WRF 29-NOV-71 11:14 8128

ferguson's gargabeWGARBAGE

this is not very easy!

WRF 29-NOV-71 11:14 8128

ferguson's gargabeWGARBAGE

nls is groovy fun! 1

ferguson's gargabeWGARBAGE

(J8128) 29-NOV-71 11:14; Title: Author(s): William R Ferguson/WRF; Distribution: William R Ferguson/WRF; Sub-Collections: SRI-ARC; Clerk: WRF;

WSD 30-NOV-71 9:42 8137

New Task--Flexible Document

REQUIREMENTS: 1

A system to aid in the manipulation, protection, and coordination of documents in the formative state. 1a

To be able to enter a document into the DSS environment which is not fixed, as a Journal entry must be. 1b

To subsequently provide support for a team of persons using that document. Things like 1c

Read-only vs. write access 1c1

Keeping track of changes 1c2

Jump to last change, ad jump to last change by XXX 1c2a

Who chnged what last 1c2b

Notifying members of group that a chnge has been made and by whom 1c2c

Notification of conflicting chnges 1c2d

e.g. if A tries to change something that B has chnged, the system should notify B of A's change (if it has not already done so) 1c2d1

Distribution of updated copies, etc. 1c3

Easy entryof flexible documents into JOurnal (Without losing flexible document capability) 1c4

Temporary locking of files 1c5

One guy is working on a change, an says....I want to lock all of these files, or at least I want to know about any canges made to them. 1c5a

BEGIN-DATE: 2

Not Known 2a

END-DATE: 3

? 3a

PUSHER: 4

DSS 4a

WORKERS: 5

?? 5a

Taskname: 6

flexdoc 6a

Design: 7

8

9

<JOURNAL>8137.NLS;1, 30-NOV-71 9:43 WSD ; Title: Author(s): William S. Duvall/WSD; Distribution: Bruce L. Parsley/BLP; Sub-Collections: SRI-ARC; Clerk: WSD;

Planning Estimate for WSD

(J8138) 30-NOV-71 10:53; Title: Author(s): William S. Duvall/WSD; Distribution: Bruce L. Parsley/BLP; Sub-Collections: SRI-ARC; Clerk: WSD;

WSD 30-NOV-71 10:53 8138

Planning Estimate for WSD

This represents what I think I should do in the minimum possible case. I will obviously not get all of it done, and the stuff that will suffer will be the non-concrete tasks, e.g. sky-bluing, etc.

WSD 30-NOV-71 10:53 8138

Planning Estimate for WSD

Estimated activity--December 1

- Imlac 1a
  - 4 days 1a1
- Flexible document system 1b
  - 5 days 1b1
- Seas stuff 1c
  - DD-NLS 1c1
  - 5 days 1c2
  - String Bugging 1c3
    - .5 day 1c3a
- Base Planning 1d
  - 1 day 1d1
- Personal Planning 1e
  - 1 day 1e1
- Correspondance, etc. 1f
  - 1 day 1f1
- Bull Sessions 1g
  - 4 days (min) 1g1
- DSS Coordination 1h
  - 4 days 1h1
- Journal Babysitting 1i
  - 2 day 1i1
- sky-blue time 1j
  - 3 days 1j1

Re-orientation time 1k

2 days 1k1

Kibitzing on various activities in ARC, finding bugs, suggesting new features, etc. 1l

3 days. 1l1

Augmentation Research Center
Stanford Research Institute
Menlo Park, California 94025

To:

Augmentation Research Handbook
Stanford Research Institute
333 Ravenswood Ave.
Menlo Park, California 94025

8162

(Secondary Distribution Copy)

A suggested parser for the primitive source level debugger. 1

NOTE: 1a

See also (Journal, 7667, 1), (Journal, 8161, 1), (deutsch, debugging,), (deutsch, lmd,). 1a1

In the following, NUM = typed in number with appropriate feedback, BUG = bug selection (DNLS or TNLS) / SP LIT, LIT = literal string of characters with appropriate feedback, SP = space character with feedback, text enclosed in angle brackets is user feedback. 1a2

It is assumed that the DDT symbol table manipulation primitives are to be used to convert values to symbols, and vise versa. 1a3

Examining globals, fields, frame data, strings: 1b

'E <Examine> 1b1

'G <Global> BUG ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]/ 1b1a

COMMENT: Mode switch is always permentent. The value of the global may be modified by typing in a new value. 1b1a1

'F <Field> BUG ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]/ 1b1b

COMMENT: The value of the field may be modified by typing in a new value. 1b1b1

'S <String> BUG ['L <Length>] ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]/ 1b1c

COMMENT: Either the length or the actural character string can be displayed and changed. 1b1c1

'R <Return address> ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]/ 1b1d

'L <Local> [NUM] ['+/'-/SP NUM] ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]/ 1b1e

COMMENT: If first NUM is omitted, it is assumed to be 1. Locals include the parameters to the procedure. 1b1e1

'A <All locals> ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA/ 1b1f

COMMENT: values can not be changed via this command. 1b1f1

'N <Next> ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]/ 1b1g

COMMENT: The next Global, Local, or Field (depending on which has been most recently accessed) will be displayed and can be modified. 1b1g1

'P <Previous> ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]/ 1b1h

COMMENT: The previous Global, Local, or Field (depending on which has been most recently accessed) will be displayed and can be modified. 1b1h1

'. <.> ['+ <+>/ '- <->/ SP NUM] ['M <Mode> ('S <Symbolic>/ 'N <Numeric>)] CA [SP LIT CA]; 1b1i

COMMENT: SP is equivelent to '+, as in DDT. 1b1i1

Moving the Record Pointer, Frame Pointer: 1c

'M <Move> 1c1

'R <Record Pointer to> BUG CA/ 1c1a

'F <Frame Pointer> (prints name of frame's procedure) 1c1b

'D <Down> [NUM] ['P <Procedure> BUG [NUM]] CA/ 1c1b1

COMMENT: First NUM is how many frames to move down the stack. Second NUM is the number minus one of frames for the specified procedure to skip. 1c1b1a

'U <Up> [NUM] ['P <Procedure> BUG [NUM]] CA/ 1c1b2

COMMENT: First NUM is how many frames to move up the stack. Second NUM is the number minus one of frames for the specified procedure to skip. 1c1b2a

'T <Top> CA/ 1c1b3

COMMENT: Moves the Frame Pointer to the frame below the debugger frame if control came from a break point. 1c1b3a

'B <Bottom> CA; 1c1b4

Back trace of procedure calls: 1d

'T <Type Calls> 1d1

'U <Up> [NUM] CA/ 1d1a

COMMENT: types the procedures which have been called, from the frame pointed to by the Frame Pointer to the top of the stack, excluding the debugger frame (on the top of the stack). 1d1a1

'D <Down> [NUM] CA; 1d1b

COMMENT: types the procedures which have been called, from the bottom of the stack to the frame pointed to by the Frame Pointer. 1d1b1

Mode setting: 1e

'S <Set Mode> ('S <Symbolic>/ 'N <Numeric>); 1e1

Break pointing: 1f

'B <Break> [NUM] BUG ['C <Call> BUG] CA; 1f1

COMMENT: The break location is examined to see if it is manipulating the stack pointer S. If it is, the break location is incremented by one. The instruction in the break location is saved in an INSTRUCTION array and a simple call to the appropriate break point routine (e.g. bp1, bp2, ... bpn) replaces that instruction in the code. If the Call BUG option is preseant, the specified procedure address is stored in a PROCEDURE array. When control passes through the break location, the call to the appropriate break point routine is executed. If the PROCEDURE array has a non-zero entry for this breakpoint, the specified procedure is called. If the procedure returns TRUE, the break will take place, otherwise it wont. Note that this allows one to insert procedure calls at the beginning of procedures. A count of the highest breakpoint set so far, BREAKCOUNT, is kept. If the NUM option is ommitted it is assumed to be BREAKCOUNT + 1. 1f1a

'U <Unbreak> (NUM/ 'A <All>) CA; 1f2

COMMENT: Removes specified breakpoint(s). 1f2a

'P <Proceed> CA; 1f3

COMMENT: Proceeds from last breakpoint. 1f3a

Passing control to the program: 1g

'G <Goto> BUG ['U <Unwind Stack>] CA; 1g1

**COMMENT: The stack can be unwound to the frame pointed to by the Frame Pointer before control is passed to the program. 1g1a

Replacing procedures -- Incrementally changing the system: 1h

'R <Replace Procedure at> BUG <by> BUG CA; 1h1

COMMENT: A branch to the specified routine replaces the instruction at the specified location (this may overwrite the instruction manipulating the stack pointer S at the beginning of procedures). The replacing procedure should, of course, expect its arguments in the same order as the replaced procedure. The replacing procedure, however, can disregard the arguments and can have different local variables. 1h1a

Immediate procedure calls: 1i

'C <Call> BUG CA; 1i1

COMMENT: The selected text is parsed to be a procedure call with at most simple variables or constants as arguments. The call is then executed. Note that this can result in another breakpoint, in which case there will be multiple occurrances of the debugger on the stack. This should cause no difficulties. 1i1a

Status of Frmae Pointer, Record Pointer, and breakpoints: 1j

'L <Location of> 1j1

'F <Frame Pointer> CA/ 1j1a

COMMENT: Types the procedure name (in symbolic mode) for the current frame (the address field of the cell pointed to by the return address in the frame). 1j1a1

'R <Record Pointer> CA/ 1j1b

COMMENT: The contents of the Record Pointer is typed in symbolic mode. 1j1b1

'B <Breakpoints> CA; 1j1c

COMMENT: Types the symbolic location of all breakpoints which are set. 1j1c1

Terminating debugger subsystem: 1k

'Q <Quit> CA; 1k1

CHI 30-NOV-71 14:33 8162

A Parser For the Primitive Souce Level Debugger

(J8162) 30-NOV-71 14:33; Title: Author(s): Charles H. Irby/CHI;
Distribution: Walter L. Bass, William S. Duvall, Mary S. Church, J. D.
Hopper, Charles H. Irby, Harvey G. Lehtman, John T. Melvin, Bruce L.
Parsley, William H. Paxton, L. Peter Deutsch, James G. Mitchell, John T.
Melvin, J. D. Hopper, Don I. Andrews/NPG LPD JGM JTM JDH DIA;
Sub-Collections: SRI-ARC NPG; Clerk: CHI;
Origin: <IRBY>DEBUGGER.NLS;8, 30-NOV-71 14:26 CHI ;

Secondary Distribution Copy

CBL 30-NOV-71 15:29 8163

Sample Document for Principle Investigators

this is the text of the document. 1

this is more of the paper. 2

this is the last of the stuff. 3

Sample Document for Principle Investigators

(J8163) 30-NOV-71 15:29; Title: Author(s): Charlotte B. LaRoche/CBL; Distribution: Thomas R. Dines, Lawrence G. Roberts, Daniel G. Bobrow, Frank E. Heart, Edward L. Glaser, Thomas M. Marill, Alan J. Perlis, T. E. Cheatham, Duane L. Stone, James W. Forgie, Alfred H. Vorhaus, Thomas N. Pyke, Keith W. Uncapher, Clark Weissman, John McCarthy, Edward A. Feigenbaum, Leonard Kleinrock, David O. Harris, William K. Pratt, David C. Evans, Douglas C. Engelbart, Bertram Raphael, Daniel L. Slotnik/PI; Sub-Collections: NIC PI; Clerk: CBL;

EW 30-NOV-71 18:28 8164

plans to stay at sri

Alex,

SRI infact held class for only the two days. However, i decided it would be constructive to stay here another day, work using the NLS system, and ask questions while we are here. Julie said that she would like to do so too.

I may then spend extra time site seeing, but do not know what Julie plans to do.

EW 1

EW 30-NOV-71 18:28 8164

plans to stay at sri

(J8164) 30-NOV-71 18:28; Title: Author(s): Ellen Westheimer/EW; Distribution: Alex A. McKenzie/AAM; Sub-Collections: NIC; Clerk: EW;

JBL 2-DEC-71 9:22 8165

Questions about nls and EXEC

I am curious (pink)

JBL 2-DEC-71 9:22 8165

Questions about nls and EXEC

John, 1

A question about the journal system: 1a

The journal users guide we have only talks about submitting messages and documents, not about retrieving them. Specifically, it mentions the option of specifying "keywords", which, presumably, are
a means of finding entries by subject. 1a1

Is there in fact a way of locating an entry by keyword? If so, is it documented? 1a2

About your EXEC: 1b

It really will be quite useful for us here if we have a way of examining the status of our connection as TENEX sees it. 1b1

Are there any plans for implementation of a means whereby a network user can get a NETSTAT, either in EXEC or by means of a separate socket? 1b2

Thanks for your help. 1c

This is the first message I have ever sent. 1d

Pax
Joel 1e

Questions about nls and EXEC

(J8165) 2-DEC-71 9:22; Title: Author(s): Joel B. Levin/JBL; Distribution: John T. Melvin, Joel B. Levin/JTM JBL; Keywords: keyword journal retrieval NETSTATUS; Sub-Collections: NIC; Clerk: JBL;

DLS 2-DEC-71 11:43 8166

RADC/SRI PROCUREMENT STATUS

I have 4 procurements in the mill here at RADC, which you will be asked to respond to during the next 1-2 months. They are: 1

An extension to the existing contract AF30602-70-C-0219 for three months for approx. 330K. This is necessary because: 1a

procurement will not write a twenty four month extension to the existing contract 1a1

the order from ARPA was not received soon enough to assure continuity of work with the time delays necessary to write a new contract. 1a2

The three month extension will hopefully reduce the next contract below 2.5 million which means that it will not have to be reviewed by HqAFSC (which could take another 3-4 weeks). 1a3

An engineering change to AF30602-70-C-0219 for approx. 75K to buy us terminals. 1b

A separate effort from RADC to obtain support for the next 6-8 months for our planned use of NLS within the section/branch. (workstatement) 1c

A 21 month continuation of ARPA support to start 9May72 and go to 9Feb74. 1d

Procurements 1&2 above are in the final stages before they will be released to you for bid---this means that you could receive them in 1-6 weeks (sorry I can't be more precise). I am scheduled to have a procurement meeting on the 3rd one on the 6th of Dec.----which means that if all goes well you should be receiving that one a couple of weeks after the first two. The 4th one is in typing in my section office now---I would guess that one should get to you by the first of Feb. 1e

You can go ahead and order the four Execuports and their cassette recorders now if you like. Have them delivered directly to RADC. You might consider ordering a cassette recorder for yourself at the same time. The Execuport marketing man said that he would deliver them within a week or two after he received a purchase order. When you have a purchase order number call and let me know so I can alert him. If you are still going to lease the Novar machine then try to lease one for us for a year. We have money to cover it if the total comes to over 75K. 1f

I have another round of breifings to give to the RADC commander and to Air Staff on our AHI project---Do you have any short statement of the general/basic prblems in society that your system addresses----also what makes your system different/superior to others which have text editors, message, journalling, programming, etc. capabilities?? 1g

What's the status of the L-10 user's guide? 1h

How does the bathroom look now?????? 1i

RADC/SRI PROCUREMENT STATUS

(J8166) 2-DEC-71 11:43; Title: Author(s): Duane L. Stone/DLS; Distribution: James C. Norton/JCN; Sub-Collections: NIC; Clerk: DLS; Origin: <STONE>DLS.NLS;8, 2-DEC-71 11:39 DLS ;

LPD 2-DEC-71 13:47 8167

Experience with the Network 11/71

This is a report on my experiences with the ARPA network for the month of November 1971. I was trying to accomplish a specific task, namely the development of a large LISP program at BBN using the TIP at Nasa Ames. 1

The table below is a copy of a log I kept of every attempt to work through the network. These data relate only to my success in pursuing this goal, and should not be taken as any more general measure of network availability or reliability. 1a

-- 1b

All times in the table are Pacific time. Add 3 hours for time at BBN. A * next to an entry indicates that it represents a continuation of the previous one, not a new attempt. 1c

Date/Time Terminal TIP BBN IMP BBN TENEX BBN LISP 1c1

1 2200 - - - - - 1c2

Success. Did not try to work, since there had been a disk crash the preceding weekend and I was waiting for listings to arrive by airmail so I could tell how much work I had lost. 1c2a

2 1200 - - - ? 1c3

TIP accepted @L, but no further response (no T R OPEN, CAN'T, TIMEOUT, or REFUSED within 2 minutes). Long distance dialup (henceforth LD) produced half the login herald and an apparent crash. Doing NETSTATUS at the SRI-ARC TENEX (henceforth NETSTAT) shows BBN "up". 1c3a

3 1200 - - - - - 1c4

Success. Machine load was too heavy for me to accomplish anything, however, so I didn't try for long. 1c4a

3 1800 - - ? ? 1c5

TIP said DEAD REFUSED in response to @L. ARC down so couldn't do NETSTAT. 1c5a

4 2100 - - - - - 1c6

Success. 1c6a

5 No attempts. 1c7

6 2100 - - ? ? 1c8

DEAD REFUSED. ARC down. 1c8a

7 0930 - - - - - 1c9

Success. Testing only, did not intend to work at this time. 1c9a

7 1130 - - - ? 1c10

Successful connection, but login mssage contained a request from Ray Tomlinson (BBN) to log out so he could crash TENEX. 1c10a

7 1230 - ? 1c11

TIP did not respond to initial E. 1c11a

7 1630 - ? ? ? 1c12

TIP gave TIMEOUT in response to @L. TIP then went into an anomalous state: @C produced T R CAN'T, @H n produced T CAN'T; no response to E on other dialup lines. 1c12a

7 2000 / BBN TENEX down 1c13

-to- / for scheduled 1c14

8 0400 / maintenance 1c15

9 ---- - - - - - 1c16

Success. 1c16a

10 No attempts. 1c17

11 ---- - - - - 1c18

Successful connection using TELNET at ARC. Testing only since ARC strongly discourages TELNET use for non-ARC related projects. 1c18a

12 ---- - - -
- 1c19

Successful TELNET connection at ARC. See comment above. 1c19a

13 No attempts. 1c20

14 1200 - - ? ?
- 1c21

Successful connection; however, went DEAD spontaneously at 1300 (before I had gotten much of anything done). Still DEAD REFUSED at 1445. ARC down. LD call produced infuriating results shown on enclosed sheet. 1c21a

14 2000 / BBN TENEX down 1c22

-to- / for scheduled 1c23

15 0400 / maintenance 1c24

15 1100 - ? ? ? 1c25

No response to @L for > 2 minutes, no TIMEOUT; LD call shows 2 network users, NETSTAT at BBN says "up", NETSER running. 1c25a

15 1200 - ? ? ?
- 1c26

Successful connection, preceded by 5 unsuccessful tries because TIP did not answer E. Also occasional waits of > 20 seconds for full-duplex echo (generated by TENEX), apparently because of scheduling anomaly. Connection "died" spontaneously after about 3/4 hour. LD voice call -> all ok there. Was able to redial, but lost all work (including an initialization procedure that consumes at least 5 minutes of CPU time). 1c26a

15* 1600 - ? ? ?
- 1c27

Connection died, TIP said DEAD. No work other than initialization lost. @L produced no response in 2 minutes. 1c27a

15 1700 - - - -
- 1c28

Success. 1c28a

16 0830 - - - -
- 1c29

Success. 1c29a

16* 1000 - ? ? ?
- 1c30

Connection died; no DEAD message, just no response for > 2 minutes. @C -> T CLOSED R CLOSED; @L -> TIMEOUT. 1c30a

16 2200 - - - -
? 1c31

Successful connection. LISP subsequently hung up in the garbage collector, a state from which it cannot be rescued. Lost a few edits. 1c31a

17 ----) TIP down for 1c32

18 ----) scheduled 1c33

19 ----) maintenance 1c34

20 ----) BBN down for scheduled 1c35

21 ----) disk and drum maintenance 1c36

22 1820 - ? ? ?
- 1c37

Success, but connection went DEAD at 1900 while I was saving my work. Luckily TENEX sometimes does the right thing (detach the job and let it continue to run), so I actually lost no work. 1c37a

23 2200 - - - -
- 1c38

Success. 1c38a

24 0000 ? ? - -
- 1c39

Success. However, the phone connection broke spontaneously 3 times so I gave up. When this happens one must dial the TIP back up, re-login, and then ATTACH back to one's old job, assuming it is still there. 1c39a

```
 24  1000     ?        ?        -          -
-                                                      1c40
```

Successful connection, but could not maintain phone connection for more than 10 minutes on numerous attempts. Then TIP apparently crashed: no response to E on any line. ARC IMP down for maintenance. 1c40a

```
 24  1030     -                 -          -
?                                                      1c41
```

LD call. LISP hung up in garbage collector again, lost 1/2 hour of work. 1c41a

```
 25  1100     ?        ?        -          -
-                                                      1c42
```

Successful connection. Lost phone connection after 1/2 hour and quit. At this time we investigated the Texas Instruments 700-series terminal I was using and found that it had problems picking up and maintaining the phone connection when connected to things other than the TIP. 1c42a

```
 26  ----     -        ?                               1c43
```

No response to E. TELNET from ARC gave message OPENF REFUSED. 1c43a

```
 27  1100     -        ?                               1c44
```

No response to E. NETSTAT at BBN via LD said NO HOSTS ARE UP. 1c44a

```
 28  ----     -        ?                               1c45
```

No response to E. 1c45a

```
 29  1800     -        -        ?          ?
-                                                      1c46
```

@L gave DEAD REFUSED. 1c46a

29 1930 - - - - 1c47

Success. Discovered that BBN had had a disk crash and backed up the disk to 1800 on 11/24. Sent a message to the operator requesting listings of all my files (via airmail) and quit. 1c47a

30 1900 - ? ? ? - 1c48

Successful connection, but went DEAD after 1/2 hour. 1c48a

As a result of these experiences I have some strong opinions about the usability of the network for terminal users (TELNET and TIP). I have cast these in the form of specific difficulties encountered and proposed cures. 2

The TIP does not accommodate T-I terminals at full speed (30 characters per second) since it does not provide enough padding characters after a carriage return. 2a

This is clearly the TIP's responsibility since it does provide sufficient padding for Execuports. The parties responsible for TIP software should provide the correct padding algorithm for this terminal and the new Anderson-Jacobson terminal. The algorithm can probably be obtained from the manufacturer in both cases. 2a1

The TIP is no help in determining why a given host cannot give service. and NETSTAT is not much better since it does not distinguish between: IMP up, host down; host up but not running NETSER; host up and running NETSER. 2b

The TIP needs a command to query a given IMP without logging in on the host, and to query the host witout logging in. NETSTAT needs to distinguish between the three cases, either by attempting to log in on the host or by some new mechanism which I can't specify since I don't know enough about the mechanics of the network. 2b1

The BBN disk is very unreliable. My impression is that they have a substantial crash once every month or two. 2c

Replace it with disk packs. The SRI AI installation has already done so with their Bryant disk, and SRI ARC is about to. Almost no research installation that I am acquainted with is happy with this disk. While the Bryant file was a bargain five or more years ago, disk packs are now a better buy, as well as being more reliable and partitionable in case a drive fails. 2c1

The TIP and IMP do not appear to be nearly as reliable as one would expect from what is supposed to be a commercial machine with many years' experience behind it. 2d

I have no idea why this is and therefore have no suggestions about remedying it. I suspect that the problem is not in the 516 processor and memory but in the various interfaces to modems, hosts, etc. In particular, SRI ARC has had a lot of trouble with their interface. 2d1

To summarize, I feel the service I have gotten from the network is probably inferior to the service I could have obtained locally (time between midnight and 5 AM on a local TENEX). While I do not feel my attempts to work at BBN through the network have been a waste of time, I feel that the network cannot reasonably be considered a service facility, as opposed to an xperimental instrument, at this time. 3

Experience with the Network 11/71

(J8167) 2-DEC-71 13:47; Title: Author(s): L. Peter Deutsch/LPD;
Distribution: Steve D. Crocker/SDC2; Sub-Collections: NIC; Clerk: LPD;
Origin: <DEUTSCH>NETLOG.NLS;5, 2-DEC-71 13:31 LPD ;

INTRODUCTION 1

The following is a suggested scenerio for the Baseline Record System (BRS) for the next several months. BLP and CHI collaborated in producing this document. 1a

The organization of this document is as follows: 1b

The next four main sections each describe a "stage" or "version" of the BRS (the current stage is hereby declared to be Stage VIII). 1b1

Each subsection of these main sections will talk about a "part" of the BRS. They will only describe changes or increments from the previous Stage. The BRS is hereby declared to have the following parts: 1b2

file or files containing Baseline data and what items of information are kept in which files 1b2a

conventions for storing information in those files 1b2b

different possible types of views of Baseline data 1b2c

procedures for getting new entries into the BRS 1b2d

procedures for updating information on entries already in the BRS 1b2e

what views of the Baseline data get periodically Journalized, i.e., of what does the Baseline Record consist 1b2f

NLS commands in the Baseline subsystem 1b2g

set of programs required for all the functions of the BRS 1b2h

observations 1b2i

The last main section consists of a bunch of features/facilities to be added to the BRS in the future. 1b3

The design criteria used in this document are: 1c out-of-date or inaccurate information is probably worse than useless 1c1

a system that gives an easily assimalated view of what is currently going on would be of value 1c2

a system that gives a picture of at least the directions in which the group is moving would be of more value 1c3

a system that allowed a person to choose (consider different) goals and then play with different strategies for reaching those goals would be of great value 1c4

probably the most important thing to consider in devising a USEFUL BRS is the balance between the amount and detail of information that people are willing to keep up-to-date and the amount and detail necessary to provide the information desired by the users of the BRS. 1c5

As regards the question of the balance of the amount and detail of information, there are several things that can effect that balance. 1d

one or more people who spend a lot of glomming the information would take the burden off the rest of the people 1d1

easily used tools or aids to entering new and update information would make the burden lighter 1d2

if users of the information made their use visible to the suppliers of the information, the suppliers motivation would be raised 1d3

if people were also users of the information they supplied, they would be highly motivated to keep it up-to-date 1d4

STAGE VIII (present system, see (8112,)) 2

Basedata 2a

There are two files: (MSR, Basedata,) and (MSR, Donetasks). In each the basic organization is by "task". Current "tasks", needs, possibilities, and bugs are kept in Basedata. Tasks that have been completed are kept in Donetasks. 2a1

There are the following (possible) items if information about each task: 2a2

task name 2a2a

begin-date 2a2b

end-date 2a2c

pusher 2a2d

other workers 2a2e

tags 2a2f

"requirements" (actually almost always just a task description) 2a2g

designs 2a2h

buyers 2a2i

subcontracts 2a2j

man-time estimates 2a2k

subtasks 2a2l

Basedata Storage Conventions 2b

NLS levels are used as some of item delimiters. The first 6 items are stored in level 2 statements as text strings. The rest are stored as level 4 branches with a level 3 statement telling which item is stored below. 2b1

Views 2c

Which tasks are to be seen can be selected on the basis of pusher, workers, tags, begin- and end-dates, and whether they are Needs and Possibilities. 2c1

There is quite a range of formatting options, including the conversion of numerical dates to calendars. 2c2

New Task Procedures 2d

New tasks are entered by filling out a form and getting it to the Baseline maintainer. 2d1

Updating Procedures 2e

Each person gets a weekly printout of the tasks on which he is working, marks it up, returns it to the Baseline Maintainer, who checks it and gives it to PSST to enter into Basedata. 2e1

Baseline Record 2f

Mostly nonexistent. Two views of Basedata, one containing all the tasks and one containing a separate page for each person is supposed to be Journalized each week. 2f1

Commands 2g

Parameter Set 2g1

Institute Sequence generator and/or Content analyzer program(s) 2g2

Status of parameters 2g3

Programs 2h

A set of programs that does task selection and formatting to produce the various views. 2h1

A program that checks the format and completeness of Basedata. 2h2

Some sort key programs. 2h3

Observations 2i

The information in Basedata is only sporadically anywhere near up-to-date. The only information that is ever up-to-date is the break down into tasks, their begin- and end-dates, the pushers and workers, and the task descriptions. The other items are mostly not present. 2i1

The amount of information that is in there is mostly the result of glomming by BLP, although its gotten a lot better with several people spending some effort each week in updating and submitting new "tasks". 2i2

The information is mostly not visibly being used. The section on NLS bugs has gotten some visible use. Also a couple of people have stated that they derived some benefit from seeing a list of the tasks on which they were supposed to be working each week. 2i3

Sporadically one could get a non-easily-assimilated view of what was going on in the software people of ARC and in NIC. 2i4

STAGE IX

3

The only change here is the addition of the command New Task. 3a

The New Task command will do the following things: 3b

First it checks if the file <MSR>Newtasks.PC exists (not deleted). If it exists, then someone else is trying to enter a new task (or maybe something is screwed up), so the user is told to try later. 3b1

The partial copy is created. 3b2

The user is asked for each item of information. "Requirements" and submitter are necessary. 3b3

The garnered information is appended onto the file in some appropriate format. 3b4

The file is updated. 3b5

Once every day or two, the Baseline maintainer will look at the file and edit it appropriately. It is then printed and given to PSST to edit into Basedata. The file is deleted and expunged. 3c

Note that this is a simple Journal. There must be some ramifications there. 3d

STAGE X

4

Basedata 4a

There will be one file only: (MSR, Basedata) -- Donetasks will be deleted as not being used or usable. 4a1

Basedata will contain many "entries", of which there are several types: 4a2

buyers 4a2a

The 0th level buyer ARC plus the 1st level buyers. 4a2a1

finite tasks 4a2b

Work that will be finished in the forseeable future -- most of the "tasks" in Stage VIII are of this type. 4a2b1

ongoing tasks 4a2c

Describable, delimitable work that is of an ongoing nature, e.g., maintaining Basedata. Most of the tasks with --- as begin- and end-dates in Stage VIII are of this type. 4a2c1

persons 4a2d

One entry for each person at ARC. It will contain only two items of information: hours worked per week, percentage of overhead time. 4a2d1

bugs 4a2e

Descriptions of bugs and glitches. 4a2e1

needs 4a2f

Descriptions of needs by buyers for which they have not as yet found a contractor. 4a2f1

possibilities 4a2g

Descriptions of ideas (potential tasks) for which no buyer has yet been found. 4a2g1

There are the the following different possible items of information about an entry: 4a3

type of entry 4a3a

It tells which of the types of entry this entry is. 4a3a1

entry name 4a3b

Every entry has an entry name which is merely a short handle on it. 4a3b1

entry description 4a3c

A short description of what the entry is all about -- entry names often don't tell enough to all users. 4a3c1

pusher 4a3d

A pusher is the person primarily responsible for work on that entry. Pushers are mostly the only ones who can get information about that entry changed. 4a3d1

other workers 4a3e

Finite and ongoing tasks may have people other than the pusher who are contributing to the task. Pushers are workers. 4a3e1

percent time 4a3f

For each worker, the percentage of his time that is being spent working on this entry. 4a3f1

time estimate 4a3g

Every finite task will have an estimate (by the pusher of the task) of the number of 8 hour, crash free, interruption free, fairly good response, working on nothing else days it would take to finish (do whatever remains of) the task. 4a3g1

subentries and priorities 4a3h

Subentries are pointers to all entries which this entry is buying, or is dependent upon, or cares about, or etc. Every subentry item must have a priority associated with it. 4a3h1

A priority is an integer from 0 to 10 (10 being the highest). 4a3h2

In the early stages priorities will be used primarily as ordering devices for plexes of subentries. 4a3h3

order 4a3i

An integer from 0 to 100 assigned by the pusher. Zero means the work on the entry is in progress. The number is used only as an ordering device when viewing all the tasks on which a person(s) is working. 4a3i1

submitter 4a3j

The person who discovered the bug or thought of the possibility; or the entry name of the entry for which this is a need. 4a3j1

hours worked per week 4a3k

This is only present in person entries. It is the average number of hours worked per week. 4a3k1

More about time things: 4a4

Each person is considered to work x hours per week, 100% of that time distributed between: overhead, ongoing tasks in progress, finite tasks in progress, and buyer entries. The x hours and overhead percent are kept in person entries, The other percent times are kept in the indicated types of entries. 4a4a

The dividing line between what is "overhead" and what gets assigned to ongoing tasks is vague and therefore flexible. As the system gets more sophisticated, we have the choice of elaborating the overhead portion of each person's entry or adding more ongoing tasks. 4a4b

There is one more number that is needed: a system wide fudge factor reflecting the average response time, system availability and reliabilty, current vibes, whatever. This number would be kept as a user settable parameter. This number (a multiplcative factor) should probably only be applied to time left estimates of finite tasks, and percent times of ongoing tasks that are dependent on the computer. 4a4c

Now with all the above numbers plus the time left estimates for finite tasks, a bullshit estimate of the end-date for current finite tasks (order = 0) can be calculated. Also, by using the priority numbers (used only as an ordering device) and assuming that each worker can be used interchangebly with the other workers on the the same task, the end-dates for all the finite tasks could be calculated. 4a4d

There is probably one other set of numbers needed: a fudge factor to be applied to the time left estimates of each person. Eventually this could be arrived at by empirical means (by keeping track of all estimates and keeping track of actual time spent). 4a4e

Buyer entries: 4a5

A buyer entry must have the following items of information: 4a5a

type, entry name, entry description, pusher, percent time 4a5a1

It may have the following items: 4a5b

other workers, subentries & priorities 4a5b1

The buyers are the following. 4a5c

ARC 4a5c1

each of the columns in ARC's current pursuit chart 4a5c2

each of the rows in ARC's current pursuit chart 4a5c3

long range research (Internal Research and Development) 4a5c4

each coordinator role 4a5c5

Possibilities 4a5c6

This is the buyer (collection point) of ideas (potential tasks) for which no buyer has been found. 4a5c6a

The pusher is the manager or coordinator of that area of ARC. The percent time is the percent spent "managing". 4a5d

The buyer ARC will have the other buyers as its only subentries. 4a5e

Each of the column and row buyers will have only the boxes in its column or row as its subentries. Thus each box is an entry (probably of ongoing task type). 4a5f

Finite tasks: 4a6

A finite task entry must have the following items of information: 4a6a

type, entry name, entry description, pusher, time estimate, order 4a6a1

It may have the following items: 4a6b

workers, percent time, subentries & priorities 4a6b1

The time left estimate is made by the pusher and is the total of "idealistic man-days" required by all of the workers to finish whatever is left of the task (which may be all of it if it hasn't started yet). 4a6c

If work on the task has begun (order = 0), then the percent time information must be there for each worker -- note that these numbers only get made when work is actually in progress. 4a6d

Ongoing tasks: 4a7

An ongoing task entry must have the following items of information: 4a7a

type, entry name, entry description, pusher 4a7a1

It may have the following items: 4a7b

workers, subentries & priorities, percent time, order 4a7b1

If work on the task has begun (order = 0 or order not present), then the percent time information must be there for each worker. 4a7c

Persons: 4a8

A person entry must have the following items of information: 4a8a type, entry name, percent time, hours worked per week 4a8a1

The entry name is the id of the person. 4a8b

The percent time is the percentage of the hours worked per week that are considered as "overhead". 4a8c

Bugs: 4a9

A bug entry must have the following items of information: 4a9a

type, entry name, entry description, submitter 4a9a1

It may have the following items: 4a9b

pusher, workers, percent time, time left estimate, order, subentries & priorities 4a9b1

Note that the entry description is a description of the bug. 4a9c

In order to get the bugs to appear on the right views, there will be one ongoing task of bug fixing with subentries of ongoing tasks of areas of bugs, e.g., NLS, TENEX, hardware. The ongoing tasks will have pushers and workers, but the bugs themselves only need to have workers if it is known who is actually going to fix it. 4a9d

Needs: 4a10

A need entry must have the following items of information: 4a10a

type, entry name, entry description, submitter 4a10a1

It may have the following items: 4a10b

pusher, workers, subentries & priorities 4a10b1

Note that the entry description is a description of the need. 4a10c

The submitter would be the entry name of the entry that has this need . That entry should have this entry as a subentry (with a priority). However since there are no backlinks yet, this entry will carry with it the name of the needy entry.. 4a10d

The pusher and workers would be people the buyer (submitter) considered as potential contractors. This would be a device for calling the entry to their attention. 4a10e

Possibilities: 4a11

A possibility entry must have the following items of information: 4a11a

type, entry name, entry description, submitter 4a11a1

It may have the following items: 4a11b

pusher, workers, time left estimate, subentries & priorities 4a11b1

Note that the entry description is a description of the possibility. 4a11c

The pusher and workers would be people the submitter considered to be potential buyers. This would be a device for calling the entry to their attention. It would be better to name entries as potential buyers than people. 4a11d

How to distinguish a Need from a Possibility from a task: 4a12

If you are the pusher of an entry that can justify buying some task (you need it), then the entry is a Need, otherwise its a Possibility. 4a12a

Needs become finite or ongoing tasks when all the information requisite for those entries is known (like who are the workers). Possibilities become tasks in the same way (here the important thing is to find a buyer). 4a12b

Of course a lot of tasks will never go thru the need or possibility stage. 4a12c

Basedata Storage Conventions 4b

In this stage, all the items of information for an entry go in one statement. Thus they will all be kept as text strings with some sort of syntactic conventions for recognizing the fields. 4b1

The reason for abandoning the Stage VII device of using NLS structure as field delimiters is that eventually the new file system (property lists) will be used to store the items. With the information all in one statement, the Baseline programs will be pretty much the same for this storage scheme and subsequent schemes -- only the item access mechanisms need be changed. 4b2

Views 4c

There will be two basic kinds of views possible: 4c1

all entries on which a person (or group of people) are workers and/or pushers 4c1a

these entries can be ordered according to: 4c1a1

finite tasks now in progress (order = 0) 4c1a1a

bugs to which he (they) is assigned 4c1a1b

buyer entries 4c1a1c

current ongoing tasks 4c1a1d

future finite tasks according to order 4c1a1e

future ongoing tasks 4c1a1f

possibilities where he (they) is a worker 4c1a1g

possibilities where he (they) was the submitter 4c1a1h

person entry 4c1a1i

there will be options here of including or not including each of the above types of entries 4c1a2

the tree of entries formed by the subentry relation, given any entry as the head of the tree 4c1b

included here is the whole ARC branch and each of the major pursuits of ARC 4c1b1

a plex of entries would be ordered according to the assigned priority 4c1b2

Displays of information from Basedata will have options as to: 4c2 which types of entries 4c2a

which items of information 4c2b

format 4c2c

New Task Procedures 4d

No change. 4d1

Updating Procedures 4e

The basic procedure, i.e., passing out printouts each week to be marked up and returned to the Baseline Maintainer, will remain the same. 4e1

The information to be changed changes. 4e2

In one way the process becomes easier. There will be a program run once a week that will automatically update all the time left estimates for finite tasks in progress. However this should still be examined each week by the pushers for possible updating. 4e3

There will be means of indicating changes or additions to all the other information items, excluding submitter, which should not change. Included here will be means of supplying the information necessary to make a need or possibility entry into a task entry. 4e4

Baseline Record 4f

There will be one view journalized which will be the tree of entries formed by the subentry relation with the level 0 buyer entry ARC as the head of the tree (the person entries will be added). Thus every entry in Basedata will appear at least once. 4f1

It may be desirable to also Journalize a view equivalent to the Stage VIII PERPERS (one branch for the entries of each person). 4f2

Commands 4g

No new commands. 4g1

The command New Entry (formally New Task) must be changed to reflect the different items of information, and which it asks for according to the type of entry. 4g2

The Status and Parameter commands will change because there are different parameters, but that turns out to mostly be some changes to tables. 4g3

Programs 4h

There will still be Baseline sequence generator and content analyzer programs, but they will have to be completely rewritten. 4h1

The sort programs will be completely rewritten. 4h2

The programs effecting the commands will be changed as indicated above. 4h3

There will be a checking program that checks: 4h4

that each entry is formatted correctly 4h4a

that each entry has all the requisite pieces of information (and no items that its not supposed to have) 4h4b

that each entry (except ARC and the person entries) is the subentry of something 4h4c

that each person is allocated between say 90 and 110 percent 4h4d

that there are no circular subentry relations 4h4e

Observations 4i

Note that the information kept for each entry in this stage is mostly the same information that was most likely to be present and reasonably up-to-date in Stage VIII. By not asking for more information than is likely to be received and by retaining the information most immediately (and visibly) useful, it is hoped that the motivation of people to keep the information up-to-date will be increased. 4i1

Another assumed improvement is that the estimate of "ideal" time left to completion of finite tasks plus the percent time for each worker is an easier and more accurate estimate to make than the estimate of the date a task will be finished. Since they are easier, people will be more likely to supply them. Since they are more accurate, they will have to be changed less often (hopefully). 4i2

Probably the largest improvement will be the facility for seeing trees of entries. This seems to be a more useful view than any previously available. 4i3

Note that the subentry relation has no particular meaning. Its just a device for relating entries for the purposes of viewing. "Subcontract" relations should certainly be reflected as subentry relations. 4i4

Note that the definition of a "task" has become looser. For instance if some people preferred to think in terms of "activities", they could create an ongoing task out of that activity (possibly with zero percent time) which had as subentries more specific work as ongoing or finite tasks. 4i5

The reason for the proliferation of types of entries is that all the types were previously there (except buyers and persons) and they were treated differently, so why not make it explicit. 4i6

STAGE XI

5

The change here is in the internal storage mechanisms for the file BASEDATA, i.e., the new file system (property lists) will be exploited. 5a

Basedata Storage Conventions 5b

Each statement in the file Basedata will have two properties: the normal text property and a basedata property. The basedata property will have several "fields". 5b1

The information items will be stored as: 5b2

entry type 5b2a

either a number field in the basedata property of this entry or one (or both) of the name delimiters of this entry 5b2a1

entry name 5b2b

normal text property, which is also the NLS statement name 5b2b1

entry description 5b2c

a text field in the basedata (or perhaps the normal text) property of this entry 5b2c1

pusher 5b2d

an arc named pusher from a person entry to this entry 5b2d1

workers 5b2e

a series of arcs named workers from person entries to this entry 5b2e1

percent time 5b2f

an array of number fields in the basedata property of the person entries; the array is indexed by the arc number of the worker arc connecting the person entry and this entry 5b2f1 time left to task completion estimate 5b2g

a number field in the basedata property of this entry 5b2g1

subentries 5b2h

a series of arcs named subentry from this entry to other entries 5b2h1

priorities 5b2i

an array of number fields in the basedata property of this entry; the array is indexed by the arc number of the associated subentry arc 5b2i1

submitter 5b2j

an arc named submitter from a person entry to this entry 5b2j1

hours worked per week 5b2k

a number field in the basedata property of each person entry 5b2k1

The reason for using all those arcs is speed. The process of producing a view of all the tasks entries on which a person is working or all the entries in a subentry tree should be quite fast. 5b3

Basedata 5c

All order information items and the submitter information item for need entries will be deleted. Since backpointers are now present, the submitter of a need is expressed by a subentry arc from the needy entry to the need entry. Priorities rather than order numbers will be used to order plexes of entries. The priorities can now be reached because of the back pointers. In a by-person view, the priority of an entry will be the total of the priority numbers associated with all the subentry arcs pointing at the entry. In a subentry-tree view, the priority wll remain as the priority associated with the subentry arc by which this entry was reached. 5c1

Programs 5d

There is going to have to be some mechanism provided for editing Basedata. 5d1

STAGE XII

6

This stage involves making a Baseline system that is highly interactive. 6a

One (simple) way would be to have a file (that always reflected the current state of Basedata) in which: 6b

the entry relations were transformed into down/successor relations 6b1

the pusher, worker, and submitter arcs are transformed into text strings in the entry to which they pointed 6b2

the percent time and priority items are moved from the person and superentry entries to the entry at which they were pointed 6b3

and everything is taken out of the basedata property and put into the normal text property 6b4

Another way would be to implement a full-blown subsystem with lots of commands. In particular structural editing would cause changes in subentry arcs rather than NLS structural relations. Each of the information items should be accessible to editing and the right thing ought to happen when they are edited. 6c

STAGES XIII thru CCC 7

The following are not a list of full blown stages, but a list of additional features in a suggested approximate order of addition. 7a

Requirements information item. 7b

It is written by the buyers of the entry. It is a necessary item for all finite tasks, ongoing tasks, and needs. 7b1

Design information item 7c

It is written (or approved) by the pusher of the entry. It is a necessary item for all finite tasks. It may be present in ongoing tasks and possibilities. 7c1

Calculate estimated task end-dates 7d

A facilty for calculating the estimated end-dates of finite tasks. This could be presented in calendar form. 7d1

More detail on each persons overhead and/or ongoing tasks 7e

Develop more detail on where each person's time is spent. 7e1

Data kept on actual time spent on each task 7f

Each worker would be asked to keep track of where his time was actually spent. Information items could be added to Basedata to store this information. 7f1

Integrate accounting system and BRS 7g

We assume that any ARC accounting system would eventually be very closely tied to the BRS. Keeping track of resources spent could be tied in with the entry organization of Basedata. Account numbers could be assigned to each buyer and thus to tasks. 7g1

Graphic portrayal of various views 7h

It would be nice to get some of the views of Basedata and various accounting information portrayed with graphics. 7h1

Automatic updating of overhead estimates and fudge factors 7i

Each week the data on actual time spent and the history of estimates would be compared. Then the infomation in Basedata on overhead, percent times, and fudge factors would be updated appropriately. 7i1

Proposed Scenario for the Baseline Record System

(J8168) 2-DEC-71 14:55; Title: Author(s): Bruce L. Parsley, Charles H. Irby/BLP CHI; Distribution: Marilyn F. Auerbach, Walter L. Bass, Mary S. Church, William S. Duvall, Douglas C. Engelbart, Beauregard A. Hardeman, Martin E. Hardy, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, John T. Melvin, Jeanne B. North, James C. Norton, Bruce L. Parsley, Ed K. Van De Riet, Dirk H. van Nouhuys, Kenneth E. Victor, Don C. Wallace, Richard W. Watson, Don I. Andrews, William R Ferguson/PRO; Sub-Collections: SRI-ARC PRO; Clerk: BLP;
Origin: <MSR>BPROP.NLS;12, 2-DEC-71 14:34 BLP ;

BLP 2-DEC-71 15:04 8169

Baseline Meeting

I would like to call a meeting about the Baseline for Tuesday, December 7 at 2:00 PM. I would like to use (Journal, 8168,) -- Proposed Scenario for the Baseline Record System -- as a basis for discussion.
Since small meetings seem to work better than large ones, don't feel compelled to attend unless you feel compelled. 1

Baseline Meeting

(J8169) 2-DEC-71 15:04; Title: Author(s): Bruce L. Parsley/BLP; Distribution: Walter L. Bass, William S. Duvall, Mary S. Church, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, John T. Melvin, Bruce L. Parsley, William H. Paxton, Douglas C. Engelbart, John T. Melvin, James C. Norton, Ed K. Van De Riet, Kenneth E. Victor, Don C. Wallace, Richard W. Watson/NPG DCE JTM JCN EKV KEV DCW RWW; Sub-Collections: SRI-ARC NPG; Clerk: BLP;

Is there a way to exercise more format control in the "output device" command to a tty? I often use tty's or other devices which have tabs, or form feed, or other capabilities, and I find it hard to position the paper reasonably with the "odt" command as it now seems to work. Thanx 1

(J8170) 2-DEC-71 15:48; Title: .HED=""output device" command query"; Author(s): Joel B. Levin/JBL; Distribution: Richard W. Watson/RWW; Sub-Collections: NIC; Clerk: JBL;

KEV 3-DEC-71 1:02 8171

new printer driver

There will not be a header page between copies 1

KEV 3-DEC-71 1:02 8171

new printer driver

(J8171) 3-DEC-71 1:02; Title: Author(s): Kenneth E. Victor/KEV; Distribution: Marilyn F. Auerbach, Walter L. Bass, Roger D. Bates, Mary S. Church, William S. Duvall, Douglas C. Engelbart, Beauregard A. Hardeman, Martin E. Hardy, Fred P. Hocker, J. D. Hopper, Charles H. Irby, Mil Jernigan, Harvey G. Lehtman, John T. Melvin, Jeanne B. North, James C. Norton, Cindy Page, Bruce L. Parsley, William H. Paxton, Jeffrey C. Peters, Jake Ratliff, Barbara E. Row, Ed K. Van De Riet, Dirk H. van Nouhuys, Kenneth E. Victor, Don C. Wallace, Richard W. Watson, Don I. Andrews, James A. Fadiman/SRI-ARC; Sub-Collections: SRI-ARC; Clerk: KEV;

new printer driver

This document overrides JOURNAL 8120 1

There is now a program to drive the printer as a background job. 2

This program takes files from the directory PRINTER and copies them to the line printer. 2a

After the files have been printed they are thn deleted and expunged. 2b

Files are taken out of the directory in GTJFN order. 2c

The header page for these files will consist of three lines of the file extension name (if it is non numeric) followed by three lines of the file name. 2d

Do not use as a filename PNTFL 2d1

If the extension name is a number, then you will get that many (decimal) copies of the file 2d2

For these purposes a numeric extension name is a name that starts with a number, e.g. XXX.2Z will give you 2 copies of the file 2d2a

There will not be a header page between copies 2d3 new printer driver

(J8172) 3-DEC-71 1:24; Title: Author(s): Kenneth E. Victor/KEV; Distribution: Marilyn F. Auerbach, Walter L. Bass, Roger D. Bates, Mary S. Church, William S. Duvall, Douglas C. Engelbart, Beauregard A. Hardeman, Martin E. Hardy, Fred P. Hocker, J. D. Hopper, Charles H. Irby, Mil Jernigan, Harvey G. Lehtman, John T. Melvin, Jeanne B. North, James C. Norton, Cindy Page, Bruce L. Parsley, William H. Paxton, Jeffrey C. Peters, Jake Ratliff, Barbara E. Row, Ed K. Van De Riet, Dirk H. van Nouhuys, Kenneth E. Victor, Don C. Wallace, Richard W. Watson, Don I. Andrews, James A. Fadiman/SRI-ARC; Sub-Collections: SRI-ARC; Clerk: KEV;
Origin: <VICTOR>DOC-PRNTR.NLS;3, 3-DEC-71 0:53 KEV ;

SETMB - Set/read mouse buttons JSYS - JSYS 515 1

(to be implemeted soon) 2

ACCEPTS: 2a

in 1: A byte pointer to a .string where: 2a1

The first byte of the string is the 7-bit definition of the command accept button 2a1a

The second byte of the string is the 7-bit definition of the command delete button 2a1b

The third byte of the string is the 7-bit definition of the soft delete button 2a1c

If mouse buttons are being read, a string pointer to where to store the current settings of the mouse buttons 2a1d

in 2: 0 - means set mouse buttons
.ne. 0 - means read mouse buttons 2a2

ACTION: 2b

The 7-bit definitions are what will be entered into the system when the command accept, command delete, or soft delete mouse buttons are pushed. (The corresponding keyboard keys are: command accept, command delete, and backspace.) 2b1

RETURNS: 2c

+1: always, with an updated string pointer in 1 if mouse buttons being read 2c1

STRMT - Set terminal type -JSYS 523 1

(to be implemented soon) 2

3

Accepts: 3a

in 1: 3a1

bits 0-17 - LINE NUMBER (0 means controlling line) 3a1a

must be a wheel or an operator to set type for lines other than your own 3a1a1

bits 26-29 - GRID TYPE 3a1b

currently: 3a1b1

0 - Local displays 3a1b1a

1 - IMLACS with long vectors 3a1b1b

2 - IMLACS without long vectors 3a1b1c

bits 30-35 - TERMINAL TYPE 3a1c

currently: 3a1c1

0 - teletype 3a1c1a

1 - local display 3a1c1b

2 - processor display 3a1c1c

3a2

Returns: 3b

+1: error, with error number in 1 3b1

WHELX1 - trying to set type for other than controlling teletype and not a wheel or operator 3b1a

STRX1 - illegal combination of parameters 3b1b

+2: successful return 3b2

JBL 3-DEC-71 8:23 8175

Query about Journal System and EXEC

This is a second attempt--please let me know if you get this

JBL 3-DEC-71 8:23 8175

Query about Journal System and EXEC

This is a resubmittal of the message I sent you earlier (as I mentioned on the phone), in case I don't get confirmation that you received it. 1

John, 2

A question about the journal system: 2a

The journal users guide we have only talks about submitting messages and documents, not about retrieving them. Specifically, it mentions the option of specifying "keywords", which, presumably, are a means of finding entries by subject. 2a1

Is there in fact a way of locating an entry by keyword? If so, is it documented? 2a2

About your EXEC: 2b

It really will be quite useful for us here if we have a way of examining the status of our connection as TENEX sees it. 2b1

Are there any plans for implementation of a means whereby a network user can get a NETSTAT, either in EXEC or by means of a separate socket? 2b2

Thanks for your help. 2c

This is now the second message I have ever sent. 2d

Pax

Joel 2e

Query about Journal System and EXEC

(J8175) 3-DEC-71 8:23; Title: Author(s): Joel B Levin/JBL; Distribution: John T. Melvin, Ellen Westheimer/JTM EW; Keywords: NETSTATUS journal keyword; Sub-Collections: NIC; Obsoletes Document(s): 8165; Clerk: JBL;

JBL 3-DEC-71 8:48 8176

mumble

Ellen,

The other funny message you have there from me is just from messing around with the journal. I just wanted to see if you got it. (Did you?)

Pax

J. 1

mumble

(J8176) 3-DEC-71 8:48; Title: Author(s): Joel B Levin/JBL; Distribution: Ellen Westheimer/EW; Sub-Collections: NIC; Clerk: JBL;

Extension to Proposed Debugger Parser

RE: (Journal, 8162, 1b1b), Examine Field command. Bruce sggests that if the word which is the field name is of the form x.y, then the Record Pointer should be moved to x and field y displayed. I think this is a very good idea. 1

Extension to Proposed Debugger Parser

(J8177) 3-DEC-71 9:31; Title: Author(s): Charles H. Irby/CHI; Distribution: Walter L. Bass, William S. Duvall, Mary S. Church, J. D. Hopper, Charles H. Irby, Harvey G. Lehtman, John T. Melvin, Bruce L. Parsley, William H. Paxton, L. Peter Deutsch, James G. Mitchell, Don I. Andrews/NPG LPD JGM DIA; Sub-Collections: SRI-ARC NPG; Clerk: CHI;

This journal entry respondes to (journal,8064), Bruce and Chuck first's thoughts about bigger and better baseline systems 1

Reading your journal item turns my mind to my past modest involvement with PERT. 2

It makes me imagen a system that would do for the individual toiler what your system does and what PERT is supposed to do for aerospace and construction program mangers. 2a

Little Tasks and How you Sum them: 3

When PERT works, it does so because the planner cuts the job into pieces small (familiar) enough to allow useful estimates of time to completion. 3a

His procedure assumes that the hard-to-plan big job is made up of familiar pieces. 3a1

Parts of ARC'S work are so developmental as to be predictable only in very small pieces. 3b

It seems to me that one problem with the present baseline system is that tasks are often too large to allow meaningful estimates to completetion, considering their unfamiliarity. 3b1

Yet entering tiny tasks is a bore for the toiler and creates a list of tasks too long to survey. 3b1a

Moral: 3c

Make it very easey to enter tasks. 3c1

Make a wonderful system of viewspecs that generate summary views. 3c2

Net vs Tree 4

PERT also works by displaying interdependence. 4a

If a management system is to serve an individual by letting him feel how his plans affect the BIG PICTURE (Remeber Paul Rech's remark about the janitor at the oil plant), it must knit and expose the interdependence among completion dates (and hence among levels of efort). 4b

That is, it must be a net, not a tree. 4b1

As long as a management system is a tree and not a net, it is merely a record keeping system, it's interaction with the f*ture is merely keeping records of predictions. 4c

Can we build something that will do this: 4d

If the date of completion of job A depends on the date of completions of job B, and some one then enters a new job T that also de pends on A, then when the doer of B goes and asks the system what will happen if he haves his effort on B, it will tell him that he will delay by so much the completion of T, which he had never heard of before? 4d1

I imagen a coommand: w a c [what are the consequences?]. 4d2

Finally there is PERT-COST. 5

PERT-COST does for dollars

more or less what PERT d es for time. 5a

Dollars could be added to what

what you suggest in mechanisms paralell to those that serve time. 5b

But

when we come to match ideal dollars against U. S. dollars, the

reconciliation might be more painful even than the reconciliation with

time. 5b1

(J8178) 3-DEC-71 11:06; Title: Author(s): Dirk H. van Nouhuys/DVN;
Distribution: Charles H. Irby, Bruce L. Parsley, James C. Norton/CHI BLP
JCN; Sub-Collections: SRI-ARC; Clerk: DVN;
Origin: <VANNOUHUYS> BASEPERT.NLS;2, 3-DEC-71 10:49 DVN ;";

CHI 3-DEC-71 12:58 8179

Toward a More Consistent Command Language in NLS

Introduction 1

I (CHI) recommend that the following scheme be implemented in the NLS command language. 1a

The command language should be made to consist of 1a1

top level commands 1a1a

frequently used commands, which are recognized by their first letter 1a1a1

The Jump commands should be made to be like the rest -- no state of its own. 1a1a2

JUmp to Successor, Jump to Predecessor requires one to type 'J 'S 'J 'P just as Insert Statement, Insert Word requires one to type 'I 'S 'I 'W. 1a1a2a

the sub commands of Jump to End should be deleted. 1a1a3

The load file command should be eliminated, since links can be used to accomplish the same thing (it seems reasonable to encourage people to use links). 1a1a4

second level commands 1a1b

infrequently used, new, or experimental commands, which are not recognized by their first letter. Recognition should be requested by typing a CA, ALTMODE, or SP (as in TENEX). Recognition within these second level commands should also be upon request. 1a1b1

This allows new commands to be added without worrying about first letter conflicts -- a significant problem currently. 1a1b1a

These commands can be repeated by making further bug selections. However, if a non-bug selection character is input it will be parsed at the top level (as is now the case). 1a1b1b

subsystems. 1a1c

The subsystem name is recognized upon request. Once in the subsystem the same pattern should be applied (top level with first character recognition, second level with recognition upon request, and subsystems with subsystem name recognition upon request). All subsystems should terminate with the "Quit" command. 1a1c1

The concept of ADDRESS EXPRESSION should be generalized for both DNLS and TNLS (maybe DEX someday), and whereever a statement name or number is currently used, an ADDRESS EXPRESSION should be allowed. 1a2

This provides a powerful extention to the link syntax, but should be compatible with extant links. 1a2a

Please respond. No response is assumed affirmative. 1b

Details 2

Top level commands -- frequently used commands, recognized by first letter 2a

append 2a1

break 2a2

copy 2a3

delete 2a4

execute (second level command) 2a5

freeze (DNLS) Fix Marker (TNLS) 2a6

goto subsystem 2a7

h (unused) 2a8

insert 2a9

jump (DNLS) j (unused) (TNLS) 2a10

Works like other commands -- no state of its own 2a10a

no substructure for Jump to End 2a10b

k (unused) 2a11

l (use links with literal typein instead of load file) 2a12

move 2a13

n (unused) 2a14

output 2a15

p (unused) (DNLS) print (TNLS) 2a16

quit 2a17

replace 2a18

substitute 2a19

transpose 2a20

update 2a21

viewspecs 2a22

w (unused) 2a23

x (unused) 2a24

y (unused) 2a25

z (unused) 2a26

'. (TNLS) show Current Statement Pointer 2a27

'" (TNLS) Comment 2a28

'/ (TNLS) type context of Current Statement Pointer 2a29

' (TNLS) print statement 2a30

linefeed (TNLS) print next statement 2a31

'↑ (TNLS) print back statement 2a32

SP ((TNLS) move Current Statement Pointer 2a33

Second level commands — infrequently used commands. Recognition upon request. 2b

assimilate 2b1

browse mode 2b2

content analysis 2b3

display area format, device set (e.g display, TI-terminal) 2b4

DA format: (vertical split, horizontal split, move boundary, character size) 2b4a

e (unused) 2b5

force case (upper/ lower) 2b6

g (unused) 2b7

h (unused) 2b8

insert sequential 2b9

j (unused) 2b10

k (unused) 2b11

l (unused) 2b12

merge, markers 2b13

new file, name delimiters 2b14

ownership 2b15

partial copy (delete/ undelete/ reset) 2b16

q (unused) 2b17

r (unused) 2b18

status, sort 2b19

t (unused) 2b20

u (unused) 2b21

verify file 2b22

w (unused) 2b23

x (unused) 2b24

y (unused) 2b25

z (unused) 2b26

susystems -- subsystem name recognition upon request. 2c a (unused) 2c1

baseline 2c2

catalog, calculator (when it happens) 2c3

debugger (when it happens) 2c4

edit (TNLS) 2c5

f (unused) 2c6

g (unused) 2c7

h (unused) 2c8

identification 2c9

journal 2c10

k (unused) 2c11

l (unused) 2c12

measurement 2c13

n (unused) 2c14

o (unused) 2c15

p (unused) 2c16

q (unused) 2c17

r (unused) 2c18

s (unused) 2c19

t (unused) 2c20

user programs 2c21

vector package (DNLS) (when it happens), viewchange (TNLS) 2c22

w (unused) 2c23

x (unused) 2c24

y (unused) 2c25

z (unused) 2c26

ADDRESS EXPRESSIONS -- To improve one's ability to specify a location within a file, I suggest the following generalization: Make the notion of an ADDRESS EXPRESSION, now available only to TNLS user's, available to DNLS users also. This ADDRESS EXPRESSION should be consistent with existing links, the same ADDRESS EXPRESSION should work in TNLS and DNLS, and the elements of the expression should be reasonably mneumonic. An ADDRESS EXPRESSION should be available in NLS whereever a statement number or statement name is now used (as in links, jumps, etc.). This ADDRESS EXPRESSION should be as follows: 2d

NOTE: ALTMODE should be equivilent to CA in DNLS and TNLS. 2d1

ADDRESS EXPRESSION elements 2d2

location number 2d2a

A statement number is D $(L/ D). 2d2a1

no alphabetic zero ('@). 2d2a1a

name 2d2b

A statement name is as defined by the name delimiter routine -- currently defined to be L $(L/ D/ ''/ '-). 2d2b1

Note that '' and '- are included in this kludgy definition of name so that attempts to terminate a name in an ADDRESS EXPRESSION with an element begining with '' or '- will not work properly. To get this sequence to work, one will have to terminate the name with a SP. 2d2b1a

(I suggest that at some point in time the definition of statement name could stand some improvement -- like make it L $(L/ D) and use upper-lower case letters (e.g. SriArc) and format routines to put in arbitrary special characters when printing or interacting with the user, as in the ident system) 2d2b1a1

A sequence of digits and letters preceeded immediately by a period can contain the following letters, with associated "Jump" meaning. NOTE: default value for <number> is 1. 2d2c [number]'s jump to successor <number> times 2d2c1

[number]'p jump to predecessor <number> times 2d2c2

[number]'u jump to up <number> times 2d2c3

[number]'d jump to down <number> times 2d2c4

[number]'a jump to ahead <number> times 2d2c5

[number]'r jump to return <number> times 2d2c6

[number]"fa" jump to file ahead <number> times 2d2c7

[number]"fr" jump to file return <number> times 2d2c8

[number]'o jump to origin 2d2c9

[number]'e jump to end 2d2c10

[number]'n jump to next <number> times 2d2c11

[number]'b jump to back <number> times 2d2c12

[number]'h jump to head 2d2c13

[number]'t jump to tail 2d2c14

[number]'l jump to the <number>th link 2d2c15

[number]'w jump to next occurance of word <number> times 2d2c16

[number]'c jump to next occurance of content <number> times 2d2c17

a sequence of digits and letters preceeded immediately by a plus (skip forward) or minus (skip backward) can contain the following letters, with associated meaning. NOTE, default value of <number> is 1. 2d2d

[number]'c skip <number> characters 2d2d1

[number]'w skip <number> word 2d2d2

[number]'v skip <number> visible 2d2d3

[number]'i skip <number> invisible 2d2d4

```
'* name   jumps to the next statement by that name            2d2e

'( text ')   link                                             2d2f

   text = [USERNAME ',] [FILENAME ',] [ADDRESS
   EXPRESSION] [': VIEWSPECS]                                 2d2f1

'[ text ']   content search                                   2d2g

   text excludes ']                                           2d2g1

'< text '>   word search                                      2d2h

   text excludes '>                                           2d2h1

'; text ';   intra-statement content search                   2d2i

   text excludes ';                                           2d2i1

'' character   character search                               2d2j

←   beginning of statement                                    2d2k

>   end of statement                                          2d2l

'# text   marker name, text = L $(L/D)                        2d2m
```

Toward a More Consistent Command Language in NLS

(J8179) 3-DEC-71 12:58; Title: Author(s): Charles H. Irby/CHI; Distribution: Marilyn F. Auerbach, Walter L. Bass, Roger D. Bates, Mary S. Church, William S. Duvall, Douglas C. Engelbart, Beauregard A. Hardeman, Martin E. Hardy, Fred P. Hocker, J. D. Hopper, Charles H. Irby, Mil Jernigan, Harvey G. Lehtman, John T. Melvin, Jeanne B. North, James C. Norton, Cindy Page, Bruce L. Parsley, William H. Paxton, Jeffrey C. Peters, Jake Ratliff, Barbara E. Row, Ed K. Van De Riet, Dirk H. van Nouhuys, Kenneth E. Victor, Don C. Wallace, Richard W. Watson, Don I. Andrews, James A. Fadiman/SRI-ARC; Sub-Collections: SRI-ARC; Clerk: CHI;
Origin: <IRBY>NEWPARSER.NLS;10, 3-DEC-71 12:54 CHI ;