output quickprint would be affected very little if at all. 1d4a

Software consideration for electrostatic printers.

(J24793) 20-DEC-74 15:25;;;; Title: Author(s): Robert Louis Belleville/RLB2; Distribution: /DPCS([INFO-ONLY]) ; Sub-Collections: SRI-ARC DPCS; Clerk: RLB2; Origin: < BELLEVILLE, ELECTROSTATIC-PRINTERS.NLS;1, >, 20-DEC-74 15:16 RLB2 ;;;;####;

DVN POOH 20-DEC-74 16:52 24794

Informal Documentation Report for Week Ending Dec 20th

pooh-continued work on help and the glossary. I have finished about 70 pages on a first time through basis. attended class on the o.p, set up a plan for an ident a sendmessage address for documentation requests. 1

KIRK Reviewed and trained with POOH on Help 2

JMB Worked on Help examples, talked with Susan, others about what ARC's person in Washington needs inthe way of documentation and documentation procedures. 3

DVN: 4

Lineprocessor Usuer's Guide: Copies ready to be proofed for printer errors. 4a

Preface to NLS: Waiting for Application's Review 4b

Introduction to NLS (replacing the Howto branch of help):Waiting to br written. 4c

TNLS Addressing: It is on me to repsond to RWW's review. 4d

COM: 4e

Ordered DDSI to Make Camera Ready Copy of The Mouse and Keyset cards. 4e1

Martin Hardy's paper Microprocessor Techology is still waiting for DDSI to get stick fonts working. 4e2

The command Summary, the TNLS-8 Primer, and Ken Victor's paper on the CML await my attention to revise them for COM printing. 4e3

Talked on the phone to a number of possible aplicants for documentation work here; asked several to fill in applications. 4f

DVN POOH 20-DEC-74 16:52 24794

Informal Documentation Report for Week Ending Dec 20th

(J24794) 20-DEC-74 16:52;;;; Title: Author(s): Dirk H. Van Nouhuys, Ann Weinberg/DVN POOH; Distribution: /JOAN([ACTION] dirt notebook please) DIRT([INFO-ONLY]) ; Sub-Collections: DIRT SRI-ARC; Clerk: DVN;

DVN 20-DEC-74 17:00 24795

Day Off Tuesday

I plan to take Tuesday the 24th off and be here the rest of the working days in the next two weeks. 1

DVN 20-DEC-74 17:00 24795

Day Off Tuesday

(J24795) 20-DEC-74 17:00;;;; Title: (Unrecorded) Title:
Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /RWW([INFO-ONLY])
POOH([INFO-ONLY]) KIRK([INFO-ONLY]) JMB([INFO-ONLY]) ;
Sub-Collections: SRI-ARC; Clerk: DVN;

JMB 23-DEC-74 09:43 24796

Re Kirk's--24780,> Notices of bug fixes, etc.

I also miss the notices of bug fixes and implemented features. This info is necessary for documenting (as well as for other roles), so if it is not announced, we have to scrounge for it some other way. 1

JMB 23-DEC-74 09:43 24796

Re Kirk's--24780,> Notices of bug fixes, etc.

(J24796) 23-DEC-74 09:43;;;; Title: Author(s): Jeanne M. Beck/JMB;
Distribution: /FEED([ACTION]) JHB([INFO-ONLY]) KIRK([INFO-ONLY
]) RLL([INFO-ONLY]) JCN([INFO-ONLY]) DCE([INFO-ONLY]) RWW(
[INFO-ONLY]) NPG([INFO-ONLY]) DIRT([INFO-ONLY]) ;
Sub-Collections: SRI-ARC NPG DIRT; Clerk: JMB;

network journal delivery

23-DEC-74 09:03:49,951
Net mail from site PARC-MAXC rcvd at 23-DEC-74 09:03:47
Date: 23 DEC 1974 0904-PST
From: DEUTSCH at PARC-MAXC
Subject: Cryptic Journal announcement messages
To: lampson
cc: taft, postel at ARC

1

JBP 23-DEC-74 09:45 24797

network journal delivery

(J24797) 23-DEC-74 09:45;;;; Title: Author(s): Jonathan B. Postel/JBP; Distribution: /FEED([ACTION]) FDBK([ACTION]) BUGS([ACTION]) ; Sub-Collections: SRI-ARC FDBK BUGS; Clerk: JBP;

network journal delivery

LPD 23-DEC-74 09:03:49,951
Net mail from site PARC-MAXC rcvd at 23-DEC-74 09:03:47
Date: 23 DEC 1974 0904-PST
From: DEUTSCH at PARC-MAXC
Subject: Cryptic Journal announcement messages
To: lampson
cc: taft, postel at ARC 1

I agree that the format used for on-line notifications of Journal distribution is surpassingly obscure. I think if it says INFO-ONLY, it means that the author (in this case JBP) just wants you to know that the document exists -- you can retrieve it from SRI-ARC via a rigamarole that I have buried in my message file somewhere.
The NIC has, to my knowledge, never sent out anything that explains what all the different variants and components of these messages mean, nor (to my recollection) how to retrieve a journal document of which you only have the citation -- Jon had to explain that in the message that accompanied the announcement of the NSW Protocol documents. I'll dig up the retrieval procedure and let you know. ------- 1a

JBP 23-DEC-74 09:55 24798

network journal delivery

(J24798) 23-DEC-74 09:55;;;; Title: Author(s): Jonathan B. Postel/JBP; Distribution: /FEED([ACTION]) FDBK([ACTION]) BUGS([ACTION]) ; Sub-Collections: SRI-ARC FDBK BUGS; Clerk: JBP;

Business Cards

We are going to print new business cards very soon. I have samples of ten different formats. Anyone who wishes to order a card should come see me to choose a style and content. Orders will be taken through the first week in January. 1

Business Cards

(J24799) 23-DEC-74 11:01;;;; Title: Author(s): Ann Weinberg/POOH; Distribution: /SRI-ARC([ACTION]) ; Sub-Collections: SRI-ARC; Clerk: POOH;

DVN 23-DEC-74 11:40 24800

Title Word Indices Don't Work Because of Changed Statement Nmae Delimiters.

The catalog system duplicates the titlewords and the names of authors in parentheses at the beginning of each statment in the online indices so that they may serve as statement names. Then jump to itme taks you where you want to go. But Now the default name delimiters are NULL NULL, and jump to item no longer works. 1

DVN 23-DEC-74 11:40 24800

Title Word Indices Don't Work Because of Changed Statement Nmae Delimiters.

(J24800) 23-DEC-74 11:40;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /FDBK([ACTION]) JCP([ACTION]) JCN([INFO-ONLY]) DSM([INFO-ONLY]) ; Sub-Collections: SRI-ARC FDBK; Clerk: DVN;

DVN 23-DEC-74 19:40 24801

More on Xmax

References:(24773,) (24784,) and (31544,)

DVN 23-DEC-74 19:40 24801

More on Xmax

Dean, in thinking about Xmax, did you conider Jon's proposal on Standard File Formats (gjournal,31524,)? 1

More on Xmax

(J24801) 23-DEC-74 19:40;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /NDM([ACTION]) JOAN([ACTION] dpcs notebook please) FDBK([ACTION]) DPCS([INFO-ONLY]) JBP([INFO-ONLY]) ; Sub-Collections: SRI-ARC FDBK DPCS; Clerk: DVN;

A Move is a combination of Delete and Copy except...

I don't know of anyone who has not unexpectedly been moved to a place they did not want to be because a statement they were moving happened to contain their command marker. There is no real good way of getting around this phenomenon if you are sorting a list into different branches. It is a pain in the ass. The Move command should work exactly like a combination Delete and Copy which means you should be left at the first character of the next statement after the "deleted" statement(s). 1

KIRK 23-DEC-74 20:04 24802

A Move is a combination of Delete and Copy except...

(J24802) 23-DEC-74 20:04;;;; Title: Author(s): Kirk E. Kelley/KIRK; Distribution: /FEED([ACTION]) JBP([INFO-ONLY]) SRI-ARC([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: KIRK;

JCN 5-JAN-75 02:58 24803

Accomplishments During 1974 Relating to the
ARC OFFICE-1 Workshop Utility Service

Connie: Maybe this adds something for you to use..? Hope so Jim

Accomplishments During 1974 Relating to the
ARC OFFICE-1 Workshop Utility Service

CONTENTS Statement Number 1

CONTENTS 1

I BACKGROUND 2

Augmented Knowledge Workshop (AKW) Technology 2a
ARC's "Community Plan" 2b
The Significance to the Department of Defense 2c

II THE WORKSHOP UTILITY SERVICE 3

ARC is organized into three main activities 3a
The purpose of the Workshop Utility Service 3b
Computer and Technical Services 3c

III ACCOMPLISHMENTS -- Activity in 1974 4

Summary 4a
A. OFFICE-1 Service via the ARPANET 4b
B. Development and Use of new transfer methods 4c
C. The Building of a User Community 4d

IV SELECTED REFERENCES 5

Accomplishments During 1974 Relating to the
ARC OFFICE-1 Workshop Utility Service

I BACKGROUND 2

Augmented Knowledge Workshop (AKW) Technology 2a

The Augmentation Research Center (ARC) of Stanford Research Institute (SRI) has developed, over a period of more than twelve years under government sponsorship (primarily from ARPA, but also from NASA, AFOSR, and RADC), a general-purpose interactive augmentation system centering about what we now call an "Augmented Knowledge Workshop," abbreviated below as "workshop." The goal of ARC's work has been to evolve a prototype Workshop system that will significantly improve the performance of individuals and teams engaged in knowledge-work activities, where the Workshop "system" involves daily use of coordinated tools, procedures, methodologies, and languages. 2a1

For further background discussion, see [1], [2], and [3]. 2a2

While the discussion in [2] is oriented toward communities of discipline- or mission-oriented users, the same types of services and knowledge workshop orientation apply to individuals and groups of workers in local environments. 2a3

ARC's "Community Plan" 2b

Introduction 2b1

ARC is a one-organization community of researchers and system developers, supported by several different contracts. The research and development activities of ARC are aimed at exploring the possibilities for augmenting individuals and groups in the performance of knowledge work with the help of computer aids. These aids range from off-line batch to on-line real-time. Exploratory development and operation of augmentation systems have been our substantive work. 2b1a

ARC's Research and Development Strategy 2b2

A new stage of application has now been established with the first year of a new "Workshop Utility Service"--described below and also in more detail in [4]--almost completed. We are involving a wider group of system users so that we can begin to transfer the results of our past work to others, and so that we can obtain feedback needed for further evolution from wider application than is possible in our Center alone. We

have been providing Workshop Utility Service to selected groups who are willing to take extra trouble to be exploratory, but who: 2b2a

1) are not necessarily oriented to being workshop system developers (they have their own work to do), 2b2a1

2) can see enough benefit from the system's application and from the experience of trying it so that they can justify the extra risk and expense of being "early users," and 2b2a2

3) can accept our assurance that reliability, system stability, and technical application help will be available to meet their conditions for risk and cost. 2b2a3

Establishment of a Workshop Utility and promotion of the type of service work described herein are part of ARC's long-term commitment to pursue the continued development of augmented knowledge workshops in a pragmatic, evolutionary manner. Our last few years of work have concentrated on the means for delivering support to a distributed community, for providing teleconferencing and other basic processes of collaborative dialogue, etc.--consciously aiming toward having experience and capabilities especially applicable to support remote and distributed groups of exploratory users. 2b3

The significance to the Department of Defense and to the public in general is difficult to overstate when judged in terms of the potential improvement in effectiveness and efficiency this new technology will offer to individuals and groups engaged in knowledge-work. Administrators, planners, analysts, lawyers, doctors, writers, secretaries, managers, scientists, educators, and many other roles in society will benefit from use of knowledge workshop technques now in the early stages of development. 2c

II THE WORKSHOP UTILITY SERVICE
A transfer and application of new technology 3

ARC is organized into three main activities: Development, Analysis, and Applications. 3a

The Development activity is responsible for the development of new or changed system features, including software, hardware and methodology domains. 3a1

The Analysis activity assists both the Development and Applications efforts at many levels and, though presently staffed at a very low level, is an essential part of the operation. 3a2

The Applications activity is responsible for delivering the workshop technology (as it develops) to a growing user community. The primary responsibility of the ARC Applications group is the provision of the new Workshop Utility Service. 3a3

The purpose of the Workshop Utility Service that began in January 1974 is to deliver useful advanced Workshop Utility computer and related technical services to subscribing organizations' users while concurrently providing the system developers (through Analysis) with useful information about further system development needs based on the real experiences of users in their work environments. The service is being provided to organizations that are willing to undertake exploratory use of knowledge workshop techniques through continued use of the on-line system (NLS) at OFFICE-1. 3b

The service is composed of two primary activities: computer services and technical services. 3c

The computer services are being supplied through the ARPANET to geographically distributed user groups from the OFFICE-1 computer facility maintained and operated by Tymshare, Inc, in Cupertino, California, under a subcontract with ARC. 3c1

Technical services are provided by ARC personnel in the following areas: 3c2

Maintaining and updating the "utility" version of ARC's application software (NLS). 3c2a

Supporting the user groups in learning how to use these

tools, both at the individual user level and at the organizational application level. 3c2b

Assisting clients in obtaining advanced display terminal and teleprinter hardware and the necessary ARPANET connections. 3c2c

Obtaining user reactions to system features and the service itself and integrating these into the system development process. 3c2d

Descriptions of the applications available for exploratory use are given in a paper by Engelbart, Watson, and Norton [3] and in an earlier paper by Engelbart [2]. 3d

The types of workshop services that we are beginning to support at varying levels of capability are described in [2] under the headings: 3d1

Collaborative Dialogue 3d2
Document Development, Production, And Control 3d3
Research Intelligence 3d4
Community Handbook Development 3d5
Computer-Based Instruction 3d6
Meetings And Conferences 3d7
Community Management And Organization 3d8
Special Knowledge Work By Individuals And Teams 3d9

Accomplishments During 1974 Relating to the
ARC OFFICE-1 Workshop Utility Service

III ACCOMPLISHMENTS -- Activity in 1974 4

Summary 4a

A. Workshop Utility Service is being provided to users in their own work environments and is, in effect, an evolving delivery of results after a significant government-supported research and development investment. 4a1

B. Development and Use of new transfer methods are progressing. 4a2

C. Many organizations are actively using the Workshop Utility. 4a3

A. OFFICE-1 Service started 18 January 1974 via the ARPANET 4b

The computer and technical services offered as the Utility commenced are being provided with increasing effectiveness with the insight gained through our experience. We have interacted with over 200 users and their organizations and have, to some extent, strengthened their appreciation for the potential benefits of the workshop technology through discussion and through their actual experience in using the system. This service not only provides assistance to users in their work and feedback of useful information into the further development process, but also takes the form of a real delivery of the results of government-supported research and development to government workers and, as the user community grows, to the public in general. 4b1

Tymshare and the ARPANET uptime records 4b2

Constant computer access availability and high system reliability are key requirements for the successful introduction of the workshop technology to new users. The first year of the service has produced an uptime record of over 98% for the Tymshare-operated computer system. The ARPANET record is comparable. 4b2a

We have gained valuable experience in the provision of service based on the new workshop technology. We and others in the field will build on this experience to provide increasingly more effective and efficient advanced services in the future. 4b3

B. Development and Use of new teaching methods 4c

This endeavour involves transfer of an advanced technology from our local group of experienced users to a much wider group of less experienced, geographically separate users. The technology consists of on-line software capabilities; a coordinated repertoire of on-line-assistance tools; associated concept and language additions dealing with the tools and with the information organization and task processes associated with their use; and new aspects to intragroup organization and working methodology. Training users in these new matters is necessary to the transfer; and to help others learn to train people in the new technology requires a transfer of the additional technology used to support the training. We are in the process of further developing these capabilities. 4c1

Some of the techniques we are using in the transfer process are: 4c2

Linked-terminal training sessions 4c2a

Use of terminals in small "classroom" training sessions 4c2b

Individual tutoring at user sites 4c2c

Special training of subscribing organizations' Workshop Architects 4c2d

On-line "user-to-supplier" communication (feedback) methodology 4c2e

C. The Building of a User Community 4d

The following organizations are now using the Workshop Utility: 4d1

RADC Rome Air Development Center (Air Force) 4d2

Over 40 users at RADC concentrating on management system use, software engineering, and document production with the goal of matching the capabilities of NLS and its related methodologies to Air Force "knowledge-worker" needs. 4d2a

Bell Canada Business Planning Group 4d3

About 10 users at Bell concentrating on online communications and document production with the goal of gaining first-hand experience with these new techniques

and assessing the possibilities for and impacts on communications services that may be provided in the future. Their access to OFFICE-1 is via direct telephone lines, rather than via the ARPANET. 4d3a

ARPA General ARPA use and National Software Works 4d4

Over 30 ARPA people have started using NLS in their work during the past few months. We expect an increasing use in program management activities by ARPA people, using techniques based on the capabilities in NLS and on specially developed methodology. In addition, the ARPA/Air Force National Software Works (NSW) program has just begun and is expected grow into a significant effort with NLS Office-1 use as a core for many developmental and communication functions. 4d4a

CBI ARPA: Computer-Based Instruction Community 4d5

ARPA-sponsored research contractors in the CBI community are beginning to use NLS as the core service in the design of their community's online information needs. 4d5a

Energy ARPA: SRI Energy Project 4d6

The ARPA/SRI Defense Energy Information System (DEIS) design effort has been using NLS for communication and general file handling. 4d6a

NIC ARPA: Network Information Center Users 4d7

This is the set of ARPA Network Information Center (NIC) users who were previously been served through the SRI-ARC computer. Their specialized online NIC service is now being provided from OFFICE-1 (over 40 user sites). The data base is being produced and accessed through NLS. 4d7a

Seismic ARPA: Seismic Data Mgt System Development 4d8

The Seismic Data Management System Development (SDMS) effort, part of the ARPA VELA program, is beginning to use NLS as the basis of dialogue among participants in the VELA program and as the basis for a set of files that will aid users of the Seismic Data system to find information about resources that will enable them to use the data being collected by the system. 4d8a

BRL Ballistic Research Laboratories (Army) 4d9

About 10 users at BRL are starting to explore application of Workshop technology to their operations. Document production, team dialogue, and personal information management are most likely initial areas of use. 4d9a

Hudson Hudson Institute (ARPA subcontract) 4d10

Hudson is starting to explore application of Workshop technology to the online and hardcopy production of foreign country profile documents under an ARPA contract. NLS will also provide a communications link with their ARPA project monitors. 4d10a

NSRDC Naval Ship Research and Development Center 4d11

Over 20 users at NSRDC are starting to explore application of Workshop technology to their operations. Document production, team dialogue, and personal information management are most likely initial areas of use. 4d11a

SRI Stanford Research Institute 4d12

Other SRI users are starting to explore application of Workshop technology to their own operations. Document production, distributed project team dialogue, and personal information management are most likely initial areas of use. 4d12a

NSA National Security Agency 4d13

Over 20 users at NSA are starting to explore application of Workshop technology to their operations related to the design of a computer network. Document production, distributed project team dialogue, information center services and personal information management are most likely initial areas of use. 4d13a

A few other prospective applications under consideration: 4d14

AFAA Air Force Audit Agency 4d15

DDC Defense Communications Agency 4d16

DDC Defense Documentation Center 4d17

Accomplishments During 1974 Relating to the
ARC OFFICE-1 Workshop Utility Service

DL-OSH	Dept of Labor-Occup, Safety & Health	4d18
NIOSH	National Institute of Occupational Safety and Health:	4d19
NSF	National Science Foundation	4d20
PTI	Public Technology, Inc,	4d21
USGS	U.S. Geological Survey	4d22

IV SELECTED REFERENCES 5

1 ARC 3906, D. C. Engelbart, "Augmenting Human Intellect: A Conceptual Framework," Summary Report, Contract AF 49(638)-1024, SRI Project 3578, Stanford Research Institute, Menlo Park, California, AD 289 565, October 1962. 5a

2 ARC 12445, D. C. Engelbart, "Coordinated Information Services for Discipline- and Mission-Oriented Communities," Stanford Research Institute, Augmentation Research Center, 12 December 1972.
Also published in "Time Sharing: Past, Present, Future," Proceedings of the Second Annual Computer Communications Conference at California State University, San Jose, California, January 24-25 1973, pp 2.1-2.4. 5b

3 ARC 14724, D. C. Engelbart, R. W. Watson, J. C. Norton, "The Augmented Knowledge Workshop," AFIPS Proceedings National Computer Conference, June 1973. 5c

4 ARC 24725, J. C. Norton, "Proposal ISU 74-254, Continued NLS Workshop Support for ARPA," Stanford Research Institute, 17 December 1974. 5d

JCN 5-JAN-75 02:58 24803

Accomplishments During 1974 Relating to the
ARC OFFICE-1 Workshop Utility Service

(J24803) 5-JAN-75 02:58;;;; Title: Author(s): James C. Norton/JCN;
Distribution: /CKM([INFO-ONLY]) DLS([INFO-ONLY]) ;
Sub-Collections: SRI-ARC; Clerk: JCN; Origin: < NORTON,
IPTREPORT.NLS;1, >, 4-JAN-75 12:50 JCN ;;;; ####;

KIRK 23-DEC-74 23:46 24804

Load File nasty bug

Takes you to an old version. I'm using WORK but I suspect it is true of the running NLS. It only seems to happen if there is a gap between the version numbers. The oldest file gets loaded. For instance, after using the Copy file command from xhelp, I have two files named help. The old one with version 90 and a new one with version 124. Load file invariably gets me version 90. Nasty. 1

KIRK 23-DEC-74 23:46 24804

Load File nasty bug

(J24804) 23-DEC-74 23:46;;;; Title: Author(s): Kirk E. Kelley/KIRK;
Distribution: /JDH([ACTION]) FEED([ACTION]) BUGS([ACTION]) ;
Sub-Collections: SRI-ARC BUGS; Clerk: KIRK;

KIRK 23-DEC-74 23:54 24805

Bug with load file turns out to be a bug with the copy command

If you don't specify an extention and version, Copy File assigns it the same version number for the old but gives the new NLS file THREE SPACES for an extention. This was causing what I thought was a load file problem. Copy file should use the same extention as the source file if none is specifed. 1

KIRK 23-DEC-74 23:54 24805

Bug with load file turns out to be a bug with the copy command

(J24805) 23-DEC-74 23:54;;;; Title: Author(s): Kirk E. Kelley/KIRK;
Distribution: /FEED([ACTION]) JDH([ACTION]) BUGS([ACTION]) ;
Sub-Collections: SRI-ARC BUGS; Clerk: KIRK;

A Multi-Level Integrated User Help System

The multi-level integrated user help system consists of: 1

1. An automatic alternatives generator and displayer (in response to questionmark) 2

2. An Automatic Syntax Generator
 > user-readable conscise but complete command syntax descriptions generated automatically from the Command Meta Language Code by pushing the SYNTAX button <CTRL-S>. 3

3. A HELP button <CTRL-Q>
 > automatically finds appropriate prose descriptions related to the user's current context by analyzing his current command state. 4

4. A Help command to get the definition of any special term and the description of any command by simply typing in words. It was also written to serve as a free-access computer based learning tool allowing the user, in this case, to teach himself how to use NLS. 5

5. A comprehensive online super-document containing complete up-to-date documentation of the NLS command language which also serves as the Help database. 6

6. A 150 page hardcopy user's glossary automatically generated from this super-document <userguides,glossary,>. 7

7. Various other documentation. (see POOH or DvN. <WEINBERG, DOCUMENTATION,> contains a rough list. 8

A Multi-Level Integrated User Help System

(J24806) 24-DEC-74 01:17;;;; Title: Author(s): Kirk E. Kelley/KIRK;
Distribution: /DCE([ACTION]) DIRT([INFO-ONLY]) JOAN([INFO-ONLY
] a dirt document) ; Sub-Collections: SRI-ARC DIRT; Clerk: KIRK;

Right Margin Control

Note that three optput processor directives are needed to control the right margin RM HJRM and SNF. Also note that these are relative to LMBase so that if someone (eg the Journal) adjusts LMBase the margin will not be where it was expected, It is critical to users with other
output devices than our line printer to be able to easily obtain files to print on their devices that do not exceed the hardware enforced margins of those devices.
--jon.

1

Right Margin Control

(J24808) 24-DEC-74 10:13;;;; Title: Author(s): Jonathan B. Postel/JBP; Distribution: /FEED([ACTION]) FDBK([ACTION]) NDM([ACTION]) DVN([INFO-ONLY]) ; Sub-Collections: SRI-ARC FDBK; Clerk: JBP;

DCE 24-DEC-74 11:32 24809

To FDBK re CONAN compile bug

The pattern in the following statement produces an immediate error message "NLS internal error; string too long" when I do a Compile Content-analyzer-pattern, bugging in front of the "9CH" word. If I put a semicolon after the second visible of the pattern (so the pattern is -- xxx), the same thing happens. This short pattern will compile perfectly well in another statement. Then, if I delete the rest of the pattern text, from the start of the "[EOL]" visible to the semicolon, leaving the EOL and "Note: ...." part, then compiling will work. CHI looked over the problem, did a bit of NDDT looking. Apparently the error hits before the compiler even gets brought in. 1

SNDMSG by A to B: 9CH ("DCE") [EOL] ["tion:"] $=(EOL/"FARBER") ("FARBER");
Note: AUthor Names both in UPPER CASE. 2

DCE 24-DEC-74 11:32 24809

To FDBK re CONAN compile bug

(J24809) 24-DEC-74 11:32;;;; Title: Author(s): Douglas C. Engelbart/DCE; Distribution: /FDBK([ACTION]) ; Sub-Collections: SRI-ARC FDBK; Clerk: DCE;

RLL 24-DEC-74 11:37 24810

Sug: A new textual entity called 'LINE'

This is only a temporarzy type entity since the graphics capbility might eliminate its need. Should be great for tables. Becasue of grphic entity LINE might be changed to some other name. Any suggestions?

RLL 24-DEC-74 11:37 24810

Sug: A new textual entity called 'LINE'

How about a new noun called "Line" (second level 'L). Line would be relative to the display or typewriter terminal at which the user is connected. 1

This would enable easy editing of tables and layouts of hardcopy documents (if user could arbitrarily give a width). Windows would be another way of limiting the length of the line. 2

One could delete, move, insert, etc. a "Line" like any other textual entity. 3

It is recognized that this might be a bit hard unless there is a parameter (easily kept) to check for current line length in specified window. 4

Sug: A new textual entity called 'LINE'

(J24810) 24-DEC-74 11:37;;;; Title: Author(s): Robert N. Lieberman/RLL; Distribution: /FEED([ACTION]) NP([ACTION]) JCN([INFO-ONLY]) JAKE([INFO-ONLY]) JHB([INFO-ONLY]) NDM([INFO-ONLY]) RLB2([INFO-ONLY]) ; Sub-Collections: SRI-ARC NP; Clerk: RLL;

JEW 24-DEC-74 13:19 24811

Support for Kirk's Comments (24802,) on MOVE

Right on.

1

JEW 24-DEC-74 13:19 24811

Support for Kirk's Comments (24802,) on MOVE

(J24811) 24-DEC-74 13:19;;;; Title: Author(s): James E. (Jim) White/JEW; Distribution: /SRI-ARC([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JEW;

PREFACE 1

The Procedure Call Protocol (PCP) is an inter-process and/or inter-host protocol that permits a collection of processes within one or more ARPANET hosts to communicate at the procedure call level. In effect, it makes the component procedures of remote software systems as accessible to the programmer as those within his own system. PCP specifies both a virtual programming environment (VPE) in which remote procedures may be assumed to operate, as well as the inter-process exchanges that implement it. 1a

The Multi-Process Software System (MPSS) whose construction PCP makes practical and of which the NSW is an example, consists of collections of "procedures" and "data stores" called "packages", in one or more "processes", interconnected in a tree structure by "physical channels". Procedures within a process have free access to the procedures (and data stores) of each process adjacent to it in the tree structure, and may call upon them as if they were local subroutines. Superimposed upon the tree structure is a more general set of interconnections which give non-adjacent processes in the tree the same kind of access to one another. 1b

The MPSS is implemented by: 1c

1) low-level protocols which provide the basic, inter-process communicaton (IPC) facilities by which channels are implemented: an inter-host IPC protocol (PCPHST), an inter-Tenex-fork IPC protocol (PCPFRK), and data structure format specifications for both connection types (PCPFMT). 1c1

2) PCP proper, which largely defines the VPE (especially, the procedure call and return mechanism) and specifies the inter-process control exchanges required to implement it. 1c2

3) a set of system packages, implemented within each process, which augment PCP proper by providing mechanisms by which user procedures can: call remote procedures (implemented by the Procedure Interface Package, PIP), manipulate remote data stores (implemented by the PCP Support Package, PSP), and interconnect processes (implemented by the Process Management Package, PMP). 1c3

4) user packages in each process. 1c4

INTRODUCTION 2

The File Package (package name = FP == 24582,) contains those procedures and data stores which a remote process requires to employ the file storage and transfering services of the local process. The package contains procedures for opening, closing, and listing directories, for creating, deleting, and renaming files, and for outputting, updating, and deleting files and elements of files. It also contains data stores of directory and file descriptors. 2a

This appendix contains some comments on implementation strategy. The thrust is to argue that the file package as specified is near minimal and that the conversion between the PCP format and the internal storage format can be encapsulated into a few subroutines. 2b

PROCEDURES 3

The procedures defined in the File Package are near the minimal set needed for any file manipulation system, however the features of some procedures for manipulating elements of a file may be more sophisticated than are necessary in the first year National Software Works (NSW). 3a

Thus in the calls GETFIL and PUTFIL it might be convenient to restrict the argument FILEELM to describe only whole files not elements of files. One approach to this would be to restrict the definition of FSELECTOR* to whole files. Another approach (the one we prefer) is to implement the procedure calls so that when ever a procedure needs to refer to a file as described by a FILEELM argument it calls an internal (to the package) procedure passing as arguments the FILEELM and a source or destination parameters. 3b

This leaves the main procedure (GETFIL or PUTFIL) to check the attributes of the file and the caller for access control and accounting purposes and to open the file. 3c

GETFIL (fileelm, disp, dst, dstype => value) 3c1

Get file performs the access and accounting functions then calls as a subroutine the procedure GETIT passing along the arguments which describe the destination of the file and the element processing as well as a handle on the open file. 3c2

GETIT (fileelm, disp, dst, dstype => value) 3c2a

This procedure outputs a copy of an element FILEELM (which may be the whole file) of the currently open file in one of the local process's previously-opened directories (implicitly named by FILEELM), to a destination DST whose nature is specified by DSTYPE, Note that the arguments include file and connection handles, in TENEX these correspond to job file numbers (JFNs). 3c2b

PARM: the file element is to be returned to the caller as VALUE (i,e, as a result of the procedure).

FILE: the file element is to replace the current value of an element DSTELM of a file in one of the local process's previously-opened directories (implicitly named by DSTELM). The invoking process must have write access to the destination file.

The file element is either replaced by EMPTY (i,e, moved) or left unchanged (copied), according to DISP. To move the element, the invoking process must have write access to the file.

NETC: the file element is to be transmitted via a network connection, to socket SOCKET at host HOST, using format FORMAT (same as for LSTDIR).

CHNL: the file element is transmitted via a PCP channel attached to the port identified by the port handle PROH of the local process. (Channels and ports are discussed in the Process Management Package document.)

Argument/result types: 3c2c

```
fileelm= LIST ( %file handle% INTEGER, <BOOLEAN
          [KEY=TRUE / INDEX=FALSE]> %element% any /
          INTEGER, ...)
disp   = INTEGER [DELETE=0 / RETAIN=1]
dstype = INTEGER [PARM=0 / FILE=1 / NETC=2 / CHNL=3]
   PARM: dst= EMPTY
   FILE: dst= %dstelm% LIST ( %file handle% INTEGER,
```

```
                    <BOOLEAN [KEY=TRUE / INDEX=FALSE]>
                    %element% any / INTEGER, ...)
          NETC: dst- %connection handle% INTEGER
          CHNL: dst- %porh% INTEGER
        value  - any / EMPTY
```

PUTFIL (fileelm, disp, src, srctype) 3c3

Put file performs the access and accounting functions then calls as a subroutine the procedure PUTIT passing along the arguments which describe the source of the file and the element processing as well as a handle on the open file. 3c4

PUTIT (fileelm, disp, src, srctype) 3c4a

This procedure replaces an element FILEELM (which may be the whole file) of the currently open file in one of the local process's previously-opened directories (implicitly named by FILEELM), from a source SRC whose nature is specified by SRCTYPE, Note that the arguments include file and connection handles, in TENEX these correspond to job file numbers (JFNs). 3c4b

PARM: the source is SRC (i.e, an argument of the procedure).

FILE: the source is the current value of an element SRCELM of a file in one of the local process's previously-opened directories (implicitly named by SRCELM). The invoking process must have read access to the source.

The source element is either replaced by EMPTY (i.e, moved) or left unchanged (copied), according to DISP. To move the source element, the invoking process must have write access to the source file.

NETC: the source will be transmitted via a network connection, from socket SOCKET at host HOST, using format FORMAT (same as for LSTDIR).

CHNL: the file element is transmitted via a PCP channel attached to the port identified by the port handle PORH of the local process. (Channels and

ports are discussed in the Process Management Package document.)

Argument/result types: 3c4c

```
fileelm= LIST ( %file handle% INTEGER, <BOOLEAN
         [KEY=TRUE / INDEX=FALSE]> %element% any /
         INTEGER, ...)
disp   = INTEGER [DELETE=0 / RETAIN=1]
srctype= INTEGER [PARM=0 / FILE=1 / NETC=2 / CHNL=3]
   PARM: src= any
   FILE: src= %srcelm% LIST ( %file handle% INTEGER,
              <BOOLEAN [KEY=TRUE / INDEX=FALSE]>
              %element% any / INTEGER, ...)
   NETC: src= %connection handle% INTEGER
   CHNL: src= %porh% INTEGER
```

There are two other procedures in the File Package which might not be necessary in the initial NSW, these are delete element (DELELM) and get file structure type (GETST). 3d

To allow maximum progress on the implementation of the File Package under the TENEX Operating System it is proposed that bulk of the File Package be implemented by BBN but that the routines GETIT and PUTIT be implemented by SRI. 3e

The File Package Appendix
FP-APP Version 2

23-DEC-74

Jon Postel
Augmentation Research Center

Stanford Research Institute
Menlo Park, California 94025

The File Package (FP -- 24582) is a file manipulation tool that operates within the setting provided by the Procedure Call Protocol (PCP -- 24459,), with which the reader of the present document is assumed familiar.

(J24812) 24-DEC-74 13:33;;;; Title: Author(s): Jonathan B. Postel/JBP; Distribution: /NSW([INFO-ONLY]) SRI-ARC([INFO-ONLY]) ; Sub-Collections: SRI-ARC NSW; Clerk: JBP; Origin: < POSTEL, NSW-FILE-APP.NLS;2, >, 23-DEC-74 11:53 JBP ;;;; ####;

PREFACE 1

The Procedure Call Protocol (PCP) is an inter-process and/or inter-host protocol that permits a collection of processes within one or more ARPANET hosts to communicate at the procedure call level. In effect, it makes the component procedures of remote software systems as accessible to the programmer as those within his own system. PCP specifies both a virtual programming environment (VPE) in which remote procedures may be assumed to operate, as well as the inter-process exchanges that implement it. 1a

The Multi-Process Software System (MPSS) whose construction PCP makes practical and of which the NSW is an example, consists of collections of "procedures" and "data stores" called "packages", in one or more "processes", interconnected in a tree structure by "physical channels". Procedures within a process have free access to the procedures (and data stores) of each process adjacent to it in the tree structure, and may call upon them as if they were local subroutines. Superimposed upon the tree structure is a more general set of interconnections which give non-adjacent processes in the tree the same kind of access to one another. 1b

The MPSS is implemented by: 1c

1) low-level protocols which provide the basic, inter-process communicaton (IPC) facilities by which channels are implemented: an inter-host IPC protocol (PCPHST), an inter-Tenex-fork IPC protocol (PCPFRK), and data structure format specifications for both connection types (PCPFMT). 1c1

2) PCP proper, which largely defines the VPE (especially, the procedure call and return mechanism) and specifies the inter-process control exchanges required to implement it. 1c2

3) a set of system packages, implemented within each process, which augment PCP proper by providing mechanisms by which user procedures can: call remote procedures (implemented by the Procedure Interface Package, PIP), manipulate remote data stores (implemented by the PCP Support Package, PSP), and interconnect processes (implemented by the Process Management Package, PMP). 1c3

4) user packages in each process. 1c4

INTRODUCTION 2

The File Package (package name = FP == 24582,) contains those procedures and data stores which a remote process requires to employ the file storage and transfering services of the local process. The package contains procedures for opening, closing, and listing directories, for creating, deleting, and renaming files, and for outputting, updating, and deleting files and elements of files. It also contains data stores of directory and file descriptors. 2a

This appendix contains some comments on implementation strategy. The thrust is to argue that the file package as specified is near minimal and that the conversion between the PCP format and the internal storage format can be encapsulated into a few subroutines. 2b

PROCEDURES 3

The procedures defined in the File Package are near the minimal set needed for any file manipulation system, however the features of some procedures for manipulating elements of a file may be more sophisticated than are necessary in the first year National Software Works (NSW). 3a

Thus in the calls GETFIL and PUTFIL it might be convenient to restrict the argument FILEELM to describe only whole files not elements of files. One approach to this would be to restrict the definition of FSELECTOR* to whole files. Another approach (the one we prefer) is to implement the procedure calls so that when ever a procedure needs to refer to a file as described by a FILEELM argument it calls an internal (to the package) procedure passing as arguments the FILEELM and a source or destination parameters. 3b

This leaves the main procedure (GETFIL or PUTFIL) to check the attributes of the file and the caller for access control and accounting purposes and to open the file. 3c

GETFIL (fileelm, disp, dst, dstype => value) 3c1

Get file performs the access and accounting functions then calls as a subroutine the procedure GETIT passing along the arguments which describe the destination of the file and the element processing as well as a handle on the open file. 3c2

GETIT (fileelm, disp, dst, dstype -> value) 3c2a

This procedure outputs a copy of an element FILEELM (which may be the whole file) of the currently open file in one of the local process's previously-opened directories (implicitly named by FILEELM), to a destination DST whose nature is specified by DSTYPE, Note that the arguments include file and connection handles, in TENEX these correspond to job file numbers (JFNs). 3c2b

PARM: the file element is to be returned to the caller as VALUE (i.e. as a result of the procedure).

FILE: the file element is to replace the current value of an element DSTELM of a file in one of the local process's previously-opened directories (implicitly named by DSTELM). The invoking process must have write access to the destination file.

The file element is either replaced by EMPTY (i.e. moved) or left unchanged (copied), according to DISP. To move the element, the invoking process must have write access to the file.

NETC: the file element is to be transmitted via a network connection, to socket SOCKET at host HOST, using format FORMAT (same as for LSTDIR).

CHNL: the file element is transmitted via a PCP channel attached to the port identified by the port handle PROH of the local process. (Channels and ports are discussed in the Process Management Package document.)

Argument/result types: 3c2c

```
fileelm- LIST ( %file handle% INTEGER, <BOOLEAN
           [KEY=TRUE / INDEX=FALSE]> %element% any /
           INTEGER, ...)
disp   - INTEGER [DELETE=0 / RETAIN=1]
dstype - INTEGER [PARM=0 / FILE=1 / NETC=2 / CHNL=3]
   PARM: dst- EMPTY
   FILE: dst- %dstelm% LIST ( %file handle% INTEGER,
```

```
                <BOOLEAN [KEY=TRUE / INDEX=FALSE]>
                %element% any / INTEGER, ...)
        NETC: dst- %connection handle% INTEGER
        CHNL: dst- %porh% INTEGER
      value  - any / EMPTY
```

PUTFIL (fileelm, disp, src, srctype) 3c3

Put file performs the access and accounting functions then calls as a subroutine the procedure PUTIT passing along the arguments which describe the source of the file and the element processing as well as a handle on the open file. 3c4

PUTIT (fileelm, disp, src, srctype) 3c4a

This procedure replaces an element FILEELM (which may be the whole file) of the currently open file in one of the local process's previously-opened directories (implicitly named by FILEELM), from a source SRC whose nature is specified by SRCTYPE, Note that the arguments include file and connection handles, in TENEX these correspond to job file numbers (JFNs). 3c4b

PARM: the source is SRC (i.e, an argument of the procedure).

FILE: the source is the current value of an element SRCELM of a file in one of the local process's previously-opened directories (implicitly named by SRCELM). The invoking process must have read access to the source.

The source element is either replaced by EMPTY (i.e, moved) or left unchanged (copied), according to DISP. To move the source element, the invoking process must have write access to the source file.

NETC: the source will be transmitted via a network connection, from socket SOCKET at host HOST, using format FORMAT (same as for LSTDIR).

CHNL: the file element is transmitted via a PCP channel attached to the port identified by the port handle PORH of the local process. (Channels and

ports are discussed in the Process Management Package document.)

Argument/result types: 3c4c

```
fileelm- LIST ( %file handle% INTEGER, <BOOLEAN
           [KEY=TRUE / INDEX=FALSE]> %element% any /
           INTEGER, ...)
disp   - INTEGER [DELETE=0 / RETAIN=1]
srctype- INTEGER [PARM=0 / FILE=1 / NETC=2 / CHNL=3]
   PARM: src- any
   FILE: src- %srcelm% LIST ( %file handle% INTEGER,
              <BOOLEAN [KEY=TRUE / INDEX=FALSE]>
              %element% any / INTEGER, ...)
   NETC: src- %connection handle% INTEGER
   CHNL: src- %porh% INTEGER
```

There are two other procedures in the File Package which might not be necessary in the initial NSW, these are delete element (DELELM) and get file structure type (GETST). 3d

To allow maximum progress on the implementation of the File Package under the TENEX Operating System it is proposed that bulk of the File Package be implemented by BBN but that the routines GETIT and PUTIT be implemented by SRI. 3e

The File Package Appendix
FP-APP Version 2

23-DEC-74

Jon Postel
Augmentation Research Center

Stanford Research Institute
Menlo Park, California 94025

The File Package (FP -- 24582) is a file manipulation tool that operates within the setting provided by the Procedure Call Protocol (PCP -- 24459,), with which the reader of the present document is assumed familiar.

(J24813) 24-DEC-74 17:05;;;; Title: Author(s): Jonathan B.
Postel/JBP; Distribution: /NSW([INFO-ONLY]) ; Sub-Collections:
SRI-ARC NSW; Clerk: JBP; Origin: < POSTEL, NSW-FILE-APP.NLS;4,
>, 24-DEC-74 13:34 JBP ;;;; ####;

DCE 24-DEC-74 17:13 24814

To FDBK, second note on re CONAN compile bug

1) A second strange type of bug -- When I do a Compile Contentanalyzer execution in Progra Subsystem, I often find that Viewspec "i" is automatically imposed upon me during a subsequent action. If it is a purposefully implemented feature, I don't appreciate its value and would like to have it reconsidered; if it is a bug, I'd like it fixed. 1

2) About the previously described bug (GJOURNAL, 24809,) -- I experimnted a little further. Found that the following version will compile all right if the bug select is done near the "9", but if the bug select is more than a few characters in front of the "9", I get the "... string too long" error. Also, if I add one more copy of the "xxx..." visible at the end of the statement, Compile Contentanalyzer gets the error every time. Delete one of the trailing visibles from the statement shown, and the pattern always compiles o.k. The error seems simply related to the length of the statement beyond the pattern. 2

```
SNDMSG by A to B:      9CH ("ENGELBART") [EOL] ["tion:"]
$=(EOL/"FARBER") ("FARBER"); xxxxxxxxxxxxxxph xxxxxxxxxxxxxxph      2a
```

DCE 24-DEC-74 17:13 24814

To FDBK, second note on re CONAN compile bug

(J24814) 24-DEC-74 17:13;;;; Title: Author(s): Douglas C. Engelbart/DCE; Distribution: /FDBK([ACTION]) ; Sub-Collections: SRI-ARC FDBK; Clerk: DCE;

JHB 26-DEC-74 08:29 24815

Location of the Command Marker or Control Marker -- why not 'pointer'?

Re KIRK (24802,)
I support KIRK's comment that the CM should be left at the first character of the statement following the moved statement or structure.
I noticed that Kirk referred to the "command marker". That is not defined in the Help DB nor is it used in the Quick Reference. Control Marker is instead. II prefer the former, but the question arrises, why not call it a Pointer? That would be consistent with users experience, the part of Help devoted to pointing, be more concrete a concept and more easily explained to new users. Could we not substitute Pointer for Control Marker in future documentation? 1

JHB 26-DEC-74 08:29 24815

Location of the Command Marker or Control Marker -- why not 'pointer'?

(J24815) 26-DEC-74 08:29;;;; Title: Author(s): James H. Bair/JHB; Distribution: /KIRK([ACTION]) FEED([ACTION]) SRI-ARC([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JHB;

JHB 26-DEC-74 08:41 24816

Changed Userguides,Locator, ?

I noticed recently that the Locator has been changed, for example, the Tenex references are no longer in NLS form. I also note that NLS Code files are referenced which may explain why users are asking how they can modify the NLS running system at Office-1. I would like to be able to give users up-to-date info on the locator and associated Documentation but have forgotton where or who authored the Journal doc explaining the changes to Locator. Please send me the citation so I can catch up on this important user service? Thanks. (locator updated last by POOH on 15 Nov) 1

JHB 26-DEC-74 08:41 24816

Changed Userguides,Locator, ?

(J24816) 26-DEC-74 08:41;;;; Title: Author(s): James H. Bair/JHB;
Distribution: /POOH([ACTION]) DVN([INFO-ONLY]) JCN([INFO-ONLY
]) RWW([INFO-ONLY]) FEED([INFO-ONLY]) RLL([INFO-ONLY]) ;
Sub-Collections: SRI-ARC; Clerk: JHB;

JHB 26-DEC-74 09:20 24817

Answer to (24780,) -- List of Completed Bug Fixes and Implemented Features

The location of all NLS Feedback was announced in (24636,) 30 Nov 74. Software bugs are listed in (feedback,feed,bugs), a top level branch in the feedback initial file, and those that have been fixed by the Utility software team are moved to (feedback,feed,fixed) by the respective programmer. That list is retained for historical purposes. All other important categories are visable in a top level view of the feedback master file (feed,).
There have not been any feature implementations reported to Feedback recently, but when they are you will find them in (feedback,feed, changes). The file is available for all to peruse , and I hope that the documentation team will find the info necessary to aid in keeping up-to-date, and will feel free to report any changes they are aware of to feedback so that User Development can keep clients informed. Since NLS is "frozen" the frequency of changes may be relatively low, therefore the person operating Feedback should let Kirk know when changes re received so that he may up-date the Help DB.
Kirk should in general feel free to ask me for any assistance needed in his job of maintaining various documentation. 1

JHB 26-DEC-74 09:20 24817

Answer to (24780,) -- List of Completed Bug Fixes and Implemented Features

(J24817) 26-DEC-74 09:20;;;; Title: Author(s): James H. Bair/JHB; Distribution: /KIRK([ACTION]) FEED([INFO-ONLY]) RLL([INFO-ONLY]) JCN([INFO-ONLY]) DCE([INFO-ONLY]) RWW([INFO-ONLY]) NPG([INFO-ONLY]) DIRT([INFO-ONLY]) ; Sub-Collections: SRI-ARC NPG DIRT; Clerk: JHB;

SDC2 26-DEC-74 09:44 24818

Long Range Planning Info Desired

Net mail from site USC-ISIB rcvd at 24-DEC-74 13:14:32 1

Date: 24 DEC 1974 1313-PST
From: HOLG at USC-ISIB
Subject: Long range NSW plan
To: CROCKER at ISIB, BALZER at ISIB, CARLSTROM at ISI,
To: POSTEL at ISI, CARLSON at ISI, LLOYD at ISI,
To: BAGGIANO at ISI, MAYHAN at ISI, CRAIN at ISI,
To: WATSON at SRI-ARC, WARSHALL at SRI-ARC,
To: MILLSTEIN at SRI-ARC, WHITE at SRI-ARC, IRBY at SRI-ARC, To: STONE at OFFICE-1, WINGFIELD at OFFICE-1,
To: RIDDLE at OFFICE-1, WEEKS at OFFICE-1,
To: LAWRENCE at OFFICE-1, UHLIG at OFFICE-1, JACOBS at BBN, To: BURCHFIEL at BBN, THOMAS at BBN, SCHANTZ at BBN,
To: WAAL at SRI-ARC, TRIOLO at SRI-ARC, SCHAFFNER at SRI-ARC cc: holg 1a

Hi folks! 1b

A little Christmas cheer to warm your hearts and wear your fingers. 1c

You are all requested to submit a long-range plan for the NSW. The need for a reasonable plan is immediate, so the fuse is short. Please submit material to me via SNDMSG. I will spend 2 or 3 days synthesizing a fairly objective composite of the contributions and fire it back to you all. I'd like to have responses by 31 Dec 74 -- one week! 1d

I have found it helpful to organize my thinking around the following questions. 1e

1. What are the operational objectives and on what dates? (Eg. Jul 75, Jul 76, Jul 78.) 1f

a) What general capabilities?
b) What tools?
c) What guarantees on the tools?
d) Who will be the users?
e) How will scarce resources be allocated?
f) What performance and/or efficiency should be expected? 1f1

2. What technical developments are required? 1g

a) of the works manager
b) of the Front-End
c) of the protocols
d) of the tools. 1g1

Long Range Planning Info Desired

3. What organizational developments are required? 1h

a) among the sponsors
b) among the contractors
c) within the government
d) in industry. 1h1

Thanks, 1i

Steve Crocker
Steve Crocker/holg 1j

------- 1k

SDC2 26-DEC-74 09:44 24818

Long Range Planning Info Desired

(J24818) 26-DEC-74 09:44;;;; Title: Author(s): Steve D. Crocker/SDC2; Distribution: /NSW([INFO-ONLY]) ; Sub-Collections: NIC NSW; Clerk: JBP;

DCE 26-DEC-74 10:34 24819

Re 24769 and a DOC IDENT: recommend using FDBK instead

Dirk, Anne: Referencing your message of 17 Dec (24769,) -- "An IDENT FOR DOCUMENTATION": -- as I mentioned to you this morning, I think that users would best have but one communication channel for questions, special-service requests, etc. I'd vote then that FDBK/FEEDBACK be used for documentation-related user communications, and that this be reflected in what HELP indicated to the users.
Regards, Doug

1

DCE 26-DEC-74 10:34 24819

Re 24769 and a DOC IDENT: recommend using FDBK instead

(J24819) 26-DEC-74 10:34;;;; Title: Author(s): Douglas C. Engelbart/DCE; Distribution: /POOH([ACTION]) DVN([ACTION]) DIRT([INFO-ONLY]) ; Sub-Collections: SRI-ARC DIRT; Clerk: DCE;

JBP 26-DEC-74 10:47 24820

network journal delivery

Net mail from site MIT-MULTICS rcvd at 26-DEC-74 10:11:46 1

From: Pogran.CompNet at MIT-Multics
Date: 12/26/74 1310-est
Subject: Distribution of File Formats Doc 1a

Jon, 1b

Got an NLS note in my mailbox which informs me of
secondary distribution of Standard File Formats,
NWG/RFC #678. 1c

Is this available in text file form?
xnls is useless.
Reqlly. 1d

Ken 1e

JBP 26-DEC-74 10:47 24820

network journal delivery

(J24820) 26-DEC-74 10:47;;;; Title: Author(s): Jonathan B. Postel/JBP; Distribution: /FEED([ACTION]) FDBK([ACTION]) ; Sub-Collections: SRI-ARC FDBK; Clerk: JBP;

DCE 26-DEC-74 11:48 24821

Reply to Bair (24815,) and Kelley (24802,) regarding Control Marker issues

I like Kirk's suggestion (24802,) about leaving the Control Marker, after a structural-move operation, at the same place it would be left if the moved entity had been deleted. 1

Regarding nomenclature, and alternate terms to refer to what is now called the Control Marker (Bair -- 24815,): I prefer "Control" to "Command" -- not with any great rationale behind the preference, but mostly because since about 1967 I've been using the term "Control" in the sense of a Control Language, Control Metatlanguage, etc, and also in this particular usage the function of the Control Marker is to mark the place where NLS's execution control is addressing (just as the term "control" is used in programming jargon). 2

And when we set up the term in the first place, we assumed that users would be familiar with "markers" as used to point to specific text locations; while "pointers" were things they wouldn't hear about until they started doing L10 programming where "Text pointers" are things very explicitly declared and used. 3

I remember explicitly requesting a terminology change from something else (that I can't remember now) to "Control Marker" some years ago. Doesn't meant that we can't reconsider, though. 4

DCE 26-DEC-74 11:48 24821

Reply to Bair (24815,) and Kelley (24802,) regarding Control Marker issues

(J24821) 26-DEC-74 11:48;;;; Title: Author(s): Douglas C. Engelbart/DCE; Distribution: /FEED([ACTION]) SRI-ARC([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: DCE;

RLL 26-DEC-74 11:50 24822

SUG: Automatic refuse link while printing in TNLS.

Clearly for the future.

RLL 26-DEC-74 11:50 24822

SUG: Automatic refuse link while printing in TNLS.

How about an automatic refuse links whenever someone is in TNLS executing an output terminal or in the print command? Possibly other commands. The system would UNrefuse automatically after printing has stopped. (i.e., accept links). 1

This would allow a clean copy of whatever is being printed without worrying about a link or forgetting about refusing and unrefusing links. 2

I would think this would be a very user oriented feature that would save on many users frustrations. 3

RLL 26-DEC-74 11:50 24822

SUG: Automatic refuse link while printing in TNLS.

(J24822) 26-DEC-74 11:50;;;; Title: Author(s): Robert N. Lieberman/RLL; Distribution: /FEED([ACTION]) NP([ACTION]) NDM([INFO-ONLY]) DVN([INFO-ONLY]) JHB([INFO-ONLY]) ; Keywords: suggestion links refuse; Sub-Collections: SRI-ARC NP; Clerk: RLL;

RLL 26-DEC-74 11:58 24823

user progs: re (31543,) by NDM on (24785,)

Rereading Dean's journal item (31543,) on user progs, I found that the syntax for insert directives is somewhat misleading. Am I correct that the prompts and additional input from the user is in the case when the SRI-format is selected and not in general? If so, this command is in fact a general one. I would hope it would be. Once the format is selected (using format number #) then any additional input from the user would be prompted for per usual. Rob 1

RLL 26-DEC-74 11:58 24823

user progs: re (31543,) by NDM on (24785,)

(J24823) 26-DEC-74 11:58;;;; Title: Author(s): Robert N. Lieberman/RLL; Distribution: /JHB([ACTION]) JCN([ACTION]) NDM([INFO-ONLY]) KIRK([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: RLL;