Procedure Call Protocol Documents

Table of Contents

PCP Table of Contents

(J24851) 30-DEC-74 10:22;;;; Title: Author(s): Jonathan B. Postel/JBP; Distribution: /JBP([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JBP; Origin: < POSTEL, PCPJUNK.NLS;1, >, 30-DEC-74 10:04 JBP ;;;;####;

Cover Letter

This note announces release of the second published version of the Procedure Call Protocol -- PCP Version 2. Version 2 is SUBSTANTIALLY different than Version 1; it and all intermediate, informally distributed PCP documents are obsoleted by this release. 1

Version 2 consists of the following documents. Each is available on-line in two forms: as an NLS file and as a formatted text file. The Journal number (e.g. 24459) refers to the former, of course, and the pathname (e.g. [SRI-ARC]<NLS>PCP.TXT) to the latter, accessible via FTP using USER=ANONYMOUS and PASSWORD=GUEST (no account required). Hardcopy is being forwarded by US Mail to all those who have expressed an interest in PCP. If you don't receive a copy and would like one of this and/or future releases, send a note to that effect to WHITE@SRI-ARC: 2

(20391,) "Some Thoughts on System Design to Facilitate Resource Sharing" 2a

This document discusses issues present in the design of network based resource sharing systems. 2a1

PCP (24459,) "The Procedure Call Protocol" 2b

This document describes the virtual programming environment provided by PCP, and the inter-process exchanges that implement it. 2b1

Pathname: [SRI-ARC]<NLS>PCP.TXT 2b1a

PIP (24460,) "The Procedure Interface Package" 2c

This document describes a package that runs in the setting provided by PCP and that serves as a procedure-call-level interface to PCP proper. It includes procedures for calling, resuming, interrupting, and aborting remote procedures. 2c1

Pathname: [SRI-ARC]<NLS>PIP.TXT 2c1a

PSP (24461,) "The PCP Support Package" 2d

This document describes a package that runs in the setting

provided by PCP and that augments PCP proper, largely in the area of data store manipulation. It includes procedures for obtaining access to groups of remote procedures and data stores, manipulating remote data stores, and creating temporary ones. 2d1

Pathname: [SRI-ARC]<NLS>PSP.TXT 2d1a

PMP (24462,) "The Process Management Package" 2e

This document describes a package that runs in the setting provided by PCP and that provides the necessary tools for interconnecting two or more processes to form a multi-process system (e.g. NSW). It includes procedures for creating, deleting, logically and physically interconnecting processes, and for allocating and releasing processors. 2e1

Pathname: [SRI-ARC]<NLS>PMP.TXT 2e1a

PCPFMT (24576,) "PCP Data Structure Formats" 2f

This document defines formats for PCP data structures, each of which is appropriate for one or more physical channel types. 2f1

Pathname: [SRI-ARC]<NLS>PCPFMT.TXT 2f1a

PCPHST (24577,) "PCP ARPANET Inter-Host IPC Implementation" 2g

This document defines an implementation, appropriate for mediating communication between Tenex forks, of the IPC primitives required by PCP. 2g1

Pathname: [SRI-ARC]<NLS>PCPHST.TXT 2g1a

PCPFRK (24578,) "PCP Tenex Inter-Fork IPC Implementation" 2h

This document defines an implementation, appropriate for mediating communication between processes on different hosts within the ARPANET, of the IPC primitives required by PCP. 2h1

Pathname: [SRI-ARC]<NLS>PCPFRK.TXT 2h1a

PCPTNXINT (24792,) "Tenex PCP Process Internal Structure" 2i

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. 2ii

Pathname: [SRI-ARC]<NLS>PCPTNXINT.TXT 2iia

The first document, PCP, is the place the interested reader should start. It gives the required motivation for the Protocol and states the substance of the Protocol proper. The reader may then, if he chooses, read the next three documents: PIP, PSP, and PMP. The latter has the most to offer the casual reader; the programmer faced with coding in the PCP environment should read all three. The final few documents -- PCPFMT, PCPHST, and PCPFRK -- are of interest only to the PCP implementer. The final (and most recent) document should be of interest to implementers of the PCP mechanisms in TENEX. 3

JBP 30-DEC-74 11:19 24852
PCP Cover Letter

(J24852) 30-DEC-74 11:19;;;; Title: Author(s): Jonathan B.
Postel/JBP; Distribution: /JBP([INFO-ONLY]) ; Sub-Collections:
SRI-ARC; Clerk: JBP; Origin: < POSTEL, PCPJUNK,NLS;3, >,
30-DEC-74 11:17 JBP ;;;;####;

NSW Frontend Issues -- reflects Decisions of Nov NSW Review Meeting

INTRODUCTION 1

The concept of the NSW Frontend is a very important yet, we think, not fully understood part of the NSW as a whole and is, we believe, very vital to the overall success of the NSW program. This paper attempts to clarify the Frontend concept somewhat, to raise some issues that have come up so far, and to present our current thoughts on those issues. This is a draft document and has been updated to reflect the outcome of the NSW Project Review meeting held Nov 6,7, and 8 at SRI (attendees: Millstein, Balzer, Crocker, Watson, Irby, White, Postel, Waal, Triolo, Michael, Lehtman, Victor, and other SRI staff). The decisions reached at this meeting are denoted by []'s. 1a

THE PURPOSE OF THE FRONTEND FOR THE NSW 2

The Frontend is a buffer between the user and the Works Manager and tools of the NSW. The Frontend provides a logical function (interfacing the user to the NSW) and will initially be implemented on a PDP-11 satellite computer. In the future, the Frontend may consist of a program running on a satellite cooperating with a program running on some larger computer. The primary reasons for having a Frontend, as we perceive them, are as follows: 2a

1) To provide the user with a coherent and consistent command language discipline throughout the NSW. 2a1

No matter whether the user is giving commands to a tool or to the Works Manager or the Frontend itself, he does so using the same methods for specifying which commands he wishes executed, the same methods for specifying arguments or parameters to commands, gets the same type of prompting and requests help in the same way, always. In addition, the general syntactic form(s) should be the same from tool to tool unless there is good reason for the tool to deviate from the standard. Of course the particular commands and vocabularies will vary with the tool and in fact the same verbs may be used with quite different semantics in different tools, but at least most other facits of the command language (including asking for help and being prompted for the proper type of input) should stay the same across tool boundaries. 2a1a

It is expected that initial users of the NSW will have to access some tools in a "transparent" mode [see discussion below in Issue 1], where the Frontend cannot provide the user with the facilities just described. In this instance, the Frontend could take on the nature of a TIP or ELF

terminal concentrator. However, we expect that as time goes on and the NSW grows, the user will be able to use most tools through the unified user interface provided by the Frontend, and it is toward this end that we should build. 2a1b

2) To provide tools with well-formed commands. 2a2

It is proposed that this be done by issuing remote procedure calls to "external" procedures in the tools to actually execute commands. This will be accomplished through the Procedure Call Protocal (see Jim Whites papers on the PCP). 2a2a

Many operating systems and application programs have elected to use half duplex, line-at-a-time terminals because of the increased computer efficiency provided by this approach. Other operating systems and application programs have chosen, instead, to utilize character-at-a-time full duplex terminal disciplines because of the opportunity this provides for utilizing a more human-engineered command language. 2a2b

The NSW Frontend is an attempt to combine these two approaches into a COMMAND-AT-A-TIME system, where the application programs do not directly interact with the terminal, but rather receive fully specified commands from the Frontend. At the same time, the Frontend will attempt to provide the user with the best possible human-engineered command language discipline. This means that interfacing to application programs developed for line-at-a-time terminals should be quite straight forward, even though the user operates from a character-at-a-time full duplex terminal, since such programs tend not to interact with the user extensively [See discussion under Issue 2 below]. 2a2c

3) To provide a terminal-independent interface to the tools. 2a3

Because the Frontend handles all terminal interaction (except in transparent mode), it will present to the tool a virtual terminal. Thus, once a tool is developed, little attention need be given to the type or particular characteristics of the terminal the end user may choose to employ while using the tool. In fact, the cost of creating new tools should be considerably reduced because of the facilities made available by the Frontend. 2a3a

This means that even though the creators of a tool envisioned the user sitting at a typewriter terminal, the NSW user who happens to be using a display terminal with a pointing device may be able to interact with the tool

in a two dimensional sense, pointing to arguments on his screen instead of typing them, etc. [See discussion under Issue 2 below] 2a3a1

For tools which wish to make more extensive use of a display terminal if the user has one, the Frontend presents primitives for allocating windows on the display and allows the tool to write/delete/move/make invisible items displayed within the windows. 2a3a2

4) Possible asynchronous operation. 2a4

In some instances, it may be possible for the execution of the user's commands to be accomplished in parallel with subsequent command specification and execution. This frees the user to do other things while a lengthy command is being executed by a tool. 2a4a

5) NSW-wide macro facilities 2a5

The user should be able to define (text substitution) macros which he can then use with any tool, since the macros will be expanded by the Frontend. 2a5a

[NOV 6-7 DECISION: 2a5b

A macro facility will probably not be made available for first year NSW.] 2a5b1

6) To provide standard mechanisms for presenting status or error conditions to the user. 2a6

an error should consist of the following: 2a6a

a human readable error message 2a6a1

a code indicating whether this error caused the command to be aborted, completed or undefined and whether the tool is now in a state to receive more commands or should be restarted. 2a6a2

certain types of errors will have to be reported to the Works Manager so that it can take action (e.g. file system errors, disk errors, etc). Does the Frontend do this or does the tool? [Could Millstein or Warshall address this?] 2a6b

In particular, from the user's standpoint, the first and last of these justifications are very important. With very few exceptions, most user's would become very frustrated in an

environment where every new tool he chose to use required that he learn a new interaction discipline. (Imagine having to speek Greek to the gas station attendant, Spanish to the grocer, French to the waiter, and so forth.) In fact, it is our belief that this would spell certain doom for the NSW. 2b

The Frontend functions could be much more extensive and could be distributed between the satellite PDP-11 and a larger host. This notion will have to evolve as we gain experience with the NSW as a whole. 2c

ISSUES THAT HAVE COME UP SO FAR 3

Some of the issues that should be raised now are the following: 3a

1) What is to be done for tools that already exist as monolithic packages where the user interaction cannot readily be separated out into the NSW Frontend? 3a1

It is our belief that the more tools which are fully integrated into the NSW framework the better the NSW will seem as a total system. However, we must also allow users to run tools which were not built or modified to run within the NSW. Aside from the impact this has on the Works Manager, the Frontend cannot be expected to help the user when he is using this type of tool. It is envisioned that a transparent mode (with a user-settable escape character) will be provided for such tools and that tool output to the user will appear as though the user had a TTY, even though he may be using a display. 3a1a

[NOV 6-7 DECISION: 3a1b

From the Frontend's standpoint, all tools have a grammar that drives the Frontend's interaction with the user. For some tools, however, this grammar may be very trivial and may cause a trivial backend to be invoked (through PCP) that simply passes characters to the actual tool. This means that the tool itself need not be modified, but that the user interacts with the tool just as he interacts with other tools and that he still has easy access to the Works Manager and to the Frontend itself. 3a1b1

This means that to install a tool into the NSW, someone must write the CML grammar for it (this may be very trivial, such as just collecting a string of characters from the user and passing them to the backend) and construct a simple backend which will interact with the frontend through PCP procedure calls

and ship characters to/from the real tool and perhaps handle some error conditions. It may be possible to use the same grammar and simple backend for several such tools if the simple backend can treat the real tools the same all the time. 3a1b1a

If it turns out to be trivial to also supply a transparent mode that allows for trial, evaluative use of a tool without any investment in a grammar, etc, then this will also be provided.] 3a1b2

2) What classes of terminals should be supported in the first year? 3a2

A Network Virtual Terminal (NVT) was defined for TELNET. We would propose that in addition to this definition of a virtual typewriter terminal, the NSW needs a similar definition of a virtual alpha-numeric display and a virtual display with a pointing device. 3a2a

It is our assumption that it is ADR's responsibility to write drivers for any terminals the NSW management decides the NSW Frontend will support to map them into the appropriate virtual terminal. 3a2b

The Frontend will provide external procedures to present error and status messages to the user and, if he is using a display terminal, to allocate and manipulate text and graphics within windows on the display. A tool will not, however, be able to effect certain areas of the screen which are used by the Frontend for command feedback. 3a2c

While many application programs in conventional operating systems now make use of line-at-a-time terminals, as discussed above, these terminals do not allow the Frontend to actively interact with the user to answer help requests, to provide keyword recognition, noise words, or to prompt him for various types of input. For these reasons, the NSW Frontend should perhaps support half duplex terminals for printing devices only (since the best hardcopy seems to be produced on some of these devices). 3a2d

Proper support of half duplex terminals as interation devices may call for quite a different interaction strategy on the part of the Frontend. Should we be planning on this or just support them awkwardly, as is now done throughout much of the ARPANET? 3a2d1

[NOV 6-7 DECISION: 3a2e

SRI will support half duplex, line-at-a-time terminals as interation devices. ARC currently believes that the specification of the command language for a tool should be, for the most part, independent of the terminal class being used with the tool. There is now a facility in the CML for the command language designer to specify that certain commands (or parts thereof) should not be available for certain terminal classes. This facility will probably have to be expanded to account for half duplex terminals. CHI will think about interaction form for a half duplex terminal -- the Frontend may be able to do special, nicer things for users at such terminals. 3a2e1

SRI (Victor) will define the following virtual terminal classes for NSW: 3a2e2

1) Half duplex, line-at-a-time terminal, 3a2e2a

2) NVT (already defined), 3a2e2b

3) Alpha-numeric display without pointing device and with and without editing functions, and 3a2e2c

4) Line Processor-enhanced alpha-numeric display (already defined). 3a2e2d

The definition of a virtual graphics terminal may be deferred until next year (May be able to use Network Graphics Protocol here.). 3a2e3

A facility should be made available for alpha-numeric display terminals to automatically page output (halt output until the user types a character to continue outputting). In addition, it would be desireable from such a terminal to be able to scroll back though recent tty output that is no longer on the screen. The frontend will accomodate this within memory size limits (may have to dump it on a file somewhere).] 3a2e4

3) Is the Command Meta Language flexible enough for first-year NSW (documentation of currently planned language will be available shortly)? 3a3

As currently planned, the NSW CML will be derived from the current CML, used in the NLS system. The re-implementation and generalization of this CML will remove problems that we have found during its use for the past year within NLS but are there other steps we should take to generalize it even further? 3a3a

We are now in the process of writing trial grammars for TECO, WYLBUR, CANDI, NETED, SOS, and two forms of DDT. A discussion of these will be distributed shortly. Are there other interactive systems that people know of that we should try to express in CML at this time? 3a3b

[NOV 6-7 DECISION: 3a3c

It was felt that if the CML could actually be used to describe the above set of user languages that it would be sufficient for first year NSW use. We were able to specify one of the most complicated commands in the Works Manager with no difficulty. We will bring up any problems that arise. 3a3c1

Bob Balzer felt strongly that the user should be able to execute "universal" (always available) commands to the Frontend or Works Manager even if these conflict with tool-specific commands. Please note that this imposes a limitation on tool command languages since they cannot use commands that begin with the same command words as the Frontend or Works Manager commands (Warshall was opposed to this in a meeting in Atlanta, feeling that it would be better to have an escape character which the user typed in order to talk to the Frontend or Works Manager). We have used the "universal" command strategy within NLS with little problem of conflicts. This approach seems (to us) to present a simpler user interface to the overall system but either approach is acceptable to us. It seems more discussion is needed here.] 3a3c2

4) What language form should be the NSW standard? 3a4

We currently envision NSW commands that will consist of command words (which might serve as verbs for commands or as modifiers for partialy specified commands), parameter specifications (pointing operations for users at displays and typing parameters), standard command confirmation (final ok to proceed), standard backup to a previous state in the specification of the current command, standard ways of determining the currently available alternatives or the general syntactic form of commands, and general semantic (data base driven) help for particular tools and commands within those tools. 3a4a

Should the NSW CML attempt to enforce some conventions about the form of commands, such as specifying command words at the begining of commands followed by parameters or vice

versa? Should the recognizers for some low-level parameter types, such as word, text-string, and file name, be built into the CML so that they will be the same for all tools? 3a4b

Should post fix be allowed? Should functional notation be allowed? 3a4c

[NOV 6-7 DECISION: 3a4d

It was felt that the CML should not be changed to be more restrictive in an attempt to enforce a standard NSW command language convention. This can be done later when the issues are more clearly understood. Although this issue was not discussed at length, we feel that certain commonly used recognizers should be built into the frontend (to facilitate writing simple command languages and to present a more uniform "system" to the user) in such a way that a command language designer can override them if necessary. Examples of such builtin recognizers are file names, free text, words, and visibles (words with ajoining punctuation).] 3a4d1

5) What interaction and presentation forms should be provided? 3a5

Should function keyboard simulation or menu selection be provided for tools? 3a5a

[NOV 6-7 DECISION: 3a5b

Function keyboards and menu selection will not be supported in the first year system. Interaction forms will be similar to those now supported by NLS.] 3a5b1

6) What sorts of things should the user be allowed to tailor in the Frontend to his own personal preferences? 3a6

Should he be able to specify the level of verbosity, the amount of prompting, the succinctness of error messages, the recognition algorithm for command word recognition, tools available to him, etc. what should a site or group manager be able to specify on behalf of his people? 3a6a

[NOV 6-7 DECISION: 3a6b

The facilities now supported by NLS for this purpose will be provided. In addition, the list of available tools will be settable by a project leader for his project personnel. A User Profile tool will be provided as a subset of the NLS backend.] 3a6b1

7) How much control should a tool be allowed to have over the user's terminal? 3a7

Our current plan is to provide a standard mechanism for presenting status and error messages and the ability to write text (and perhaps graphics) in or to subdivide windows on a portion of the screen. Is anything else needed or does anything beyond this constitute a violation of the primary reason for the frontends existence -- i.e. to present a standard interface to all NSW tools? 3a7a

[NOV 6-7 DECISION: 3a7b

It was not felt that additional facilities would be needed for the first year system.] 3a7b1

8) Semantic help 3a8

It is envisioned that the user will be able to obtain English help with tools in the NSW. This will be accomplished by providing a separate tool, capable of interacting with the user (via a grammar in the Frontend) and using a structured data base provided along with the tool grammar. This help tool will not run in the satellite machine but will be invoked by the satellite whenever the user asks for semantic help with a tool. The help tool will be provided with the name of the help data base for the tool the user was using and a representation of the user's command state at the time he requested help. (Once a connection has been established to the help tool for a user, the connection will probably be maintained until the user terminates the session.) It is expected that the command language designers will provide the data bases. It is expected that there will be one data base for the NSW as a whole, describing global concepts, organization, and purpose of the NSW. This data base will be available at all times to the user. In addition, we may wish to produce a data base that is a high-level guide to all the tools accessable through NSW. 3a8a

We would propose that for first-year NSW, this help tool is simply a set of calls on the NLS backend, with the data bases being NLS structured files (this approach is now being used within NLS). 3a8a1

In the future it is expected that the help tool will be able to take on the character of a tutor and show the user how to execute commands and what the effects of doing so are as well as providing the user an environment in which he is

free to try things without running the risk of destroying anything or leaving unwanted trash around. (We refer you to the NLS-SCHOLAR project at BBN and to the COTCO system being designed at ISI as examples of more active help facilities.) Initially and for some time, however, we expect the help tool to be more of a browsing aid, using a structured data base to allow the user to more quickly find the information he seeks. 3a8b

[NOV 6-7 DECISION: 3a8c

This was felt to be an acceptable plan for first year NSW.] 3a8c1

9) UNDO and REDO 3a9

What mechanisms must be in the Frontend to facilitate UNDO and REDO capabilities on an NSW-wide basis and what must be done for tools which can undo the effects of previous function executions. 3a9a

[NOV 6-7 DECISION: 3a9b

UNDO was not discussed and will not be available in first year NSW. REDO was discussed briefly and was felt to be valuable. If resources permit, the frontend will provide this for the most recently specified commands for first year NSW.] 3a9b1

10) Checkpointing 3a10

Must the Frontend do anything with respect to NSW checkpointing, backing up to a checkpoint, etc? What does the Frontend do when the Works Manager crashes and restarts? What must the Frontend do to allow the user to pickup where he left off at the end of his last session? 3a10a

[NOV 6-7 DECISION: 3a10b

This was not discussed (we ran out of time). Perhaps Warshall or Millstein could write a short blurb on this? Is it to be available in first year NSW? Suspending and resuming a session will not be provided by the Works Manager for the first year system.] 3a10b1

11) Terminal Linking 3a11

The ability for two or more users to connect their terminals (especially displays) together in order to work together or

simply to type to each other has been found to be a very valuable facility. Should this be offered in the NSW. 3a11a

[NOV 6-7 DECISION 3a11b

Terminal linking, although a very valuable facility, will not be supported in first year NSW.] 3a11b1

THE CML AND ITS IMPLICATIONS FOR NSW TOOLS, TOOL BUILDER, TOOL INSTALLERS, AND COMMAND LANGUAGE DESIGNERS 4

The Frontend system that is being planned for the NSW consists of the following: 4a

1) A formal language (CML) for specifying NSW user interfaces 4a1

2) A compiler for that formal language that runs under TENEX as a subsystem or from NLS 4a2

3) Tool grammars, products of the CML compiler or any other such program 4a3

4) A CML interpreter that processes a CML grammar in order to work with the user in specifying syntactically correct commands to the NSW. 4a4

5) A user profile data base that is used by the CML interpreter while interacting with the user. This data base allows the Frontend to be tailored to the individual preferences of the users. 4a5

6) A user statistics data base, where, if desired, statistics can be accumulated on commands used by a user, error rates, etc. This will be accumulated on a file or perhaps reported to the Works Manager. 4a6

7) Access to a semantic help tool which is employed by the Frontend when the user requests semantic level help with a tool or a command. It is presumed that each tool, in addition to supplying the Frontend with a grammar will also supply it with the name of a help data base file whose structure and content, as with the grammar, are the sole responsibility of the tool builder/supplier. 4a7

This help tool could also be kept informed of the user's dialog with the Frontend, can have access to the tool grammar, the current parse state of the user, and the user's profile. 4a7a

Detailed discussions of the CML and the CML interpreter are being prepared and will be forwarded to you as soon as they are completed. 4b

DOCUMENTATION THAT SEEMS NECESSARY FOR THE NSW TO FUNCTION [Someone (Carlson, Balzer, or Crocker) should specify such a list of necessary documentation and affix responsibility for providing it.]: 5

A system guide to installing and running an NSW Frontend 5a

System specs for tool bearing host execs 5b

A system manual on the flow of control in the NSW as a whole 5c

A system guide for Command language designers 5d

A system guide for tool installers 5e

System documentation on the Procedure Call Protocol 5f

System documentation on debugging NSW tools 5g

System documentation on (PCP) external procedures for each tool 5h

System documentation on CML 5i

System documentation on functions the Frontend provides to a tool 5j

System documentation on functions the Works Manager provides 5k

CHI 30-DEC-74 11:36 24853

NSW Frontend Issues -- reflects Decisions of Nov NSW Review Meeting

(J24853) 30-DEC-74 11:36;;;; Title: Author(s): Charles H. Irby/CHI;
Distribution: /CHI([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk:
CHI; Origin: < NSW-SOURCES, FE-ISSUES.NLS;4, >, 13-NOV-74 07:59
CHI ;;;;####;

JMB 30-DEC-74 12:23 24854

Response to NSW Documentation thing (24848,>

I read <gjournal,24848,> --NSW Documentation draft--but have no comments to offer right now. If I think of anything I'll let you know. Pray tell, what is DPCS????? 1

JMB 30-DEC-74 12:23 24854

Response to NSW Documentation thing (24848,>

(J24854) 30-DEC-74 12:23;;;; Title: Author(s): Jeanne M. Beck/JMB; Distribution: /DVN([ACTION]) ; Sub-Collections: SRI-ARC; Clerk: JMB;

Procedure Call Protocol Documents

Table of Contents

* These documents are added to this collection in this printing. The other documents are reprinted without change.

PCP Table of Contents

(J24855) 30-DEC-74 15:21;;;; Title: Author(s): Jonathan B. Postel/JBP; Distribution: /JBP([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JBP; Origin: < POSTEL, PCPTOC.NLS;1, >, 30-DEC-74 14:54 JBP ;;;;####;

National Software Works PCP Documents

Table of Contents

* These documents are added to this collection in this printing. The other documents are reprinted without change.

NSW Table of Contents

(J24856) 30-DEC-74 15:22;;;; Title: Author(s): Jonathan B.
Postel/JBP; Distribution: /JBP([INFO-ONLY]) ; Sub-Collections:
SRI-ARC; Clerk: JBP; Origin: < POSTEL, NSWTOC.NLS;1, >,
30-DEC-74 15:06 JBP ;;;;####;

KIRK 30-DEC-74 15:35 24857

Response to > ROUGH DRAFT NSW Documentation Work Breakdown and Time Allocation <24848,>

Comments in response to ROUGH DRAFT NSW Documentation Work Breakdown and Time Allocation > <24848,>. First the bad news, then the good news followed by some recommendations. 1

Primers, scenarios, and "discursive introductions" for sending a letter, reading mail, producing a document [5 or so primers in this], and DEX, should be included in documentation for "inexperienced users". 2

I should have 9 weeks to spend on the NSW documentation (counting vacation), not 5 as listed. 3

It seems to me that if a multi-subsystem nls is to be a single "tool" in the nsw, those subsystems could be a part of one big file. It is clear that additional code and conventions for a multi-file help system will be necessary for subsystems or "tools" which are not an integral part of nls. A design for this should be started immediately so that it can be planned for in the help documentation. There are several questions that need to be answered in that respect. 4

With respect to the time element, it looks like the design documents for each part of the NSW should be agreed upon by everyone including documenters and be complete enough to document from (or preferably BE the documentation). In addition, FOOLPROOF procedures need to be worked out to ensure that documenters are not just informed of, but have a say in any deviations from the design. Otherwise, inconsistencies between the code and the documentation must be considered bugs in the code. 5

KIRK 30-DEC-74 15:35 24857

Response to > ROUGH DRAFT NSW Documentation Work Breakdown and Time Allocation <24848,>

(J24857) 30-DEC-74 15:35;;;; Title: Author(s): Kirk E. Kelley/KIRK; Distribution: /JOAN([ACTION] dirt) DVN([INFO-ONLY]) POOH([INFO-ONLY]) JMB([INFO-ONLY]) EKM([INFO-ONLY]) CHI([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: KIRK;

POOH 30-DEC-74 16:25 24858

This is a test of insert sendmailform

this is a test of the glossary

POOH 30-DEC-74 16:25 24858

This is a test of insert sendmailform

```
EXE MES MOV MES JOURNALD                                        1

EXE PRO DEL LAS                                                 2

TITLE: This is a test of insert sendmailform
COMMENT: this is a test of the glossary
AUTHOR(S):POOH
NUMBER:24858

DISTRIBUTE FOR ACTION TO: POOH
DISTRIBUTE FOR INFO-ONLY TO: POOH
SUBCOLLECTION(S):
KEYWORD(S):
HANDLING INSTRUCTION:
RECORDING INSTRUCTION:
OFFLINE ITEM -- LOCATED AT:
RFC NUMBER:
OBSOLETES ITEM NUMBER(S):
ACCESS STATUS:
UPDATE TO ITEM NUMBER(S):
INSERT LINK TO FOLLOW:
FORWARD ITEM NUMBER:
MESSAGE: none
BRANCH AT:
PLEX AT:
GROUP AT:
FILE:
SEND THE MAIL,                                                  3
```

POOH 30-DEC-74 16:25 24858

This is a test of insert sendmailform

(J24858
) 30-DEC-74 16:25;;;; Title: Author(s): Ann Weinberg/POOH;
Distribution: /POOH([ACTION]) POOH([INFO-ONLY]) ;
Sub-Collections: SRI-ARC; Clerk: POOH;

KIRK 30-DEC-74 16:29 24859

Some reasons for having <:wg> and using angle brackets in general in the proposed new sendmail 'envelope' format

I think placing <:wg> before the link to a message that has been delivered to a user is an excellent idea to save user and cpu time, while at the same time allowing a single process (jump to link) for seeing all sendmail items. 1

In general, I think the use of angle-brackets consistant with the rest of NLS (file commands, origin statement, insert link command, etc.) should be encouraged instead of parentheses for the following reasons (copied from help). 2

problems with using parens ():
Since parentheses are legal characters in a filename, placing two links delimited by parens next to each other causes an irresolvable confusion. Also, it is possible for parenthetical clauses in a statement to become confused with links. If you are using the Link entity while editing a link that does not have proper syntax, it is possible to end up editing the parenthetical clause instead. To avoid confusion, we suggest you use angle-brackets <> instead of parentheses. 3

KIRK 30-DEC-74 16:29 24859

Some reasons for having <:wg> and using angle brackets in general in the proposed new sendmail 'envelope' format

(J24859) 30-DEC-74 16:29;;;; Title: Author(s): Kirk E. Kelley/KIRK;
Distribution: /RLL([INFO-ONLY]) FDBK([INFO-ONLY]) ;
Sub-Collections: SRI-ARC FDBK; Clerk: KIRK;

DVN 30-DEC-74 21:01 24860

CML Paper and Command Summary go to COM

On Saturday the 28th I put off on tape 121 at ISI a corrected COM version of Ken's CML paper. Note this version will not have statement numbers because of an error on my part in confusing SNFFontshow with SNFontshow. On the same tape I put a COM version of the cuurrent command sumary. That file name is commands. 1

DVN 30-DEC-74 21:01 24860

CML Paper and Command Summary go to COM

(J24860) 30-DEC-74 21:01;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /JOAN([ACTION] dpcs notebook please) KEV([INFO-ONLY]) JMB([INFO-ONLY]) ; Sub-Collections: SRI-ARC DPCS; Clerk: DVN;

DVN 30-DEC-74 21:37 24861

More on Current Marker

Follows journal items (24815,) (24802,) and (24821,)

DVN 30-DEC-74 21:37 24861

More on Current Marker

In Help generally we have used the term Current Marker and the acronym CM rather than any of the terms suggested in the cited journal items. In particular we have avoided the word "pointer" which is used to refer to a text point in L-10 which points between rather than to characters. I believe Kirk's "command marker" was a slip of the mind. However, we have also been trying to avoid the use of any such esoteric term and to use instead constructions such as "moves you to" or "you are at", see for example Help's account of "address" or effects under "delete group". 1

DVN 30-DEC-74 21:37 24861

More on Current Marker

(J24861) 30-DEC-74 21:37;;;; Title: Author(s): Dirk H. Van Nouhuys/DVN; Distribution: /JOAN([ACTION] dirt notebook please) DIRT([INFO-ONLY]) ; Sub-Collections: SRI-ARC DIRT; Clerk: DVN;

PI-Write-up, Abbott

RISOS PROJECT
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Robert P. Abbott
Lawrence Livermore Laboratory
Box 808 L-307
Livermore, California 94550

The program's initial intent was to provide a group which is capable of examining any operating system (O/S) for integrity flaws. The group has evolved into what is better described as an audit team in that a given problem in an O/S may or may not be regarded as a security breach depending on the environment of the computer installation itself. As such, it is necessary to identify problems in general and leave the security determination up to the installation.

The underlying philosophy of the project recognizes the similarities in design among the various O/S. It follows that integrity flaws must have similarities which are applicable across the various systems.

The work procedes along three major lines:

1. A set of programs has been constructed which aid in the examination of O/S. The tools are semi-automatic requiring programmer interaction. The tools operate on a data base which is constructed and maintained for each O/S. The data base elements are:

   a. The output edit from the assembly of the O/S.

   b. The Parsed Data Structure (PDS) is a fixed assembly format in which all O/S edits are placed.

   c. The Total Source Listing (TSL) is idential to the assembly edit with the addition of the sequence numbers from the PDS.

   d. The Master Cross Reference (MCR) is an inverted file of all symbols in an O/S.

The tools themselves consist of:

a. The Source Program Alteration Module searches a module of PDS or
ISL according to a Boolean list containing opcodes, operands,
and labels. User comments may be inserted at each match point. 8a

b. A Statistical Analysis program operates on the modules in the
Parsed Data Structure. 8b

c. A Cross Reference Interface and Search Program operates on MCR
according to a Boolean string encompassing multiple instruction
lines. 8c

d. The External Reference Program produces a listing of all
external references. 8d

e. Compare - capable of comparing two system releases identifying
diffences, additions, deletions, etc. 8e

2. A modeling effort to produce a graph model of a given O/S in which nodes represent parts of that system - procedures or data structures - and edges show either synchronizing operations between shared components or changes in the use of capability controlled resources. Flaw suspects are to be uncovered at those places where resources are unexpectedly acquired or released, or where the synchronization of shared information is inadequate. 9

3. The development of a TENEX "exerciser" program consisting of a set of programs designed to drive the TENEX monitor throught all possible exits of conditional instructions and to note any fault situations that occur. 10

The tools have been applied to TENEX; GECOS-3; EXEC-8; IBM - VM, OS/MVT. The data base includes mulipie copies of some of these systems. It is currently around three times ten to the ninth bits. All programs and the data base are on the LLL Octopus system, but plans are under way to transfer the tools to our PDP-11 system for greater utility to ARPA and DOD. 11

A taxonomy of generic error classifications is being consructed. The taxonomy is based on our experiences with the various O/S and reflects an analysis of the RISOS file of reported and confirmed errors found in each manufacturer's product. 12

JAKE 20-MAR-75 13:59 24916

PI-Write-up, Abbott

(J24916) 20-MAR-75 13:59;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, ABBOTT.NLS;2, >, 23-FEB-75 15:46 JAKE ;;;; ####;

LLL-RISOS

NIC 24916
Part of NIC 24980

RISOS PROJECT
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Robert P. Abbott
Lawrence Livermore Laboratory
Box 808 L-307
Livermore, California 94550

The program's initial intent was to provide a group which is capable of examining any operating system (O/S) for integrity flaws. The group has evolved into what is better described as an audit team in that a given problem in an O/S may or may not be regarded as a security breach depending on the environment of the computer installation itself. As such, it is necessary to identify problems in general and leave the security determination up to the installation.

The underlying philosophy of the project recognizes the similarities in design among the various O/S. It follows that integrity flaws must have similarities which are applicable across the various systems.

The work procedes along three major lines:

1. A set of programs has been constructed which aid in the examination of O/S. The tools are semi-automatic requiring programmer interaction. The tools operate on a data base which is constructed and maintained for each O/S. The data base elements are:

   a. The output edit from the assembly of the O/S.

   b. The Parsed Data Structure (PDS) is a fixed assembly format in which all O/S edits are placed.

   c. The Total Source Listing (TSL) is idential to the assembly edit with the addition of the sequence numbers from the PDS.

   d. The Master Cross Reference (MCR) is an inverted file of all symbols in an O/S.

The tools themselves consist of:

   a. The Source Program Alteration Module searches a module of PDS or TSL according to a Boolean list containing opcodes, operands, and labels. User comments may be inserted at each match point.

b. A Statistical Analysis program operates on the modules in the Parsed Data Structure.

c. A Cross Reference Interface and Search Program operates on MCR according to a Boolean string encompassing multiple instruction lines.

d. The External Reference Program produces a listing of all external references.

e. Compare - capable of comparing two system releases identifying diffences, additions, deletions, etc.

2. A modeling effort to produce a graph model of a given O/S in which nodes represent parts of that system - procedures or data structures - and edges show either synchronizing operations between shared components or changes in the use of capability controlled resources. Flaw suspects are to be uncovered at those places where resources are unexpectedly acquired or released, or where the synchronization of shared information is inadequate.

3. The development of a TENEX "exerciser" program consisting of a set of programs designed to drive the TENEX monitor throught all possible exits of conditional instructions and to note any fault situations that occur.

The tools have been applied to TENEX; GECOS-3; EXEC-8; IBM - VM, OS/MVT. The data base includes muliple copies of some of these systems. It is currently around three times ten to the ninth bits. All programs and the data base are on the LLL Octopus system, but plans are under way to transfer the tools to our PDP-11 system for greater utility to ARPA and DOD.

A taxonomy of generic error classifications is being consructed. The taxonomy is based on our experiences with the various O/S and reflects an analysis of the RISOS file of reported and confirmed errors found in each manufacturer's product.

JAKE 20-MAR-75 14:09 24917

PI-Write-up, Abramson

ALOHA SYSTEM RESEARCH
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Norman Abramson
THE ALOHA System
University of Hawaii
Honolulu, Hawaii 96822

1. ALOHA REPEATERS - ALAN OKINAKA

THREE ALOHA REPEATERS HAVE BEEN SUCCESSFULY BUILT AND TESTED FOR USE IN THE ALOHANET. BY MEANS OF THE REPEATER THE REACH OF THE ALOHANET CAN BE EXTENDED BEYOND THE 50 TO 100 MILE RANGE OF A SINGLE RADIO TRANSMITTER, THE NETWORK CAN BE EXTENDED OVER OR AROUND OBSTACLES AND THE GEOGRAPHICAL COVERAGE OF THE SYSTEM CAN BE SHAPED TO CONFORM TO OTHER REQUIREMENTS. THE REPEATERS HAVE MADE POSSIBLE THE STUDY OF TRAFFIC WHICH ACCESSES THE MENEHUNE THRU SINGLE AND MULTIPLE REPEATER HOPS (SEE ITEM 3). SINCE THE REPEATERS BUILT SO FAR DO NOT INCLUDE PROGRAMMABLE UNITS, WORK ON PROVIDING THIS CAPABILITY IS CONTINUING. AS SOON AS PROGRAMMABLE CAPABILITY IS AVAILABLE, WE PLAN TO USE THE ALOHA REPEATERS IN SUPPORT OF OUR THEORETICAL WORK ON SPATIAL CHANNEL CAPACITY.

2. ALOHA CHANNEL PROTOCOL AND PCU'S - CHRISTOPHER HARRISON

DURING THE PAST YEAR THE FIRST ALOHA PROGRAMMABLE CONTROL UNITS (PCU'S) WERE COMPLETED AND PUT INTO SERVICE. THE PCU'S ARE BUILT AROUND AN INTEL 8080 MICROCOMPUTER CHIP AND PROVIDE IMPORTANT EXPERIMENTAL SUPPORT TO THE THEORETICAL STUDY OF PACKET BROADCASTING CHANNELS AND TO THE STUDY OF PROTOCOLS FOR SUCH CHANNELS (SEE ITEM 3). BY MEANS OF PCU'S WE ARE NOW ABLE TO INTEGRATE CHARACTER-BY-CHARACTER TRANSMISSION, VARIABLE LENGTH PACKETS AND FILE TRANSFERS WITHIN THE EXISTING ALOHA CHANNEL. AN UNEXPECTED BYPRODUCT OF THIS WORK IS THE DEMONSTRATION OF THE FLEXIBILITY OF PACKET BROADCASTING CHANNELS IN PERMITTING A WIDE VARIETY OF SYSTEM AND PROTOCOL CHANGES WITHIN AN ESTABLISHED SYSTEM WITHOUT REQUIRING ANY CHANGES IN THE OPERATION OF EXISTING USERS.

3. STATISTICS COLLECTION SYSTEM - RICHARD BINDER, MICHAEL FERGUSON

AN INITIAL STATISTICS COLLECTION SYSTEM WHICH MONITORS THE PERFORMANCE OF THE ALOHANET HAS BEEN PUT INTO OPERATION AND HAS PROVIDED THE FIRST SET OF STATISTICS AVAILABLE ON ALOHANET TRAFFIC.

THE SYSTEM MEASURES ALOHANET DOWNTIME, USER PACKETS ON THE RANDOM ACCESS CHANNEL, PACKET LENGTHS, INTERPACKET TIMES, THE NUMBER OF PACKET REPETITIONS AND OTHER QUANTITIES. THE SYSTEM NOW MONITORS ONLY THE RANDOM ACCESS CHANNEL BUT WORK ON UPGRADING THE EXISTING SYSTEM TO ALLOW STATISTICS COLLECTION ON THE BROADCAST CHANNEL FROM THE MENEHUNE TO THE USERS IS IN PROGRESS. THE STATISTICS COLLECTION SYSTEM HAS ALREADY PROVIDED DATA FOR GUIDANCE IN THE USE OF THE ALOHA SIMULATION FACILITY (ITEM 4). 10

4. ALOHA SIMULATION FACILITY - RICHARD BINDER, MICHAEL FERGUSON 11

A SIMULATION FACILITY FOR THE ANALYSIS OF PACKET BROADCASTING CHANNELS HAS BEEN COMPLETED. THE FACILITY CAN ACCOMODATE A VARIETY OF CHANNEL PROTOCOLS AND OF USER CHARACTERISITICS TO ALLOW THE STUDY OF PACKET BROADCASTING SYSTEMS. THE OUTPUT OF THE STATISTICS COLLECTION SYSTEM (ITEM 3) HAS BEEN USED AS A GUIDE IN THE SELECTION OF USER
CHARACTERISTICS FOR THE SIMULATION FACILITY. THE OUTPUT OF THE SIMULATION FACILITY IN TURN HAS BEEN USED TO SUGGEST NEW THEORETICAL RESULTS WHICH WILL BE REPORTED IN EARLY 1975. 12

5. SATELLITE PACKET BROADCASTING - DAVID WAX 13

THE FIRST USE OF A SATELLITE TRANSPONDER IN A PACKET BROADCASTING MODE AMONG MORE THAN TWO USERS WAS SUCCESSFULY DEMONSTRATED IN 1974. THE TRANSPONDER USED WAS THE ATS-1 VHF TRANSPONDER AND THE EARTH STATIONS IN THIS EXPERIMENT WERE NASA/ARC, THE UNIVERSITY OF ALASKA AND THE ALOHA SYSTEM. THE EARTH STATIONS EMPLOYED WERE SMALL VHF STATIONS SUPPLIED BY NASA; NEW FAST ACQUISITION ALOHA MODEMS WERE LATER EMPLOYED TO IMPROVE ERROR RATES THRU THE ATS-1 TRANSPONDER. IN ADDITION, A NEW TECHNIQUE WHICH ALLOWS THE TRANSMISSION OF SHORT DATA PACKETS AT THE SAME TIME AS A CONVENTIONAL VOICE SIGNAL ON A SINGLE VOICE CHANNEL HAS BEEN TESTED ON THE ATS-1 SATELLITE. 14

6. THEORY OF ALOHA DYNAMICS - THOMAS GAARDER 15

A MATHEMATICAL MODEL OF AN ALOHA CHANNEL WITH BLOCKING AND CARRIER SENSE HAS BEEN CONSTRUCTED TO DESCRIBE THE DYNAMIC BEHAVIOR OF THE CHANNEL. IN THIS MODEL NEW PACKETS AND BLOCKED PACKETS ARE EACH SENT AT TIMES WHICH FORM A POISSON POINT PROCESS. FROM THE SOLUTIONS OBTAINED IT HAS BEEN FOUND THAT THE THRUPUT MAY APPROACH 1/E, BUT IN THE SATURATED CONDITION THE AVERAGE DELAY IS PROPORTIONAL TO THE NUMBER OF USERS AND THE USERS ARE BLOCKED MOST OF THE TIME. BY REDUCING THRUPUT BELOW THE CHANNEL CAPACITY, THE AVERAGE DELAY AND THE AVERAGE NUMBER OF BLOCKED USERS CAN BE REDUCED. 16

7. VARIABLE LENGTH PACKETS - RICHARD BINDER, MICHAEL FERGUSON 17

THE BEHAVIOR OF AN ALOHA CHANNEL USING VARIABLE LENGTH PACKETS WITH A GEOMETRIC DISTRIBUTION OF PACKET LENGTHS HAS BEEN INVESTIGATED USING THE ALOHA SIMULATION FACILITY AND A MARKOV MODELLING TECHNIQUE. AVERAGE CHANNEL THRUPUT AND USER DELAY DISTRIBUTIONS WERE OBTAINED UNDER A VARIETY OF CONDITIONS. RESULTS SHOWING THE EFFECTS OF CHANNEL DATA RATE ON RANDOM ACCESS CHANNEL STABILITY WERE ALSO OBTAINED. 18

8. THEORY OF MULTIPLE USER CHANNELS - SHU LIN 19

WE HAVE OBTAINED NEW CODING RESULTS FOR TWO CHANNEL MODELS. THE FIRST CHANNEL MODEL IS REFERRED TO AS A NOISELESS MULTIPLE ACCESS BINARY ERASURE CHANNEL. IN THIS MODEL, IF THE TWO TRANSMITTED BITS FROM THE TWO USERS ARE BOTH ZEROS (ONES), A ZERO (ONE) IS TRANSMITTED OVER THE CHANNEL TO THE RECEIVER; IF THE TWO TRANSMITTED BITS ARE DIFFERENT, AN ERASED SYMBOL X IS TRANSMITTED TO THE RECEIVER. IN THE SECOND CHANNEL MODEL, NOISE IS INTRODUCED. FOR BOTH CHANNEL MODELS, THE INPUT TO THE DECODER IS A VECTOR WITH SYMBOLS FROM (0,1,X). THE DECODER PROCESSES THE RECEIVED VECTOR AND DECODES IT INTO TWO CODE WORDS,ONE FOR EACH OF THE DATA SINKS. 20

JAKE 20-MAR-75 14:09 24917

PI-Write-up, Abramson

(J24917) 20-MAR-75 14:09;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, ABRAMSON,NLS;3, >, 23-FEB-75 17:35 JAKE ;;;; ####;

PI-Write-up, Berstein

Interactive Systems Research
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Morton I. Bernstein
System Development Corporation
2500 Colorado Avenue
Santa Monica, California 90406

The goal of SDC's Interactive Systems Research (ISR) program is to provide new and improved technology required by computer system end users to interface with these systems in ways that are most natural to their needs, skills, backrounds, and disciplines. The ISR program presently consists of three projects: (1) Speech Understanding Research, (2) Lexical Data Archive, and (3) Common Information Structures. The following summarizes the activities of each of these projects during 1974.

1. SPEECH UNDERSTANDING RESEARCH - H.B. Ritea, J.A. Barnett

The objective of this project, being conducted jointly with SRI, is the development of a data management system that is controlled by a variety of users through a free form spoken English vocabulary of 1,000 words. The target data base contains information about the naval fleets of the U.S., U.K., and the U.S.S.R.

The voice controlled data management system (VDMS) incorporates a variety of techniques and sources of knowledge including: speech waveform parameterization techniques, acoustic feature extraction, acoustic-phonetic analysis, phonological processes, word-matching procedures, prosodics, semantics, and pragmatics.

During 1974, substantial progress was made in acoustic feature extraction, acoustic-phonetic analysis, and software support. Two acoustic feature extraction programs were developed that perform automatic formant frequency analysis and fundamental frequency extraction. An experiment was conducted on the acoustic correlates of styles of speech to attempt to determine whether or not there are significant differences between read speech and spontaneous speaking as used in testing speech understanding systems. Preliminary results indicate that there are significant differences in the formant structure of vowels in read and spontaneous speech.

PI-Write-up, Berstein

Continuing efforts in basic acoustic-phonetic research is providing algorithms that improve the overall accuracy of the system. Results of an experiment on vowels in a retroflexed environment have provided the basis for developing a vowel identification method based on the non-retroflexed vowel formant space. 9

A new programming language, called CRISP, has been designed specifically for implementing large complex systems, and in particular, SUR systems. CRISP will provide the ability to implement both numerical bottom-end processes and the linguistic top-end processes in a single language without loss of efficiency to either. The design specifications have been distributed to a wide audience for comment. Implementation has begun and CRISP will be operational on the IBM 370/145 under VM/370 in July 1975. 10

By the end of 1974, considerable progress had been made toward the implementation of the latest version of VDMS that combines SRI's linguistic components with the SDC portions to form a complete prototype system. This version has been designed to permit continuous, incremental expansion to the 1976 system. 11

2. LEXICAL DATA ARCHIVE - T.C. Diller 12

The Lexical Data Archive (LDA) project has as its primary objective the collection and dissemination of semantic information pertinent to the 3,000-odd words appearing in the SUR lexicons. LDA monitors a broad range of sources of lexical data, selects that data most relevant to the SUR tasks, formats the data for archiving purposes, and makes the data accessible to the SUR lexicon builders both by printout and by on-line access via the ARPA network. 13

At the end of 1974, LDA offered the SUR lexicon builder (a) about 300 of the best analyses of the semantics of words (from the linguistic literature), (b) explanations of about 200 semantic components used as elementary constituents in the semantic analyses, (c) summaries of the best analyses of 15 basic semantic notions (from the philosophic literature), (d) collocational information on about 1,000 SUR words (from Webster's Seventh), (e) contexts of occurrence for about 2,000 SUR words (from the Brown Corpus), and (f) bibliographic reference to nearly 3,000 documents relevant to speech understanding. 14

In addition to augmenting each of these files, LDA is currently preparing (g) a semantic field file that will show all English words connected to a given SUR word by a definitional, synonymitive, antonymitive, or morphological link, and at the request of the BBN SUR group, (h) a field intended to show definitional links between words in particular lexicons. 15

3. COMMON INFORMATION STRUCTURES - A. Shoshani 16

PI-Write-up, Berstein

This project addresses the problem of converting and transferring data bases among disparate data management systems. The approach is based on the concept that the data conversion process can depend basically on conversion at the logical level. At this level, conversion can be achieved by using the existing query and generate functions of the data management systems themselves to move the data from their physical representation to the logical level and back. 17

The conversion system has three principal components: (1) a source reformatter, (2) a translator from source to target format, and (3) a target reformatter. The reformating process does not involve any logical restructuring of data, but rather a one-to-one mapping of values. 18

The major achievement during 1974 was the development of the translator, which is the heart of the system. This translator accepts statements in a Logical Data Description Language for the source and target data bases and statements in a Logical Data Translation Language that describes mappings and associations between source and target data fields. Its design has been completed and one of its two main components, the Converter, has been implemented and tested. Implementation of the other component, the Analyzer, will be completed next year. 19

20

JAKE 20-MAR-75 14:53 24918

PI-Write-up, Berstein

(J24918) 20-MAR-75 14:53;;; Title: 3"; Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, BERNSTEIN.NLS;3, >, 23-FEB-75 19:11 JAKE ;;;; ####;

PI-Write-Up, Bryan

COMPUTER NETWORK AND USER DEVICE DEVELOPMENT
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Roland F. Bryan
Computer Systems Laboratory
University of California
Santa Barbara, California 93106

This summary covers research and development work carried out during calendar year 1974. Our research took place under the designations:

Research in Computer Network Development
Research in Micro-Terminal Development

RESEARCH IN COMPUTER NETWORK DEVELOPMENT

----NCP 360/370 (M. Krilanovich, R. Stoughton)----

Work on the UCSB Network Control Program was completed in August. The last additions were completion of billing routines to incorporate traffic charges to Network users both on and off site; to finalize RJS, Network File service and to complete documentation for continued maintenance. The NCP has been replicated at most of the 360 and 370 sites on the Network. Specifications for the addition of the Stanford WYLBUR System were completed. This was to be added for Network use as well.

----User Support Services, Hardware and Software (Berggreen/McAfee)----

Assistance was provided to other sites so that testing of Network program additions could take place. This work included Very Distant Host connection for CHI, Benchmark testing on services provided by various sites, development of Graphics Protocol Level 0 as a test front-end to the UCSB On Line System, export to 370 sites of 370/IMP controllers with NCP programming support, and consulting to other sites on techniques for Network attachment.

----Remote Conferencing by ARPANET (Pickens/Pfeifer)----

This work was to develop techniques for both voice and visual transmission on the packet-switched Network. The basic goal was to find ways to smooth the flow between conferencing sites by design of a data "Streaming" processor.

PI-Write-Up, Bryan

Equipment for the task was late in arriving which resulted in the substitution of a plasma terminal from the Illinois PLATO system as a test unit. This terminal requires the continuous flow of display information to produce dynamic CAI presentations. Hardware and software additions were made to Network interfaces at UCSB and at Illinois ANTS. Protocol modifications were implemented to allow this equipment to operate on the Network, and the flow of information was monitored in order to assess buffer size and the effect of packet switching upon the display flow. This work is continuing under a separate contract from ARPA-HRRO and includes the fabrication of a number of special processors for operation of plasma terminals in Network conferencing. Methods for delta modulation were explored and algorithms written for voice communication between the UCSB Sel-810B signal processor and alternate Network sites. 12

RESEARCH IN MICRO-TERMINAL DEVELOPMENT 13

----Micro-Terminal Development
(Berggreen/Wells/Danielson/McAfee)----- 14

The CSL design team is to seek out the latest developments in miniature display, keyboard, storage, and micro-processors so as to incorporate these components in the implementation of a highly portable computer terminal for individual use. Where devices do not exist to accomplish the task, the Laboratory is to direct and support research in the designated area. The proposed terminal will weigh less than 10 pounds, will have 8,000 characters of storage and will equal certain minicomputers in performance. The terminal is to provide the user with enough processing power to handle text editing and message creation in a manner similar to ARPANET techniques. A small tape cassette will be added for auxiliary storage. During the first four months of the project the prototype terminal was specified and processor, display, keyboard, storage, and power source were ordered. The assembly of the prototype should begin in March. 15

----End to End Error Recovery (Proctor/Pickens)-------- 16

This task involves the investigation of the requirements and the application of end-to-end error checking/re-transmission equipment for use on the Network. Implementation of a pair of programmable units is planned. These devices will to insure terminal-to-terminal integrity of the data being transmitted. Consideration is also being given to a physical means for detection and correction of errors produced by the keyboard of the sending terminal and the printing mechanism of the receiving terminal. 17

PI-Write-Up, Bryan

(J24919) 20-MAR-75 15:01;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, BRYAN.NLS;4, >, 25-FEB-75 22:10 JAKE ;;;; ####;

JAKE 20-MAR-75 15:08 24920

PI-Write-up, Cerf

Internetworking Research
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Vinton G. Cerf
ERL 228B
Stanford University
Stanford, California 94305

This summary covers work performed during fiscal year 1975 (1 July 1974 through 30 June 1975).

1. Internetwork Protocol Design

A detailed specification for an internetwork protocol was written and implementations of the protocol were started in January, 1975. The protocol permits host computers on different networks to communicate through one or more intermediate networks using a logical GATEWAY facility which connects the networks together. The GATEWAY appears to be a host from the point of view of the network communications nodes (e.g. IMPs in the ARPANET), however, the GATEWAY need not be a real host machine but might be implemented in software in communication nodes adjacent to each other in two different nets.

The internetwork protocol is realized by a Transmission Control Program (TCP) resident in each network host which wants to communicate with hosts in other networks. The TCP views all the intervening networks as communication facilities which tie the GATEWAYs together. To allow for wide variations in actual network facilities, the TCP assumes only that there is a finite lifetime during which an internetwork packet can be delayed in the intermediate networks. After the lifetime, T, expires, either the packet has been delivered or it has been lost. An internetwork packet arriving at a GATEWAY may be split into smaller pieces for transmission through an intermediate network, but need not be put back together until it reaches the destination TCP. This permits alternate routing to different GATEWAYs connected to the same intermediate network. No assumptions about the order of packet arrival at the destination are made, so the TCP can put things back in order if they have been permuted by the transit across the networks.

PI-Write-up, Cerf

An end-to-end retransmission scheme and duplicate detection scheme have been designed which can screen duplicates of packets which may be the result of earlier incarnations of a given connection. Hosts which crash and lose all TCP related information can start up again without injecting potentially confusing duplicate packets into the net using an initial sequence numbering scheme proposed by R. Tomlinson of BBN for this purpose. A non-incremental, end-to-end flow control scheme has also been implemented which is derived from the notion of a "window" of sequence numbers the receving TCP permits the sending TCP to emit. Loss of flow control information is not a cause for deadlock since such information can safely be retransmitted without confusing the receiver even if duplicates should arrive. 5c

Cooperating in this effort are BBN (J. Burchfiel, R. Tomlinson, W. Plummer, and E. Mader) and University College London (P. Kirstein, D. Lloyd, P. Higginson, M. Galland). At Stanford, the working party includes Y. Dalal, R. Karp, C. Sunshine. 5d

2. Internetwork Experiments 6

A series of experiments using GATEWAYs and TCP's at the three sites listed in (1) above has been scheduled for second quarter 1975. A number of design issues can only be resolved through measurement (e.g. timeouts, delays, algorithms for setting window sizes, relative "out of order" of packet arrival, retransmission frequency, and buffer allocation strategies. Furthermore, these experiments should reveal the statistics of throughput and delay that result from multiple GATEWAY transitions. To achieve this kind of experimenting, we will first simulate a number of nets using the ARPANET and the three sites (BBN, UCL, SU-DSL) as both TCP and GATEWAY. Through a simple trick (using different link numbers on ARPANET packets), it is possible to emulate multiple networks. Later we expect to try connecting ARPANET with EPSS to observe real internetworking in action. 6a

3. Hypothetical Studies 7

PI-Write-up, Cerf

Interconnection of ARPANET with terminal oriented networks can pose some very tricky problems (e.g. with terminal echoing and control) so we have been designing a hypothetical interface between ARPANET and TYMNET to uncover some of the fundamental issues. Two distinct approaches are being considered. One involves the use of a TCP in hosts on both nets and a GATEWAY of the sort imagined in section (1) above. The second approach is more along the lines proposed by the UCLondon team in which the GATEWAY performs an actual translation of protocols. This latter path has not had much success for us so far, largely owing the the number of restrictions on services one would have to live with to achieve anything -- e.g. line at a time service only. T. Wolpert is the active worker. 7a

4. International Standards 8

As chairman of IFIP working group 6.1, I have been active in the promotion of packet internetwork experimentation and protocol design. This effort has been going on since October 1972 and is beginning to bear fruit in the form of contributions to CCITT. 8a

5. ELF System 9

As users of the ELF system, we have implemented both a Simple Minded File System compatible with DOS files and also a basic Server FTP which can accept network mail, text files and binary files. At the time of this writing, the FTP is still in the checkout stage. D.Rubin and J. Mathis are responsible for this effort; and earlier, T. Haugland (Norwegian Defense Research Establishment) contributed a version of FTP during his visit with us. 9a

JAKE 20-MAR-75 15:08 24920

PI-Write-up, Cerf

(J24920) 20-MAR-75 15:08;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, CERF.NLS;5, >, 27-FEB-75 14:02 JAKE ;;;; ####;

JAKE 20-MAR-75 17:08 24922

PI-Write-up, Culler

Network Speech Compression
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Glen J. Culler
150A Aero Camino
Goleta, California 93017

The work undertaken by CHI may be summarized as threeseparate tasks, although they were directly coordinated andmutually supportive. These are:

1. Development of an interactive signal processing system of significant power.

2. Real time implementation of the Markel algorithm on the ARPANET.

3. Basic research on the application of discrete wave theory to speech signal processing.

In descriptive terms, the hardware of the signal processing system was completed and checked out during the early part of the year. The connection to the network in hardware and software came up in November. The real time implementation of network speech compression was demonstrated jointly with Lincoln Lab in December. The part of our program which is in basic research stands on the shoulders of the foregoing and got underway in December. It will be our primary activity for the remainder of our contract year (which ends September 1).

SUMMARY:

1. A System for Interactive Signal processing Hardware (R. Bjorkman, J. Vanderford, B. Lum)
Software (M. McCammon, D. Taylor, G. Ball)

A. This system has a VDH-connection to the UCSB-IMP. The hardware consists of a pair of micro-programmable processors, four user stations that are graphics consoles with function keyboards, 64K of 1/2 microsecond, 18-bit word memory, three 2314-type disk drives, and an analog subsystem currently being extended to control an array of 6 microphones in a sound tunnel.

B. The operating system provides a time-sharing facility with interactive features in the foreground and signal processing, library transfers and system processes in the background. Real time applications temporarily lock out all other processes when required. Each user station is cued by a lock out light when this occurs. 7b

C. The languages supporting the system consist of a micro-programming language, a macro-assembly language and a user interactive language tailored to the needs of signal processing research, and which includes a console programming facility. armed with this capability, we can select the nature of programming best related to the requirements of the computational tasks desired to be implemented. 7c

2. Real Time, Packet Speech - NSC Prototype
(M. McCammon, D. Taylor, G. Ball) 8

As a means to guarantee an early capability, the NSC-group selected the LPC algorithm defeloped by J. Markel of SCRL. A pair of sites, Lincoln Lab and CHI, were selected to carry out a prototype experiment for real time LPC experience. A simplified form of the Network Voice Protocol and a parcel format of 67 bits per 20 milliseconds speech frame were used. Aside from some hangups, mostly concerned with the way the UCSB-IMP treated our VDH connection, the whole thing went very well. Expected difficulties with network delays were validated and an effort to improve several aspects of the overall process have begun. These include: 8a

a. Improvement of the IMP-VDH buffering.
b. Further study of network delays by appropriate sites.
c. Better matching of LPC algorithms at different sites.
d. Improvement of quality of LPC analysis.
e. Investigation of type 3 message usage.
f. Development of better packet handling for network speech communication. 8a1

3. Discrete Wave Theory Applied to Speech Signal Processing
(G. Culler, J. McGill) 9

PI-Write-up, Culler

Consider a sound wave propagating from a small source into a domain bounded by walls with separations less than acoustical wave lengths associated with speech signals. If an array of microphones is placed inside this domain or imbedded in its walls, then the data collected by synchronously sampling the outputs of these microphones
comprises a discrete wave on multiple indices, one for time and the rest for space. If the array is linear and oriented outward from the source, then we have a discrete wave on two indices. The kick-off for a theory of discrete waves on two indices is the following theorem: 9a

> "The discrete Fourier transform with respect to a time index of finite extent satisfies one and (up to common factors) only one linear, second order difference equation with real coefficients." 9a1

In the cases of primary interest, these coefficients are a function of the geometry of the domain and thus, with a computer system providing synchronized D/A stimulation of the source and A/D sampling of the microphone array, we have a direct phenomenological means of deriving discrete difference equations of compound cavities. Research is under way to determine such equations for cavities of interest in speech generation and relate them to area functions as determined by the lpc technique. our hope is that through a better understanding of discrete wave equations, we may improve the relationship of speech quality to data rate in representation. 9b

JAKE 20-MAR-75 17:08 24922

PI-Write-up, Culler

(J24922) 20-MAR-75 17:08;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, CULLER,NLS;2, >, 23-FEB-75 00:54 JAKE ;;;; ####;

PI-write-up, Dertouzos

Internetworking Research
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Michael L. Dertouzos
Project MAC
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139
(617) 253-2145

In 1974 Project MAC continued work under ARPA support in the areas of Automatic Programming, Automation of Programming Technology,

ARPANET RELATED ACTIVITIES, AND THE MATHEMATICAL LABORATORY.

A. Automatic Programming

The principal goal of this work is to seek a long-term solution to the high-cost of software by automating the process of program generation from high-level descriptions of desired tasks. We are well underway in the planning and development of our prototype called PROTOSYSTEM-I which is an expert system in many aspects of the design problem to be solved. This system helps the human designer to express his solution in terms of methods already known to the system, and in a very high-level language. Using this information and knowledge of the environment in which the system being designed will run, PROTOSYSTEM-I finds an efficient way of implementing the specified system and produces the required software. PROTOSYSTEM-I also stands ready to explain the design and its behavior, for future maintenance purposes in a natural language familiar to the system maintainer.

In 1974, several experiments were done on a data base query system; modules for operations management were implemented; heuristic algorithms for designing IBM-370 PL/I files were tested; the number of data items generated in PROTOSYSTEM-I was studied; the OWL system for building expert problem-solving systems was redesigned and its implementation was begun. In addition, a conceptual framework was produced for automatically programming micro computers from a very high-level language.

B. Automation of Programming Technology

PI-Write-up, Dertouzos

The goal of this work, which is part of our Automatic Programming effort, is to increase programmer productivity through a system and associated tools that make possible the creation, debugging and documentation of programs. During 1974, a version of ABSTR, an automated abstractor, was made operational; a sophisticated message system was completed; CALICO the system with over 2,000 well-documented subroutines was completed and generalized to MUDDLE (higher-level language) subroutines. In addition, an initial TENEX MUDDLE was com- pleted along with several additional projects in graphics. With these various subsystems it is now possible to augment a programmer's ability to specify, retrieve, prepare, test, debug, validate, document and update programs in an effective way. 10

C. Arpanet Activities 11

During 1974, the ARPANET was successfully connected to the MULTICS development machine, thereby making possible through this second connection a variety of testing, modification and diagnostic procedures. In addition, several hardware and software interface related activities were carried out and buffering strategies were revised leading to 40,000 useful data bits per second, end-to-end file transfer rates via standard ARPANET protocols. 12

D. Mathematical Laboratory 13

This system, one of the most advanced operational expert programs continues to become progressively more important as a research tool for people who need expertise in symbolic manipulation. During 1974, work was completed on the use of the Hensel Lemma in algebraic manipulation; power series and poisson series packages were completed; new algorithms were developed for inverting matrices with symbolic entries; and new versions of the reference manual were produced. In addition, much time was devoted to user interaction. We are hopeful that in 1975, we will be able to offer this unique program to many more researchers through a consortium arrangement. 14

JAKE 20-MAR-75 17:14 24923

PI-Write-up, Dertouzos

(J24923) 20-MAR-75 17:14;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, DERTOUZOS.NLS;4, >, 3-MAR-75 22:43 JAKE ;;;; ####;

PI-Write-up, Feigenbaum

HEURISTIC PROGRAMMING PROJECT
1974 ARPA Project Summary

1

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

2

Prepared by: Edward A. Feigenbaum
Stanford University
Heuristic Programming Project
Department of Computer Science
Serra House
Stanford, California 94305

3

ACCOMPLISHMENTS OF THE PROJECT IN 1974:

4

1. Knowledge Acquisition by Theory Formation processes: The Meta-DENDRAL Program

5

Our objective has been to design and implement computer programs that use inductive inference to extract facts, regularities, and good heuristics in naturally occurring processes, so that reasoning programs can use this knowledge to assist human experts in solving complex real-world problems. Specifically, our objective has been to complete a program that automatically infers rules of analysis of instrument-data concerning the fragmentation of molecules in a mass spectrometer; and to put this program to work in the service of expert analytical chemists doing theory formation in mass spectrometry.

6

In 1974, significant progress was made on the three main parts of Meta-DENDRAL:

7

a) the Interpreation/Summarization program (INTSUM) for making first-level empirical generalizations about mass spectral processes from an underlying (fixed) conceptual base and a great deal of real-world data;
b) the Rule Generalization program (RULEGEN), which attempts to generalize the output of INTSUM into rules of broader applicability in mass spectral theory; and
c) the Rule Modification program (RULEMOD), which modifies the output of RULEGEN by retrospectively pruning and/or aggregating the rules, using a global evaluation over the full rule set.

7a

PI-Write-up, Feigenbaum

The INTSUM program is now essentially finished. The RULEGEN program has been written and debugged, and is undergoing "fine tuning" through application to real data, e.g., the families of androstanes and aliphatic amines. RULEGEN has now reached a point of completion where we can begin perturbing the model of mass spectroscopy to study the effect of such perturbations on the explanatory power of the program. The RULEMOD program is still under development. Its purpose is to modify and condense a set of rules in order to find a more economical way of explaining the data. The data gathered in response to the questions asked about the rules will determine, for example, whether those rules should be made more specific or more general. 8

The INTSUM and RULEGEN programs have been transferred to mass spectroscopists at Stanford for routine use as intelligent assistants to their theory-formation activities, and their use constitutes a normal part of the scientists' work. 9

2. Heuristic Computing in X-ray Crystallography: Another Application of AI to Scientific Hypothesis Formation 10

The general objective of this research is to apply problem solving techniques, which have emerged from artificial intelligence (AI) research, to the well known "phase problem" of x-ray crystallography, in order to determine the three-dimensional structures of proteins. 11

In 1974, our group, working with protein crystallographers at the University of California at San Diego, developed and implemented some of the basic analytical techniques (Patterson search, superposition, rotation function) which will comprise the performance modules of the overall system. 12

3. Knowledge Deployment Research: Inexact Reasoning 13

Often the knowledge acquired from Experts is intrinsically imprecise, i.e., although it accurately captures the Expert's best understanding of the situation, his understanding is imprecise. By what processes can a program be made to reason with such knowledge of a domain? 14

PI-Write-up, Feigenbaum

In a recent paper (to appear in the March, 1975 issue of Mathematical Biosciences) we examined the nature of such nonprobabilistic and unformalized reasoning processes, considered their relationship to formal probability theory, and proposed a model whereby such incomplete 'expertise' might be quantified. We have developed this model of inexact reasoning in response to the needs of AI Knowledge-based systems. That is, the goal has been to permit the opinion of experts to become more generally usable by programs and thus more available to nonexperts. The model is, in effect, an approximation to conditional probability. Although conceived with one problem area in mind, it is potentially applicable to any domain in which real world knowledge must be combined with expertise before an informed opinion can be obtained to explain observations or to suggest a course of action. 15

4. Knowledge Deployment Research for Real-World Applications: The Problem of Explaining a Program's Reasoning 16

As AI's research in knowledge-based systems moves toward application to real-world problems, it becomes essential for the intelligent agents so constructed to be able to explain to their users the knowledge and inference paths used in solving problems (or suggesting solutions). Without this ability, it is our belief that AI systems will not receive widespread acceptance. Nor will it be possible adequately to "debug" the knowledge in the systems' knowledge bases. 17

To conduct this research, we turned once more to the specific task domain and program (MYCIN) for diagnosis and treatment of infectious disease (an NIH-sponsored effort). 18

Recent progress in the development of the MYCIN system has included the development of a capability for providing sophisticated explanations of the program's reasoning steps and the strategy employed which resulted in those steps. 19

Several aspects of the implementation of MYCIN facilitate the accomplishment of this goal -- the modularity of the program's rules simplifies the task of maintaining a record of the program's chain of reasoning, which the use of an interpretive language like LISP makes feasible the examination by the program of its own knowledge base, as well as the translation of the rules into English for display to the user. This ability of the program to keep track of its reasoning and to examine its own knowledge and data is the central component in its ability to explain itself. 20

5. Application of AI Techniques to a Programmer's Task: Automatic Debugging 21

The work on Automatic Programming has been done in the context of a Ph.D. Thesis on Automatic Debugging of Logical Program Errors. The long-term goal of this research is to provide a system which will detect, diagnose, and treat logical programming errors. We make a distinction between three classes of errors: (a) Syntactic errors, (b) Semantic errors, and (c) Logical errors. A prototype system is now up which is capable of detecting a small but interesting class of InterLISP bugs, and diagnosing and treating a subset of the detected bugs. The prototype has correctly repaired a 'real' bug in a 'real' program. 22

6. Technology Transfer: to Medicine, Biology, Chemistry, Crystallography 23

Yet another objective has been the transfer of our science and technology to disciplines other than computer science, and to federal agencies other than DOD/ARPA. In 1974, we received funding from NIH for, and established, a national computer facility for Application of Artificial Intelligence to Medicine and Biology, tied to its user community by national computer networks (see below). We received NIH support for research on automatic theory formation; and received favorable review from NSF for extension of our research on heuristic computing in x-ray crystallography (the grant was officially received in February, 1975). 24

ESTABLISHMENT OF THE SUMEX-AIM FACILITY 25

The most significant instance of technology transfer, involving the transfer of concepts and techniques developed under ARPA support to another area of science and another source of support, occurred during the past year. The Co-principal Investigators of this project were successful in obtaining grant funds from the Biotechnology Resources Branch of NIH to establish a computing facility to satisfy not only the expanding needs of this project, the NIH-sponsored DENDRAL project, and the other NIH-sponsored 26

activity; but also the needs of an embryonic but rapidly growing national community of scientific work in the application of artificial intelligence techniques to Biology and Medicine. The facility (with staff), called SUMEX, has been established at the Stanford Medical School. Its complement of equipment is similar to that at the ARPA-sponsored AI laboratories -- the main frame is a PDP10I, operating under the TENEX operating system. It is currently connected to the TYMnet, and in the near future will be connected to the ARPAnet by a VDH connection. 27

SOME PROJECT PERSONNEL 28

The Co-principal Investigators are Edward A. Feigenbaum, Professor of Computer Science, and Joshua Lederberg, Professor of Genetics. Associate Investigator is Dr. Bruce Buchanan. Leading the Protein Structure Application project is Dr. Robert Engelmore, in collaboration with x-ray crystallographers at UC San Diego (Professor Joseph Kraut and Dr. Stephan Freer). Manager of the SUMEX Facility is Mr. Tom Rindfleisch. 29

SELECTED PUBLICATIONS IN 1974 30

[1] D. H. Smith, L. M. Masinter and N. S. Sridharan, "Heuristic DENDRAL: Analysis of Molecular Structure," Proceedings of the NATO/CNNA Advanced Study Institute on Computer Representation and Manipulation of Chemical Information (W. T. Wipke, S. Heller, R. Feldmann and E. Hyde, eds.) John Wiley and Sons, Inc., 1974. 31

[2] L. Masinter, N. S. Sridharan, R. Carhart and D. H. Smith, "Applications of Artificial Intelligence for Chemical Inference XII: Exhaustive Generation of Cyclic and Acyclic Isomers", Journal of the American Chemical Society, 96 (1974), 7702. (Also Stanford Artificial Intelligence project Memo No. 216.) 32

[3] B. G. Buchanan, "Scientific Theory Formation by Computer." To appear in Proceedings of NATO Advanced Study Institute on Computer Oriented Learning Processes, 1974, Bonas, France. 33

[4] E. A. Feigenbaum, "Computer Applications: Introductory Remarks," in Proceedings of Federation of American Scoieties for Experimental Biology, Vol. 33, No. 12 (Dec., 1974) 2331-2332. 34

[5] E. H. Shortliffe and B. G. Buchanan, "A Model of Inexact Reasoning in Medicine, July, 1974. To appear in Mathematical Biosciences, March, 1975. 35

JAKE 20-MAR-75 17:28 24925

PI-Write-up, Feigenbaum

(J24925) 20-MAR-75 17:28;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, FEIGENBAUM,NLS;3, >, 27-FEB-75 17:06 JAKE ;;;; ####;

JAKE 20-MAR-75 17:31 24926

PI-Write-up, Fralick

PACKET RADIO CHANNEL MEASUREMENT AND CHARACTERIZATION
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Stanley C. Fralick
Stanford Research Institute
Menlo Park, California 94025

One of the long range objectives of the Packet Radio project is to prove that packet-switched radio networks are technologically feasible and viable in a military context. SRI has undertaken the task of coordinating and integrating an experimental packet radio network being developed by a group of ARPA IPTO contractors during CY '74 and '75. The network will be used to demonstrate feasibility and to support application-oriented measurements. During CY '74 our activities have included:

1. Mobile Instrumented Packet Radio Repeater.

During CY '74 a flexible and highly instrumented mobile van facility was developed. In addition to an experimental packet radio spread-spectrum receiver, the van is equipped with considerable RF instrumentation and a minicomputer-based data acquisition system. This van was used to obtain propagation and noise measurements and will soon be used as an instrumented repeater for RF-link measurements and network testing. This facility is described in SRI Packet Radio Note (PRN)3, "A Measurement Program for Packet Radio Channel Characterization", by D. L. Nielson and R. A. Shepherd.

2. Packet Radio Propagation and Noise Measurements.

Introducing a wide-band digital radio communications system such as packet radio into the urban and suburban environment requires a knowledge of the limitations imposed by that environment on a communications signal. We collected and analyzed sufficient propagation measurement data to quantify, with high resolution and over a wide range of urbanization, distortion caused by multiple transmitter-receiver paths. We described the impulse distortion, the time variability, and the spatial coherence of the signal at street level--the latter being of importance to space-diversity reception. The noise inherent to the urban and suburban environment was found to be almost totally due to automobile ignition systems.

We made the first amplitude and time characterization of that noise in the frequency range above 1000 MHz. The results of these measurements are documented in SRI PRN 4, "Microwave Propagation and Noise Measurements for Mobile Digital Radio Application", by D. L. Nielson. 9

3. Packet Radio Link Analysis. 10

The results of the measurements have been used to support packet radio network RF link design. SRI consulted with and provided data to Collins Radio Group of Rockwell International to establish a preliminary design that will be tested during CY '75. We analyzed and compared modulation schemes and random-access methods to determine those that are feasible and to predict the effect of the radio environment on performance. This analysis was required to arrive at a design for the experimental network; however, it has broader implications to a large variety of military and commercial digital communications networks since it compares a variety of modulation schemes and network access modes in a realistic environment. These results are documented in SRI PRN 1, "RF Channel Capacity Considerations" by S. C. Fralick, SRI PRN 2, "Study of Throughput and Delay of Spread Spectrum Multiple Access Modes", by S. C. Fralick and J. K. Leung, and SRI PRN 7, "Technological Considerations for Packet Radio Networks" by S. C. Fralick and J. C. Garrett (Garrett is with the Collins Radio Group). 11

4. Packet Radio Traffic Sources and Terminals. 12

To provide controlled sources to generate required time- and geographic-traffic patterns for planned network measurements, we developed a microprocessor-controlled portable (suitcase-size, 30 lb.) traffic source. The traffic source has a full ASCII keyboard, 80 characters of display, and a 20 char/line printer so that it can double as a terminal. The microprocessor is programmed to format messages into packets and to support minor text editing tasks. The microprocessor will provide a simple but powerful tool to generate packets with a wide (and software changeable) variety of lengths, formats, etc. This work is documented in "The Role of Microprocessors in High Speed Portable Data Communications Terminals" by S. C. Fralick and D. Brandin, Prceedings of Journees d' Electronique, Lausanne, SW, 1974, and SRI PRN 6, "Digital Terminals for Packet Broadcasting by S. C. Fralick, D. H. Brandin (both of SRI) and F. F. Kuo, and C. Harrison (both of the ALOHA system, University of Hawaii). 13

JAKE 20-MAR-75 17:31 24926

PI-Write-up, Fralick

(J24926) 20-MAR-75 17:31;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, FRALICK,NLS;3, >, 23-FEB-75 19:16 JAKE ;;;; ####;

PI-Write-up, Frank

THE PRACTICAL IMPACT OF RECENT COMPUTER ADVANCES ON THE ANALYSIS AND DESIGN OF LARGE SCALE NETWORKS

1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Howard Frank
Network Analysis Corporation
Beechwood, Old Tappan Road
Glen Cove, New York 11542
(516) 671-9580

This summary describes Network Analysis Corporation's accomplishments in the study of local, regional and large scale data communications problems.

PACKET RADIO SYSTEM NETWORK STUDIES:

Efforts during the past contract year were aimed at establishing base performance characteristics of a single station, fixed repeater location Packet Radio System and to evaluate the effects on performance of a number of fundamental hardware design decisions. Analytic and simulation studies of throughput and delay were conducted to enable various design decisions including: use of multiple or single detectors at repeaters and stations, evaluation of tradeoffs between range, power and interference, incorporation of single or dual data rate repeaters, common versus split channel operation, and the use of omni versus directional antennas. In addition, numerous studies were performed to quantify system delay, throughput and blocking under various routing alternatives, acknowledgement schemes, repeater network organization and to insure that gross system performance using unoptimized operating parameters and algorithms was within a level that would justify further design efforts.

PACKET RADIO SYSTEM NETWORK ALGORITHMS AND CONTROL:

During the year, the main effort has been towards developing workable network algorithms, to insure order of magnitude performance and design robustness for a single station multiple repeater, multiple terminal network. Preliminary designs of eleven routing algorithms were evaluated using combinatorial analysis, three were selected for detailed design, and two of these were simulated and tested, and based on these tests, recommended for implementation. Single station, multirepeater initialization, network mapping, and transmission algorithms were proposed.

PI-Write-up, Frank

Hop-by-hop and end-to-end acknowledgement schemes were developed, simulated and tested. Simple terminal search and local terminal control algorithms were developed and simulated. In addition, a repeater location optimization algorithm was developed, programmed, and tested. The above family of algorithms provided a basis for demonstrating the reliable transmission of packets within the Packet Radio System, but further work is required to improve efficiency, to handle multiple stations, and to increase the number of types of terminals that can be handled by the system. 9

LOCAL AND REGIONAL DATA NETWORK PERFORMANCE, COST COMPARISONS AND ALTERNATIVES: 10

A variety of tools have been developed to allow economical cost/performance tradeoff studies. Accomplishments include: the practical demonstration that low cost terminal access can be achieved by hardware multiplexing at TIPs; the proof of the cost-effectiveness of multipoint lines for connecting low and medium speed terminals into ARPANET; the demonstration of the use of software demultiplexing as a means of increasing the terminal handling capacity of a TIP by a factor of 10; and the theoretical calculation of capacity, error rates and delay to establish of feasibility incorporating broadcast packet radio techniques on a wideband coaxial cable local distribution network to serve a large group of densely located military users. 11

INTEGRATED LARGE SCALE PACKET-SWITCHED NETWORK COST AND PERFORMANCE: 12

During the year, the groundwork was laid to complete the study of cost and performance tradeoffs in large scale packet-switched networks. Basic analysis and design algorithms for optimization of terminal processor location, topological optimization, throughput and delay analysis, and reliability analysis were completed. In addition, a number of cost/performance studies were completed. These include the proof of the cost effectiveness of using satellites to increase the capacity of ARPANET, the establishment of the feasibility of a 1,000 IMP packet-switched network using terrestrial links, and studies of the cost/effectiveness of packet-switching within an environment containing several thousand terminals. These studies are expected to lead to methods for handling large numbers of both terminals and processors and various packet access methods implemented within different hierarchy levels of large integrated command and control communication networks for the DOD. 13

PI-Write-up, Frank

SUPPORT FACILITY DEVELOPMENT: 14

During the past year, a basic packet radio simulator was developed. The simulator handles a single station, up to 48 repeaters and several hundred terminals of the same type. Imbedded in the simulator are models of the repeater, station, and terminals, two routing algorithms, non-persistent carrier sense and unslotted ALOHA random access schemes, zero capture receivers, single and dual data rate channels, omnidirectional antennas, and an interactive terminal to station protocol. All device actions required to initiate, relay, and receive a packet are simulated in the same sequence of events that would occur in the actual packet radio system. Last year's experience showed that for systems like packet radio, interactive, graphical display can greatly reduce the time required to carry out certain forms of system studies such as repeater data rate, power, and operating parameter variations. During the contract period, the first phase of a graphical display system, specifically designed to deal with network problems was developed. In addition, several stand alone analysis and design algorithms and programs were developed, including a repeater location algorithm and a basic network editor. 15

JAKE 20-MAR-75 17:37 24927

PI-Write-up, Frank

(J24927) 20-MAR-75 17:37;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, FRANK,NLS;4, >, 23-FEB-75 19:34 JAKE ;;;; ####;

PI-Write-up, Kirstein

1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Peter Kirstein
Department of Statistics and Computer Science
44 Gordon Sq
London, W.C.1
U.K.

During 1974 our activities have concentrated on the attachment of Hosts by means of a front-end computer with the minimum of software modification in the main Host, measurement and evaluation of the Network in our environment, investigation of the requirements for support activities, investigation of the management problems in the international environment, and the investigation of the requirements for internetworking. It should be noted that although this project has access to ARPA provided facilities, it is largely funded from other sources. Our activities are discussed with, and reported to IPTO; however only certain aspects of the work belong to main IPTO research program.

1. FRONT-END ATTACHMENT OF HOSTS

A considerable effort has gone into improving the attachment of the Rutherford Laboratory (RL) IBM 360/195 as a Host via a front-end PDP 9 computer. the Host is now attached to ARPANET in a rugged manner, and File Transfer is possible. Users connected directly to the RL 360 can get out to access ARPANET Hosts. The excercise showed where certain changes were required for the network environment--and improved, when the changes were implemented, local facilities also. Examples of the the improvements required are better flow control, HELP facilities and Status information.

The Cambridge Computer Aided Design Center (CADC) ATLAS Computer was attached also in a preliminary way; the PDP 9 provided a single TELNET channel. The technique was shown to work also in attaching the University of London Computer Center (ULCC) CDC 6000/7000 complex. These Hosts are not being incorporated operationally yet, since to have several Hosts attached simultaneously via the PDP 9 in this way requires some operating system modifications; these will be completed during 1975.

PI-Write-up, Kirstein

2. MEASUREMENT AND EVALUATION 8

In order to measure and evaluate ARPANET in our environment, it was necessary to build up a significant level of usage. An adequately diverse user community has been built up--most in projects which involve cooperation between UK and US research groups. We are analysing which features of networks have been most significant for our communities. We are preparing to measure the overhead in the different levels of protocol, and the traffic characteristics of users as a function of their applications. In this we have only developed much of the infrastructure during 1974, 9

3. SUPPORT ACTIVITIES 10

In order to develop the User Community of $2, and even for many administrative activities, a number of support activities are necessary. An on-line catalogue of documents is being prepared, and the documents distributed by the British Library. The POST System permitting the sharing of of an account by several users for mail purposes has been improved. A number of short FACT cards have been produced, and courses run. it has become clear that facsimile transmission is required in the coordination of certain projects--hence an activity of incorporating facsimile transmission into the message facilities has been started. 11

4.MANAGEMENT PROBLEMS 12

It has become clear that in this international environment, "Balance of payments" issues become very serious. With the number of government agencies involved, and international restrictions on payments, even finding criteria to measure a balance of payments problem is difficult. Some initial papers have been prepared on this subject, and are being discussed with various bodies in the US and UK. 13

5. INTERNETWORK ACTIVITIES 14

We have started preparing to implement the HOST-HOST protocols of the Internetwork Protocol proposed by Cerf and Kahn as part of a joint experiment with Stanford U and BBN. Independently, we have considered what the salient factors might be in the interconnection of networks, and have concluded a virtual circuit connection at gateways , with protocol mapping there, looks the most promising. Following this approach , we have started simulating the operation of the Packet Switch on the proposed British Post Office Experimental Packet Switching Service (EPSS), and are starting to implement the link to the CADC ATLAS as if it was via an Internetwork Gateway and EPSS. When EPSS becomes operational in late 1975, this interconnection will be implemented in earnest. 15

JAKE 20-MAR-75 17:41 24930

PI-Write-up, Kirstein

(J24930) 20-MAR-75 17:41;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, KIRSTEIN,NLS;4, >, 23-FEB-75 21:25 JAKE ;;;; ####;

PI-Write-up, Kleinrock

UCLA COMPUTER NETWORK RESEARCH
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Leonard Kleinrock
Computer Science Dept., 3732 B.H.
University of California at Los Angeles
Los Angeles, California 90024

At UCLA, we are concerned with advanced teleprocessing systems, emphasizing network measurements, packet radio for terminal networks and for satellite communications, secure operating systems and networks, and the general problem of resource allocation and sharing of finite-capacity resources. Below, we describe accomplishments for calendar year 1974, namely: results from the Network Measurement Center; performance evaluation of packet switching in satellite and ground radio; secure computing systems; tradecff analysis in large shared systems; performance evaluation of realistic operating systems models; and prediction and observation of network deadlocks and degradations.

(1) As a result of our activities as the Network Measurement Center (NMC), we have predicted and/or observed numerous and serious deficiencies with the ARPANET throughput i.e., network deadlocks and degradation. We predicted the "piggyback allocate" deadlock condition by examining the IMPSyS code and provided a solution to it which BBN has implemented. The famous "Christmas deadlock" was exposed as a result of experiments performed by the NMC; this deadlock occurred due to a lack of pointers to allocated buffers in the destination IMP and has since been corrected. As a result of our packetized speech experiments in which we measured single-packet throughput, we identified serious degradation due to out-of-order packets in the data stream. Furthermore, we identified the source of unacceptably long delays as being due to persistent looping caused by the routing procedure; we are currently working on a loop-free solution to this problem.

(2) We have been successful in evaluating the performance of packet switching in satellite communication systems and also in ground radio systems. The carrier sense acccess mode for ground radio has been completely analyzed in the single hop case; that access mode has been shown to be superior to slotted ALOHA. The serious problem of hidden terminals has been analyzed and the busy-tone solution to this problem has been shown to yield performance which is only slightly degraded as compared to a system with no hidden terminals. Furthermore, a reservation scheme for using the ground radio channel has been suggested and analyzed and appears to offer significant advantages over an important operating range. These random access modes offer distinct advantages over the more classical access modes when the traffic is bursty (as with terminal traffic and other interactive traffic). With the simple slotted ALOHA method as compared to the classical FDM methods, we find that we can reduce the required bandwidth and/or increase the number of users and/or reduce the delay over a wide range of system parameter choices; these advantages are even greater with the use of carrier sense. 6

(3) We have made important progress in the field of secure computing systems. The major thesis underlying our prototype development has been that those portions of an operating system which insure its security can be separated into small modules, whose security correctness can be mathematically proven. The bulk of our security work falls into three major categories: security design principles, security verification and certification, and prototype development. The majority of the design principles concern operating system structures. Strong evidence has been developed which argues that the traditional practices of placing supervisor code into user tasks, as well as locating virtual memory support at the lowest system levels, are design flaws with respect to security. The simplifications provided by virtual machine designs are now apparent. Progress has been made in developing greatly simplified approaches to I/O handling, usually one of the most serious sources of security flaws. Subtle, generic communication paths in systems have been identified. In addition, the general principle of "least common mechanism" was developed and illustrated. Work on verification and validation of security software has now progressed to the point where it is clear that the approach is viable. The design of a complete, practical, verifiable security kernel has been completed. Parts of that kernel code are now debugged and running. Design of the virtual machine monitor, which runs over the kernel, is also complete. Substantial portions of that prototype software are also debugged. Operating systems such as ANTS have already been successfully run in virtual machine environments. 7

(4) We have proven that there are very significant gains to be had with large shared systems. This has been demonstrated mathematically for a reasonably broad class of systems and we are currently exending that class. In particular, we have shown that as one scales up the system capacity and the throughput of any finite-capacity system, then the delay in passing through that system decreases by the same scale factor. We have shown this as a bound for general systems and as an exact result for more specific systems. The significance here is on the design of processing and communication systems. "Bigger is better" seems to be the message when the question is posed correctly. The exact form of these results and their impact on system design will bring considerable light to cost-effective system design. 8

(5) We have made major progress in the modelling of multiple-resource models of operating systems. We are now capable of identifying system bottlenecks, calculating throughput and delay and suggesting system configuration. This work is based on queueing network models of multiple resource operating systems. We have been able to generalize these models to include much of the realistic behavior of these systems. The costly errors of the past need no longer be made, and intelligent design can now be done. The cost-effectiveness of proposed design options and changes can now be evaluated. 9

(6) Analysis and measurements of the effect of network overhead on the performance as seen by the user have been conducted. We find that what appear to be harmless options offered to the system implementers can have significant effects on network performance. We have identified a number of these and have pointed out how they should be restricted in order that really major improvements (one to two orders of magnitude) in network throughput be achieved. The overhead on the communication lines due to the various levels of protocol have been found to have profound effects on throughput. The whole mechanism of allocating buffer space in the destination HOST and in the destination IMP must be very carefully examined. We find that the maximum line efficiency is roughly 80% under the most ideal conditions, and that it is as low as 1% under very common conditions. If the current traffic patterns continue, then we can obtain only 20-25% line efficiency in the ARPANET. 10

11

PI-Write-up, Kleinrock

(J24931) 20-MAR-75 17:44;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, KLEINROCK,NLS;3, >, 27-FEB-75 21:14 JAKE ;;;; ####;

PI-Write-up, Lebow

Information Processing Techniques Research
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: IrWin L. Lebow
Data Systems Division
M.I.T. Lincoln Laboratory
Box 73, Lexington, Ma. 02173

The following is a summary of Lincoln project activities during CY 74 in the four areas: A) Network Speech Compression, B) Speech Understanding, C) Surface-Acoustic Wave Convolvers for Packet Radio, and D) Airborne Command and Control.

A) NETWORK SPEECH COMPRESSION (B. Gold)

The overall goal of the Network Speech Compression (NSC) program is to investigate and demonstrate the technology required to carry speech over packet-switched networks. A potentially large economic gain can be realized if secure digital voice and data can be carried over a single integrated packet switched network. To achieve this goal, the NSC group is investigating the delay and throughput characteristics of the ARPANET for speech at different rates and is developing speech compression algorithms matched to Network properties. Lincoln is contributing to the areas described below:

1. Network Delay Measurements. Using the ARPANET "fake host" facility we have obtained measurements of fixed and random components of network transmission delays as well as delays caused by network flow control mechanisms. These results have helped focus attention on ongoing speech-oriented network modification.

2. CVSD Experiments on ARPANET. Lincoln and ISI communicated via CVSD at a 10 Kbps rate. As a result of this experiment, several specific network changes have been implemented to increase throughput.

PI-Write-up, Lebow

3. LPC Speech Demonstration on ARPANET. In Dec. 1974, the first successful network speech communications demonstration using LPC was carried out between Lincoln and CHI at 3450 bps. This experiment involved the use of a new Network Voice Protocol designed at ISI. Single-packet messages were used avoiding the reservation delay of the higher-rate multi-packet CVSD messages. The LPC speech at 3450 bps was of much higher quality than CVSD speech at 10 Kbps, and it was generally agreed that this speech link could be used without significant annoyance due to delays or speech quality. 9

4. Conferencing. Lincoln has specified and procured from industry 12 CVSD equipments. These will be used initially for conferencing experiments between Lincoln and ISI. 10

5. Voice Excited Linear Prediction Vocoders. A voice-excited LPC system is being implemented on the FDP to run at 8 Kbps. This data rate will permit specific comparisons between this system and the better known adaptive predictive coding (APC) algorithm. 11

6. Variable Rate LPC Processors. In packet communication, channel bandwidth utilization is adjusted automatically to the current trans- mission rate, which facilitates bandwidth reduction via variable rate coding techniques. We have achieved reasonable quality speech at 1400 bps by eliding silences, using decreased rates during voiceless sounds, and speaker adaptation. 12

B) SPEECH UNDERSTANDING (J. W. Forgie) 13

In April 1974, Lincoln Laboratory successfully demonstrated its mid-term speech understanding system for the government's Joint Advisory Committee for the Laboratory and later for other groups including the SUR Steering Committee. The system recognized sentences which were constrained to be appropriate to a particular limited task domain and a 250 word vocabulary. The system correctly recognized sentences from a large number of speakers without prior training or speaker-dependent adjustments. Informal testing showed about 75% correct sentence recognition. Controlled testing with more difficult sentence material (275 sentences from 6 speakers) showed approximately 50% completely correct with another 30% having errors in only one word or phrase. Processing time for a typical 3 to 4 second sentence was usually less than one minute. 14

C) SURFACE ACOUSTIC-WAVE CONVOLVERS for PACKET RADIO (E. Stern) 15

PI-Write-up, Lebow

In the second half of CY 74, a program was begun at Lincoln Laboratory to develop advanced acoustic convolvers for use as matched filters in Packet Radio. Our goal is to achieve high-time-bandwidth (TW) product systems which synchronize rapidly for protection against interference, jamming and interception. Conventional high-TW matched filters require long sync times. As a result, these systems often give up jamming and/or interception resistance to preserve reasonable access times. Our studies indicate that a SAW convolver which spreads 100 Kbps data over 100 MHz (TW = 1000) should synchronize in under one msec. 16

Furthermore, a code could be implemented with a structure which could be changed with sufficient rapidity to make cryptographic analysis and subsequent interception virtually impossible. In a preliminary experiment a data stream of 140 Kbps spread over 67 MHz (TW = 500) with a changing code was successfully decoded. We are currently perfecting a convolver with TW = 1000. 17

D) AIRBORNE COMMAND and CONTROL (A. J. McLaughlin) 18

Early in CY 74, a study was undertaken of aircraft digital communications with emphasis on satellite links. A report summarizing this work was delivered to ARPA. It was generally concluded that satellites provide the only means of achieving highly reliable long range communications to aircraft. Major increases in system data rates should be possible by exploitation of electronically steered, high gain satellite antennas and by efficient sharing of network capacity by packet access techniques. 19

During the second half of CY 74, a study was begun to investigate the applicability of computer networking technology to the Airborne Command Post Command and Control problem. It has been concluded that netting the computers of several airborne command posts and ground facilities provides the framework for achieving significantly enhanced performance. A series of experiments is being defined using the ARPANET as an R&D test bed to validate and demonstrate this conclusion in an environment in which both the ADP capacities and the communication rates are constrained. In these experiments some of the nodes will be airborne and some simulated by ground nodes on the network. The experimental program will permit evaluation and refinement of developmental techniques of data management, resource sharing, and packet communications in a realistic environment. 20

PI-Write-up, Lebow

(J24932) 20-MAR-75 17:47;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, LEBOW,NLS;3, >, 23-FEB-75 21:37 JAKE ;;;; ####;

THE ARC OFFICE-1 WORKSHOP UTILITY SERVICE
1974 ARPA Project Summary

Prepared for: ARPA IPTO Principal Investigators Meeting
March 12-14, 1975

Prepared by: James C. Norton
Augmentation Research Center
Stanford Research Institute
333 Ravenswood Avenue
Menlo Park, California 94025

I BACKGROUND

AUGMENTED KNOWLEDGE WORKSHOP (AKW) TECHNOLOGY

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." 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.

ARC'S "COMMUNITY PLAN"

The research and develoment 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. Exploratory development and operation of augmentation systems have been our substantive work.

A new stage of development has been established with the first year of a new "Workshop Utility Service." We are involving a much wider group of system users, a community, so that we can 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.

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, consciously aiming toward having experience and capabilities especially applicable to support remote and distributed groups of exploratory users.

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

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

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.

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

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

Supporting the user groups in learning how to use these tools, both at the individual user level and at the organizational application level.

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

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

The types of workshop services that we support at varying levels of capability include:

Collaborative Dialogue
Document Development, Production, And Control
Research Intelligence
Community Handbook Development
Computer-Based Instruction
Meetings And Conferences
Community Management And Organization
Special Knowledge Work By Individuals And Teams

III ACCOMPLISHMENTS -- Activity in 1974

A. OFFICE-1 service started 18 January 1974 via the ARPANET
The computer and technical services offered as the Utility commenced are being provided to over 200 users and their organizations. 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.

B. Development and use of new teaching methods
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.

Training users is necessary for the transfer; and requires a transfer of the additional technology used to support the training, e.g. on-line classroom sessions, remote linked-terminals, special "architect" seminars and on-line feedback services to users.

C. The Present User Community

The following organizations have been using the Workshop Utility:

RADC	Rome Air Development Center (Air Force)
Bell	Bell Canada Business Planning Group
ARPA	General ARPA use and National Software Works
CBI	ARPA: Computer-Based Instruction Community
Energy	ARPA: SRI Energy Project
NIC	ARPA: Network Information Center Users
Seismic	ARPA: Seismic Data Mgt System Development
BRL	Ballistic Research Laboratories (Army)
Hudson	Hudson Institute (ARPA subcontract)
NSRDC	Naval Ship Research and Development Center
SRI	Stanford Research Institute
NSA	National Security Agency

PI-Write-up, Nilsson

Artificial Intelligence--Research and Applications
(Computer-Based Consultant Project)
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Nils J. Nilsson
Artificial Intelligence Center
Stanford Research Institute
Menlo Park, California 94025

The Computer-Based Consultant (CBC) project at SRI has as its goal the development of a computer system that can engage in a spoken dialog with a human user to give him detailed advice about some particular subject area. We have selected maintenance, assembly/disassembly, troubleshooting, repair and operation of electromechanical equipment as a subject. To create a CBC system requires the integration of subsystems for natural language input and output, vision, diagnosis of faults, planning assembly/disassembly sequences and information retrieval. We are conducting research in each of these areas, and in addition, in the area of integrating these abilities into a coherent system. To provide specific goals and focal points for our research we have planned a series of demonstrations to occur in April of each year 1975 through 1978 inclusive. The final demonstration is planned to illustrate the feasibility of applying this technology to any of several DOD problem situations requiring interactive expertise.

During calendar 1974 we achieved the following specific results:

1. PROCEDURAL NET SYSTEM -- During 1974 we developed a prototype version of an integrated problem-solving and execution-monitoring system. The system uses a new method of storing in a computer memory information about actions. The new representation, called the "procedural net," represents actions in a structured way, so that the same action appears several times, represented at different levels of detail. The problem solving portion of the system easily solves some problems involving simultaneous goals and long chains of actions. The execution monitoring system engages in a dialog with a novice mechanic, varying the level of detail in response to the mechanic's needs.

2. MODELING -- We have designed and implemented a modeling system for describing the changing physical states of the workstation environment as tasks are being performed. This system allows one to specify in a highly modular form relations, predicates, derivation functions, state transition functions, and models of the actions that a human can be asked to do. Furthermore, we have defined in this system a set of standard relations (such as ATTACHED and REMOVED) and actions (such as INSTALL and CONNECT) that in effect provide a descriptive language in which electromechanical equipment and assembly/disassembly tasks can be described. This modeling package allows the system to maintain a coherent description of the workstation both while monitoring actual operations being done by a human and while considering hypothetical operation sequences during planning. 7

3. SELECTION OF VOCABULARY AND SEMANTIC CONCEPTS -- We have selected a vocabulary of about 650 English words extracted from protocols between expert consultants and novice mechanics. These have been grouped into semantic categories to help define the key concepts to be used in dialogs. This achievement is important because it permits us to begin work on the programs for syntactic and semantic analysis of natural language dialog in the computer consultant system. 8

4. SCANNING LASER RANGEFINDER -- We have implemented a laser rangefinder device that works like a radar to measure distances to objects in a scene. The device uses mirrors to scan a raster of points in a scene and determines range to each point by measuring the time of flight of a modulated CW laser signal. The range finder is playing an important role in our automatic scene analysis work because it provides a crucial additional parameter, range, that can be used with color and brightness data to interpret visual information. 9

5. POINTING SYSTEM -- We have implemented a pointing system that 1) points a laser beam at an object named by the user and 2) identifies an object pointed at by the user. The system can point at any of about a dozen parts of an air compressor that the user requests by voice input. It also can identify, by voice output, any of these same parts that the user might point at using a wand with a lightbulb on the end. The system uses a TV camera and a stored internal model of the compressor to control its pointing behavior. This system is an important I/O tool in our Computer-Based Consultant project and achieves several functions directly that would be much more cumbersome to achieve by conventional graphics. 10

6. DISTINGUISHING FEATURES SCENE ANALYSIS SYSTEM -- We have implemented a scene analysis system that takes as input a multisensory (i.e., range, intensity and color) image and locates objects in the scene based on the most distinguishing sensory properties of the object. For any problem, the system itself automatically generates the best strategy for finding an object based on internal models of its own perceptual abilities. Presently, the system can locate several objects and parts of objects in office scenes (desk tops, chairs, chair seats, doorways, etc.). 11

7. COMPILING QLISP USER CODE -- The computer language QLISP has been enhanced by the addition of the ability to compile user code containing QLISP. Previously, any part of the user's program which contained any QLISP at all could only be executed through the interpretative facilities provided by QLISP, but now the flexibility of the INTERLISP system has been used to interface the standard compiler to the particular ways in which QLISP extends the INTERLISP language. Our preliminary measurements show that compiled QLISP runs about twice as fast as noncompiled, but the main advantage should show up in the way users can now freely intermix QLISP code with their regular LISP without the need to introduce artificial separations for the sake of efficiency. 12

8. DEMONSTRATION SYSTEM -- We have put together a preliminary computer consultant demonstration system with rudimentary abilities to give instructions about assembly and disassembly of an air compressor. It also combines the laser pointer system (item #5 above) and can answer certain questions about the status of the compressor. This initial version will serve as a base for a much more elaborate system to be ready in April 1975. 13

JAKE 20-MAR-75 18:34 24938

PI-Write-up, Nilsson

(J24938) 20-MAR-75 18:34;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, NILSSON.NLS;2, >, 23-FEB-75 01:55 JAKE ;;;; ####;

PI-Write-up, Pirtle

INSTITUTE FOR ADVANCED COMPUTATION
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: DR. MEL PIRTLE
NASA AMES RESEARCH CENTER
MOFFETT FIELD, CALIFORNIA 94035

GENERAL:

The Institute for Advanced Computation (IAC) is a NASA/AMES Research Center Institute established under a NASA-ARPA inter- agency agreement with the charter to develop and operate a large computing and information storage facility for NASA, ARPA and other groups invited by either of these two. The major equipment of the institute is the ILLIAC IV, PDP-10/TENEX, the B6700 and the UNICON 690. This equipment, except for the B6700, is incorporated into a single system which also includes several PDP-11 management processors, small capacity rotating memories and assorted peripheral devices. the B6700 is operated as a secondary facility and is utilized for certain ILLIAC program compilations and for some test programs.

SUMMARY OF ACTIVITIES:

The primary resource in the IAC system today is the ILLIAC IV. We had two major objectives for the ILLIAC IV during this past year. These were:

1. To provide an ILLIAC IV service to network users; and
2. To continue the checkout of the machine with emphasis on improved reliability and maintainability.

There were significant accomplishments in both of these areas. At the beginning of 1974, we were providing about 4 hours of ILLIAC time daily to about a dozen ARPA and NASA users. at that time, the quality of this service, particularly the state of the ILLIAC IV, was uneven. Today, we schedule about 10 hours of ILLIAC IV time daily to a group of 18-20 ARPA and NASA users. The quality of this service is relatively uniform and the state of the ILLIAC IV is good. The results the users are getting are almost always correct when the ILLIAC IV is up. (The "ILLIAC IV" here refers to the I4 processor, array memory, and one-half of the total I4 disk memory capacity.) Some specific accomplishments in these two general areas are:

PI-Write-up, Pirtle

USER SERVICES: 9

Several notable ILLIAC IV user successes have been accomplished this year. For example, RAND has successfully completed a 30-day global climate simulation which runs for 2 hours on the I4. NASA'S GODDARD INSTITUTE FOR SPACE STUDIES (GISS) has successfully run 2-week weather simulations. The NAVAL RESEARCH LABS CASE PROGRAM (SIGNAL PROCESSING) became operational on the ILLIAC IV in November of this year. NRL has about 100 raw data sets to process; each data set takes about 1 hour of I4 processing time. SYSTEMS, SCIENCES AND SOFTWARE (SSS) has run several 3-D finite element calculations on the ILLIAC IV, each of which takes in excess of 1 hour processing time on the machine. SDAC successfully converted the first phase of thier "Long Wave" analysis programs to the ILLIAC IV. In general, all users have had successful runs on the ILLIAC IV with increasing frequency and of longer duration. Runs of 30 to 120 minutes are becoming common as compated to runs of 2-3 minutes at the beginning of the year. 10

ILLIAC CHECKOUT 11

The main objective was to increase reliability and maintainability. In this regard, many hardware modifications were made to replace faulty components and to correct design errors. Specific examples include: 12

> The design, fabrication and replacement of several boards in the control unit and in the processing elements (PE) and the setup of an "assembly line" modification to all PE'S which includes replacement of all terminators, sleeving of the back plane pins, and some logic corrections. 12a

In order to improve our ability to maintain the machine, IAC designed and developed a PE simulator (PESO). This simulator, which runs on the ILLIAC IV, accepts fault detection test data as input, simulates the execution of a PE, and produces as output specifically identified component failures consistent with the input test data. With this sophisticated diagnostic tool most PE faults can be quickly isolated to a specific component for corrective repair action. 13

In the early part of this year, IAC established (in conjunction with the owners) the objective of having one half of the I4 disk memory capacity solid, and routinely available for users. To achieve this objective, both of the disk memory subsystems were extensively modified. These modifications were principally in redistribution of power, adding capacitors, and logic error corrections. One disk system is relatively solid today, and is rountinely used successfully by I4 users. The second sub-system with further improvements is in final testing. A comprehensive I4 disk memory test was also developed. 14

PI-Write-up, Pirtle

UNICON 15

IAC'S objective for the UNICON memory this year was to make the device reliable and, by the end of the year, available for the storage of data files associated with ILLIAC IV applications. During the year, the UNICON controller (designed and fabricated last year) was checked out and numerous modifications were made to the laser recording unit. These efforts have significantly increased the stability of the device and have provided for verification of correct recording during the write operation. A "clean room" was built around the device, providing a clean environment for both the laser recording unit and the off-line storage of recording strips. The UNICON has been used for the storage of selected ILLIAC I4 user data files since November. 16

CENTRAL SYSTEM 17

The major accomplishment in the central (e.g. TENEX) IAC System has been the routine provision of a stable PDP-10/TENEX service to ILLIAC IV and selected network users. Specific enhancements in this Central System include: 18

1. The addition of a second PDP-10/TENEX system in May of this year. 18a

2. The implementation of a magnetic tape facility for ILLIAC IV users. This implementation included the design and integration of hardware, and the design and development of diagnostic and operating software. 18b

3. Design, development, and implementation of an "Operator-11" facility which runs on a PDP-11. The "Operator-11" greatly simplifies the IAC system opera- tors function by performing real time system status monitoring and providing a single interface to the various devices and software in the system. 18c

ORGANIZATIONAL 19

Considerable effort has been spent over the year to carry out the Transition from a development to an operational system. Organizationally, these efforts have included the establishment and staffing of an administration group, increasing the user support staff and extensively changing the operations group including the addition of personnel, retraining personnel, and restructuring the group. Key personnel additions have occurred at all levels in the Institute, notably with the addition of experienced management in the operation groups. 20

PI-Write-up, Pirtle

Future plans for the Institute are under study and development. Discussions have begun with both ARPA and NASA, with increased frequency in the latter half of the year. This planning activity involves a significant outlay of IAC management time. 21

MISCELLANEOUS 22

In September, IAC initiated a survey of application areas for the ILLIAC IV. Initial results were presented to ARPA in November. This survey will be completed in April or May of 1975. 23

Plans for accomodating classified data on the I4 are progressing and a preliminary approach has been reviewed with representatives from ARPA, the NAVY and the ARC facility contractor. 24

JAKE 20-MAR-75 18:38 24940

PI-Write-up, Pirtle

(J24940) 20-MAR-75 18:38;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, PIRTLE.NLS;3, >, 23-FEB-75 21:51 JAKE ;;;; ####;

PI-Write-up, Pratt

IMAGE PROCESSING RESEARCH AT THE UNIVERSITY OF SOUTHERN CALIFORNIA IMAGE PROCESSING INSTITUTE 1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: William K. Pratt
Image Processing Institute
University of Southern California
Powell Hall 306 - University Park
Los Angeles, California 90007

The research project in image processing is concerned with the analysis and development of techniques for efficiently generating, processing, interpreting, transmitting, and displaying visual images and two dimensional arrays of data.

IMAGE CODING:

During 1974 initial steps were taken by various Federal and commercial organizations toward the implementation of image coding systems based upon the adaptive linear predictive coding and transform coding concepts developed at the USC Image Processing Institute. Further research at USC in 1974 has led to the discovery of a new reconstruction algorithm for linear predictive and transform image coding systems. This algorithm which involves joint linear processing of groups of quantized image variables can reduce the mean square coding error by 20% to 50% and improve subjective quality as compared to conventional processing techniques. The hybrid DPCM/transform coding technique previously developed for coding individual image frames has been successfully extended to the removal of image redundancy between television or movie frames. In this system a two dimensional image transform is taken over small image blocks in each frame and differential pulse coding is performed on the coefficients between frames. Only a single frame of data storage is required. Bandwidth reductions of about 15:1 to 20:1 have been demonstrated.

IMAGE RESTORATION AND ENHANCEMENT:

A new method of computer image restoration has been developed to correct for astigmatism and curvature of field aberrations caused by imperfect optical systems. Neither type of aberration error can be removed by conventional filtering techniques because the error is space variant.

PI-Write-up, Pratt

The new restoration method involves an inverse geometric correction, space invariant image restoration, followed by another inverse geometric correction to compensate for the aberrations. A novel image restoration technique has also been discovered for the restoration of images degraded by general forms of space variant blur. This technique, called singular value decomposition, involves decomposing a blurred image into a series of sub-images of increasing resolution which are sequentially processed and recombined to yield the restored image. The beauty of the restoration technique is its ability to avoid numerical errors which plague conventional restoration methods. 9

IMAGE DATA EXTRACTION: 10

A prototype laser/computer optical feature extraction system has been built and is presently undergoing applications testing. This system consists of a minicomputer, electronic image scanning camera, and laser optical device which act together to detect and locate optical features in an image. The computer driven scanner searches for and locates gross objects or regions within a picture. Then the computer directed laser beam performs a high resolution spatial frequency analysis of the region. The spectral data is processed by a pattern recognition program on the computer to quantitatively determine the location and character of objects within a picture. The overall system is capable of analyzing large size, high resolution pictures in under five minutes. The optical processing technique has been combined with digital edge and texture analysis algorithms for image recognition and interpretation. This combined processing exploits the parallel processing capbility of the optical system and the adaptive nonlinear attributes of computer image manipulation. 11

IMAGE ANALYSIS: 12

A new model of the human visual system for color images has been found. This model accurately predicts known visual phenomena such as color sensitivity and color constancy under luminance changes. The major attribute of the model, in addition to its modelling accuracy, is its relative simplicity. The model has led to a form of image pre-processing in which a simple nonlinear operation is performed on a color image before conventional coding, filtering, enhancement, etc. The addition of this simple pre-processing, dictated by the model, has resulted in improved color image quantization, better subjective filtering results, and an additional bandwidth reduction of 2:1 for transform coding of color images. 13

PI-Write-up, Pratt

IMAGE PROCESSING FACILITIES: 14

In August 1974 a PDP-KI10 computer operating under TENEX was added to the USC Engineering Computer Laboratory for support of the image processing research program. The initial stages of implementation of a front end image processing software system on the PDP-10 have begun. The objective of the software system is to handle file manipulation for an image processing user and permit transparent access to large scale network computers for image processing tasks. Construction of a real time color image display TIP terminal was completed in 1974. Implementation of a real time color image magnetic tape recorder/playback unit is also underway. This unit is capable of recording a one minute segment of real time color television data at conventional scan rates and playing back the data at a slow rate for computer entry. The inverse operation of transferring digital pictures from a computer to a real time display is also possible. 15

JAKE 20-MAR-75 18:41 24941

PI-Write-up, Pratt

(J24941) 20-MAR-75 18:41;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, PRATT,NLS;6, >, 23-FEB-75 22:57 JAKE ;;;; ####;

PI-Write-up, Sutherland

Research on Command and Control Related Computer Technology
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: William R. Sutherland
Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Massachusetts 02138

This summary covers work done during the calendar year 1974.

I. Speech Compression - John I. Makhoul

In our speech compression research we have developed a time-asynchronous linear predictive vocoder that transmits high quality speech at low bit rates. The transmission rate varies according to the properties of the incoming speech signal. We have developed several methods for reducing the redundancy in the speech signal without sacrificing speech quality. Included among these methods are:

1) Adaptive optimal selection of predictor order. A new information theoretic criterion was employed to determine the optimal (lowest) order that adequately represents the speech signal in each analysis frame.

2) Optimal selection and quantization of transmission parameters. The reflection coefficients were judged to be the best for use as transmission parameters. An optimal procedure for quantizing the reflection coefficients was developed by minimizing the maximum spectral error due to quantization.

3) Variable frame rate transmission. A scheme was used to transmit speech parameters at variable rates in accordance with the changing characteristics of the incoming speech.

4) Optimal encoding. Variable length (Huffman) coding was used to encode the parameters at the lowest bit rate possible for a given quantization scheme.

5) Improved synthesis methodology. We found that with the time-synchronous method of analysis, improved speech quality was obtained when synthesis was also done time-synchronously.

II. Packet Radio Network - Jerry D. Burchfiel

PI-Write-up, Sutherland

The Packet Radio Network, an ARPA project in the CCC Program, is developing techniques for secure mobile digital communications which are inexpensive, difficult to detect or jam, and which permit a high degree of coexistence with existing applications of a broadcast band. The Packet Radio Network (PRN) consists of a shared common broadcast channel, fixed and mobile terminals which are sources and sinks of digital information, (e.g. test, graphics, encrypted compressed voice) repeaters which provide area coverage for mobile terminals by store-and-forward techniques, and stations which provide centralized control of network routing, statistics, debugging, and connections to other networks. 8

BBN is responsible for development of the PRN Station and the protocols which support interprocess communication between processes (terminals and hosts) of the PRN and other processes on the PRN or other networks. We are active in design of systemwide protocols, and have published numerous packet Radio temporary notes in this area. Initial integration of Station hardware will begin at BBN in March 1975, and an initial demonstration of working Station-terminal protocols is planned for August 1975. 9

10

JAKE 20-MAR-75 18:44 24944

PI-Write-up, Sutherland

(J24944) 20-MAR-75 18:44;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, SUTHERLAND.NLS;6, >, 23-FEB-75 22:02 JAKE ;;;; ####;

PI-Write-up, Walker

SRI SPEECH UNDERSTANDING RESEARCH
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Donald E. Walker
Artificial Intelligence Center
Stanford Research Institute
Menlo Park, California 94025

INTRODUCTION -- There have been five major accomplishments in the Speech Understanding Research Project at SRI this year. First, we have completed with the System Development Corporation the initial implementation of a system being built jointly by the two contractors. Second, within this system we have developed a control strategy, embedded in the parser, that is able to focus the operation of the system and to reduce time spent on incorrect interpretations. Third, we have written a performance grammar, based on studies of task-oriented dialogs, that relates information about meaning and about stress and intonation patterns to that of syntax in the process of analyzing an utterance. Fourth, we have developed a way of partitioning spaces in a network representation for semantic structures that is particularly well-suited for working with complex task domains. Fifth, we have introduced procedures, building on the semantics, that allow us to establish a discourse history, that is, to use information from previous utterances in the analysis of the current one.

1. SYSTEM IMPLEMENTATION -- In the joint system, SRI is concentrating on grammar, semantics, pragmatics (information about a situation that is changing dynamically), and the integration of these components through a parser to provide analyses of the structure of English questions, statements, and commands so that they can be interpreted and an appropriate response made. The parser also constitutes the mechanism for coordinating knowledge about acoustics, phonetics, and phonology toward the analysis of spoken language. The system will allow the efficient use of many different kinds of knowledge in processing an utterance. Currently, we are working on two task domains: one provides data management capabilities for querying a file of information on attributes of ships; the second involves using the computer as a consultant to guide a technician in the repair of electro-mechanical equipment.

2. A FOCUSED PARSER -- The lack of separation between words in fluent speech and the variability in pronunciation as a function of context require a different parsing strategy for speech understanding than that needed for text understanding and question answering. Consequently, we have developed a parser that both can predict words or phrases on the basis of context and can build up phrases and clauses from words that have been identified acoustically at some place in an utterance. The parser coordinates knowledge relating to the structure of English, to the nature of the task being undertaken, to the relevant features in the world of the task domain, to previous conversations, to variations in stress and intonation (prosodic features), to the effects of noise, and to individual differences in speaking, to mention only some of the critical elements involved. The uncertainty of the input and the variety of kinds of knowledge required can lead to consideration of a large range of interpretations in the analysis of an utterance. The parser contains mechanisms that enable it to examine the most reasonable alternatives first and to focus its activities with respect both to processing time and to space used in the computer. 6

3. A PERFORMANCE GRAMMAR -- The grammar developed for the system differs from other grammars in several important ways. First, it is based on a careful study of protocols recorded of people performing tasks like those in the area of intended application. Thus it reflects the way people actually talk, rather than how they should talk. Second, the grammar is written so that it is not restricted to a particular kind of parsing strategy. That is, it can be used both in building up more complex grammatical structures from words that have been identified acoustically, and in working down from a grammatical structure to the words it contains. Third, the grammar incorporates in its rule statements information on semantic and prosodic features, as well as on grammatical ones. Thus, when the grammar is compiled with the lexicon into an internal representation that the parser operates on, it results in a more efficient use of knowledge sources in the system. 7

4. PARTITIONED SEMANTIC NET SPACES -- Semantic information, which is directly coordinated with the grammar, is embodied in a network in which the nodes represent concepts--objects or events--and the arcs represent structural relations among the concepts. Our semantic nets differ from other network representations in the way nodes can contain structural relations and in the partitioning of nodes and arcs into net spaces. Net spaces provide a uniform mechanism for distinguishing hypothetical and imaginary situations from reality. The result is a more effective procedure for encoding the multiple alternative states of a changing task or situation. The hierarchical structure of the net spaces results in a more economical storage of information, because elements common to related nodes can be stored once for all of them. This mechanism allows attention to be focused on particular levels of detail as appropriate. In addition, the net space partitioning makes it much easier to handle general statements and rules, items that have caused considerable difficulty in previous programs for doing semantics. 8

5. DISCOURSE ANALYSIS -- In discourse, whether spoken or written, a given utterance or statement may depend directly on what has already been said or written for even a minimal understanding of the content. Pronouns are used to avoid having to repeat the names of people or objects, and, particularly in spoken language, the subject or predicate may be omitted and the listener expected to recover it from the context. For our speech understanding system, we are developing procedures that can handle instances of anaphoric reference and of ellipsis. We now have programs that provide these kinds of information based on an analysis of the previous utterance. We intend to extend these mechanisms so that for complex goal-oriented tasks we will be able to predict what a person is saying and to use that information to increase the efficiency of the system. 9

JAKE 20-MAR-75 18:47 24946

PI-Write-up, Walker

(J24946) 20-MAR-75 18:47;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, WALKER,NLS;2, >, 23-FEB-75 14:34 JAKE ;;;; ####;

JAKE 20-MAR-75 18:50 24948

PI-Write-up, Watson

SRI Augmentation Research Center
1974 ARPA Project Summary 1

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975 2

Prepared by: Richard W. Watson
Augmentation Research Center
Stanford Research Institute
Menlo Park, California 94025

3

MAJOR R&D ACCOMPLISHMENTS FOR CALENDAR YEAR 1974 4

1) Released a Major New Version of NLS (NLS-8) 5

The main design goals of this system were listed in last year's accomplishments. The significance of this year's accomplishment was their successful implementation, checkout, documentation, and user training. 5a

NLS-8 from the user point of view consists of new capabilities for tailoring the interaction to user preferences through a User Profile database; a multilevel Help capability, providing prompting and optional showing of next-alternative command terms, command syntax, or relevant entry into a Help database with a simple query facility for full online documentation (Hardcopy documentation is derived from these facilities as well); greater consistency in command language forms; new commands where there was a strong need; and the ability to write sequences of commands and have them executed from a file. 5b

From a system point of view, NLS-8 has a number of structural changes and ideas for specification of the user interaction at a high level, compilation of this specification into a data structure that in conjunction with the User Profile controls an interpreter. These ideas are being adopted in other ARPA programs such as the National Software Works (NSW), ISI message system, and are under consideration in ARPA programs under plan. 5c

2) Released Line Processors to Support DNLS on Cheap, Commercial CRT Terminals 6

Last year we designed a micro computer based box that would adapt a class of low cost commercially available alphanumeric CRT systems into true two dimensional devices for output (multi-window split screen operation) and allow use of two dimensional input pointing devices. 6a

This year a number of these devices are in field use supporting terminals of four different manufacturers. The Air Force through the NSW program will be exploring their use. The basic display techniques and communication protocols were published and have influenced other ARPA contractors in their design of terminal control systems. 6b

The Line Processor is now being adapted to handle general graphics displays as well as offline cassette devices. 6c

3) Designed Advanced Protocols for Resource Sharing on the ARPANET 7

As part of the work for NSW, a new approach to protocols has been designed and thoroughly documented for inter-process and/or interhost communication and control. We call the approach a Procedure Call Protocol. It creates a distributed programming and process control environment. In effect it makes procedures and data structures of remote software systems as accessible to the programmer as those within his own system. 7a

This approach will make it quite easy for new systems to be constructed from appropriate parts of existing systems and should greatly facilitate crossnet and cross process resource sharing. 7b

4) Designed a Distributed-Service Frontend System 8

A mini-computer system has been designed to provide a coherent command language environment for the multi-tool NSW system. We expect this approach to have considerable impact on system organizations of other systems to operate within an ARPANET like marketplace of information services. It will not only supply services to users to simplify the number of conventions they have to know when using a variety of ARPANET tools, but also provide services for tool builders to greatly simplify the task of specifying the user interface. The Frontend will provide all terminal handling and command parsing facilities and thus decrease the cost of providing new tools. 8a

The initial Frontend will be implemented on a PDP-11 running the ELF operating system. We have developed a cross compiler and debugging environment for use of our system programming language L-10 for use with the PDP-11. 8b

5) Designed a Distributed-Service Operating System Interface 9

One important goal of the work ARC is doing on NLS and the NSW Frontend is to provide mechanisms to simplify and decrease the cost of movement of the programs developed to a variety of machines and operating system environments. To this end we have designed a virtual environment that all application level programs will see as their operating System Interface (OSI). The OSI will in turn contain the actual calls on a given operating system. 9a

6) Designed Extensions for the NLS File System 10

Designs have been completed that will enable NLS to support text and other media such as Graphics, voice and so forth in an integrated fashion. Many systems support text, or speech, or graphics, but this development will open the way for tool developments that utilize mulltimedia. 10a

7) Designed Access Support for R&D Software Workers from ARPANET Sources 11

We made the plans and ordered the hardware necessary to allow us to obtain the computer needed by our development staff from ARPANET hosts. 11a

The significance of this development is that it is the first case of a fairly large project giving up its local computing capacity to obtain equivalent capacity through the Network, from sites specializing in providing service. 11b

JAKE 20-MAR-75 18:50 24948

PI-Write-up, Watson

(J24948) 20-MAR-75 18:50;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, WATSON,NLS;5, >, 23-FEB-75 22:34 JAKE ;;;; ####;

PI-Write-up, Wintz

Image Analysis and Modeling
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: P.A. Wintz and T.S.Huang
School of Electrical Engineering
Purdue University
W. Lafayette, Indiana 47906

This summary covers work done during calender year 1974. we have concentrated on research on image structure analysis leading to applications in information extraction, image enhancement, and coding.

1. Image Segmentation.

We have developed a computer algorithm called blob which segments an image into regions so that points in the same region have similar characteristics. The blob alogrithm was used to increase the accuracy of classifying multispectral ERTS data. These data had been classified point by point using spectral signatures. By the application of blob, regional classification became possible. This not only increased the classification accuracy (by about 5 % on the average) but also reduced the calssification time (by a factor of about 30:1).

2. Image Decomposition.

Many image processing tasks are facilitated if the image is decomposed into simpler components and each component is handled according to its own characteristics. We have been developing algorthms to decompose an image into three components: edges, background (slowly-changing), and textures. In this connection we have developed several edge detection and texture analysis algorithms. This idea of image decomposition has been applied to image noise reduction. Applying a wiene filter to a noisy image reduces the noise but also blurs the edges of the object in the NIC 24950 Part of NIC 24950 image. By treating the edges separately, one is able to reduce the noise and at the same time retain edge sharpness.

3. Error-Free DPCM Codes for ERTS Imagery.

PI-Write-up, Wintz

The different statistics of ERTS imagery have been measured, and based on these several easily implementable classes of DPC_M codes have been developed. They reduce the bit rate form 8 bits per picture element to 1.0

about 4 bits per picture element, A slightly more complicated adaptive code reduces the bit rate to about 2,5 bits per picture element, These codes are error-free in the sense they do not introduce any distortion to the images. 11

4, Recursive Image Restoration. 12

By modeling images as two-dimensional random fields, it becomes feasible to apply the Kalman formulation of recursive estimation to image restoration, However in trying to derive the optimum two-dimensional estimator, one encounters fundamental mathematical difficulties, We have never the less developed several suboptimum estimators which in practice may perform almost as well as the optimum. 13

5, Three-Dimensional Reconstruction. 14

We have studied several methods of reconstructing three-dimensional structures from two-dimensional x-ray pictures, Specificaly we have investigated quantizing effects, beam divergence, and unknown beam strength, It was found that a beam divergence of up to ten degrees can be tolerated and that an unknown beam strength introduces a ring-like structure to the reconstruction. 15

6, Iterative Image Restoration. 16

Many image degradtions can be approximated by linear models, Then image restoration on the computer becomes the problem of solving a a set of linear algebraic equations, Because of image noise conventional iterative methods are unusable, We have developed an iterative method for doing the restoration which offers a tradeoff between noise and image sharpness. 17

7, Image Phase. 18

Although it is well known that the phase of the fourier transform of an image is generally move important than the magnitude, most past work on two-dimensional digital filter design concentrated on magnitude specification only, We demonstrated that the phase accuracy of image processing filters are extremely improtant, Even if the desired filter has linear phase, failure to specify it may lead to desaster, We also developed methods of designing two- dimensional digital filters which specify both the magnitude and the phase. 19

JAKE 20-MAR-75 18:56 24950

PI-write-up, Wintz

(J24950) 20-MAR-75 18:56;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, WINTZ.NLS;3, >, 23-FEB-75 22:13 JAKE ;;;; ####;

JAKE 20-MAR-75 17:50 24952

PI-Write-up, Magill

SPEECH DIGITIZATION AND TRANSMISSION ON THE ARPANET BY ADAPTIVE LPC TECHNIQUES
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: D.T. MAGILL
STANFORD RESEARCH INSTITUTE,BLDG. 306A
333 RAVENSWOOD AVE.
MENLO PARK,CA. 94025

THIS SUMMARY COVERS WORK DONE DURING CALENDAR YEAR 1974. OUR RESEARCH HAS BEEN CONCENTRATED IN TWO THEORETICAL AREAS. IN ADDITION, CONSIDERABLE EFFORT HAS BEEN DEVOTED TO THE DEVELOPMENT OF A COMPUTER FACILITY THAT WILL PERMIT REAL-TIME, LOW-RATE DIGITIZED SPEECH TRANSMISSION ON THE ARPANET.

BROADLY SPEAKING OUR OBJECTIVE HAS BEEN THE DEVELOPMENT OF BETTER QUALITY AND LOWER DATA RATE VOICE DIGITIZERS. THE BASIC APPLICATION IS TO THE PROBLEM OF ECONOMICALLY ACHIEVING SECURE COMMUNICATION WITH HIGH INTELLIGIBILITY AND GOOD QUALITY. OUR RESEARCH EFFORT HAS BEEN IDENTIFIED IN TWO MAJOR TASKS WHICH ARE DESCRIBED SEPARATELY BELOW. IT SHOULD BE NOTED THAT PRIOR TO 1975 OUR MAJOR SIMULATION RESULTS HAVE BEEN ACHIEVED IN NON-REAL TIME. REAL-TIME DEMONSTRATIONS ARE PLANNED IN EARLY 1975.

A. SPEECH DIGITIZATION BY LPC ESTIMATION TECHNIQUES

THE OBJECTIVE OF THIS TASK HAS BEEN THE DEVELOPMENT OF AN LPC SPEECH DIGITIZATION SYSTEM THAT ACHIEVES LOWER DATA RATES(WHILE MAINTAINING HIGH QUALITY) THAN PREVIOUS SYSTEMS. THE RESULT OF OUR EFFORTS HAS BEEN THE DEVELOPMENT OF THE DELCO ALGORITHM FOR SPEECH ENCODING. DELCO RECOGNIZES REDUNDANCIES IN THE LPC PARA- METERS AND REMOVES THEM. CONSEQUENTLY, THE DATA COMPRESSION IS INCREASED BEYOND THAT ACHIEVED WITH CONVENTIONAL LPC TECHNIQUES. AN ADDITIONAL REDUCTION OF APPROXIMATELY 3 TO 1 HAS BEEN DEMONSTRATED. DELCO PRODUCES ASYNCHRONOUS DATA TRANSMISSIONS AS DO MOST GOOD COMPRESSION TECHNIQUES. PACKET COMMUNICATION SYSTEMS, EITHER PACKET SWITCHING OR RADIO, ARE IDEALLY SUITED TO CAPITOLIZE ON SUCH TECHNIQUES. IN SUMMARY, DELCO HAS BEEN DEMONSTRATED TO PROVIDE A REDUCTION OF TRANSMITTED BIT RATE(FOR LPC PARAMETERS) OF APPROXIMATELY THREE TO ONE WITH ESSENTIALLY NO SPEECH QUALITY DEGRADATION.

PI-Write-up, Magill

B. SPEECH DIGITIZATION EXCITATION STUDY 8

THE OBJECTIVE OF THIS TASK HAS BEEN TO DEVELOP MORE EFFECTIVE METHODS OF ENCODING THE EXCITATION SIGNAL USED TO DRIVE THE LPC SYNTHESIZING FILTER AT THE RECEIVER. PREVIOUS TECHNIQUES HAVE EITHER REQUIRED TOO HIGH A BIT RATE OR HAVE SUFFERED FROM THE WELL-KNOWN PROBLEMS OF PITCH EXTRACTION. OUR INITIAL RESEARCH CONSIDERED SOME METHODS OF IMPROVING THE PITCH EXTRACTION PROCESS. HOWEVER, OUR MAJOR EFFORT HAS BEEN DEVOTED TO MORE ROBUST METHODS OF ENCODING THE EXCITATION SIGNAL. THAT IS, WE HAVE CONCENTRATED ON THE DEVELOPMENT OF AN ENCODING SYSTEM THAT IS CAPABLE OF OPERATING IN THE PRESENCE OF BACKGROUND NOISE AND/OR MULTIPLE SPEAKERS. TWO SYSTEMS HAVE BEEN DEVELOPED AND DEMONSTRATED AS A RESULT OF THIS EFFORT. BOTH DIGITIZERS ARE BASED ON THE SAME FUNDAMENTAL CONCEPT. 9

NIC24952
PART OF NIC 24980 9a

THESE DIGITIZERS ARE BASED ON SHORT-TERM POWER SPECTRUM MATCHING AND THE PRINCIPLE OF THE VEV. BOTH SYSTEMS USE RESIDUAL ENCODING AND LINEAR PREDICTION AND ARE KNOWN AS RELP SYSTEMS. THE FIRST USES ADAPTIVE DELTA MODULATION(ADM) FOR ENCODING THE RESIDUAL WHILE THE SECOND USES ADAPTIVE DIFFERENTIAL PULSE CODE MODULATION(ADPCM). THE RELP/ADM SYSTEM HAS BEEN DEMONSTRATED TO PROVIDE VERY GOOD QUALITY IN THE RANGE OF 8 TO 9.6 KBPS. FURTHERMORE, IT HAS BEEN DEMONSTRATED TO OPERATE SUCCESSFULLY WITH TWO SIMULTANEOUS SPEAKERS. THE RELP/ADM SYSTEM WAS THE FIRST LPC EQUIVALENT TO THE VEV TO BE DEMONSTRATED. 10

THE RELP/ADPCM SYSTEM HAS ALSO BEEN DEMONSTRATED. THE MOTIVTION FOR THIS SYSTEM WAS TWO-FOLD. FIRST, IT WAS DESIRED TO COMPARE THE QUALITY DIFFERENCES BETWEEN THE TWO ENCODING SCHEMES. SECOND, IT MAY BE POSSIBLE TO ACHIEVE 4800 BPS OPERATION WITH THE RELP/ADPCM SYSTEM WHILE PROVIDING ACCEPTABLE QUALITY. WE HAVE DEMONSTRATED THAT THE ADM SYSTEM PROVIDES BETTER HIGH FREQUENCY CONTENT IN THE SYNTHETIC SPEECH THAN THE ADPCM SYSTEM. HOWEVER, THE LATTER APPEARS TO BE ACCEPTABLE. WE HAVE THEORETICALLY SHOWN THAT 4800 BPS OPERATION SHOULD BE POSSIBLE; HOWEVER, ADEQUATE QUALITY HAS YET TO BE DEMONSTRATED. 11

IN SUMMARY, TWO FORMS OF THE RELP SYSTEM HAVE BEEN DEMONSTRATED TO PROVIDE VERY GOOD QUALITY IN THE 8 TO 9.6 KBPS RANGE. FURTHERMORE, THESE RELP SYSTEMS HAVE BEEN DEMONSTRATED TO OPERATE SUCCESSFULLY IN PRACTICAL ENVIRONMENTS THAT GIVE CONVENTIONAL VOCODERS GREAT DIFFICULTY. 12

JAKE 20-MAR-75 17:50 24952

PI-write-up, Magill

(J24952) 20-MAR-75 17:50;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, MAGILL,NLS;3, >, 23-FEB-75 21:45 JAKE ;;;; ####;

PI-Write-up, Farber

Computer Network Security
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: David J. Farber
University of California at Irvine
Department of Information and Computer Science
Irvine, California 92664

The charter of this research group is the study of a new protocol for networks in which the transmitting and receiving processes designate each other by the use of mutually agreed upon algorithms. This protocol is to be investigated and documented on the UCI distributed computer system (DCS). We are developing and will implement, evaluate, and document an internetworking protocol for the DCS based on traffic flow security, which can be applied to other computer networks such as the ARPA Net. The project is currently focusing on two research areas. The research team includes Ken Larson and John Pickens.

Area 1. Protocol Design and Testing

A protocol based on dynamic process renaming has been developed which provides the basic elements of a security system. The protocol scheme is based on the creation of a chain of names. Each message contains a return address which is the next name by which the sending process will be known. Thus, if process A sends a message to process B, process A would include its new name, C, as the return address. When process B returns a message to process A, it would use the new name, C, and would send its new name, D, as the return address. The names used can be generated by a pseudo-random number generator or other suitable scheme. Multiple use of a name can be prevented by maintaining a list of names in use.

This simple protocol scheme does not allow for the loss of a message or the introduction of noise into a message. To handle these possible problems, the protocol has been extended to include a backup means of reestablishing the communications link if the names are altered or destroyed. A scheme has also been developed to prevent these errors from influencing the generation of names by cluttering the list of names in use. A difficulty in such a scheme is the initial creation of the chain of names. To strengthen this aspect of the protocol a procedure for dynamic user identification has been developed.

To test the protocol, name generation, and user identification procedures we have developed a simulation of the system. The initial design is being implemented on a conventional system and is designed around a control terminal. A system consisting of two hosts will be simulated. The control terminal will be used to simulate user entries and logons from as many as ten terminals on each host. The control terminal will also be used to destroy or alter messages on the network. This flexibility will allow us to fully exercise the protocols. 8

To test the protocol in an asynchronous environment, we will then transfer the simulation to the Distributed Computer System network at the University of California, Irvine. The design of the current implementation is specifically chosen so this change can be easily accomplished. The same flexibility will be maintained in the new simulation, but two separate machines will be used to simulate the two hosts, and files representing pseudo-terminals will be used to maintain operation at reasonable speeds. The initial simulation includes no use of encryption techniques; however, the design of the protocol specifically allows for this option if desired. 9

Area 2. Security Detection, Debugging, and Monitoring 10

The goal of this effort is to create descriptive and control mechanisms which enable the observability and verification of distributed process behavior. Security detection, debugging, and monitoring with distributed processes all have the following attributes in common: 11

1) Monitoring hooks must be placed either statically or dynamically
   in key locations,
2) Intra-module communications must be observable and describable
   at a high level,
3) The sequential and/or parallel behavior of a set of modules must
   be observable and describable at a high level,
4) Arbitrary behavioral attributes must be observable and
   describable at a high level. 11a

Additionally, in the case of debugging, it may be desired to invoke breakpoints and DDT-like operations internal to modules as a function of externally observable conditions. 12

PI-Write-up, Farber

This effort has been developed into four phases: In the first, a meta-language is being developed which may be used to describe the syntax and semantics of single logical message channels between processes. Some of its key attributes are: 1) the language is a human engineered BNF with facilities for structured editing and macro programming, and 2) metarules within the BNF may be made dependent upon external conditions (such as the state of a program graph or other communication channels). In the second phase the mechanisms for describing the sequential/parallel behavior of a group of distributed modules are being investigated. Past and present work on graph models has been examined, and a modified model is being created which is better suited for use in the context of this effort. In the third phase all the levels of potential monitoring/debugging will be integrated into a single consistant system. Constructs for controlling these tools and passing smoothly from one level to the next will be outlined. In the fourth and final phase two potential implementations will be mapped out. The first will consider the Distributed Computer System network and the second will consider the ARPA network. In both cases the problems of obtaining effective monitoring taps will be considered, and the requirements upon host operating systems (if any) and/or programming conventions will be outlined.

JAKE 20-MAR-75 17:19 24954

PI-Write-up, Farber

(J24954) 20-MAR-75 17:19;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, FARBER,NLS;3, >, 23-FEB-75 17:45 JAKE ;;;; ####;

PI-Write-up, Weiner

Advanced Computer Science Applied to Defense Needs
1974 ARPA Project Summary 1

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975 2

Prepared by: Peter Weiner
The Rand Corporation
1700 Main Street
Santa Monica, California 90406

3

Rand's current ARPA-IPTO computer research project started 1 July 1974. All research summarized below was performed during the last six months of calendar year 1974. 4

1. Intelligent Terminal Research - Robert H. Anderson 5

We have explored the design of "user agents" for intelligent terminals, by creating prototype agents capable of operating on Rand's PDP-11/45 minicomputer. In December 1974 we demonstrated RITA (Rand's Intelligent Terminal Agent), an agent whose behavior is governed entirely by a set of production rules (i.e. pattern-action rules). The design of RITA is heavily influenced by the MYCIN system of E. H. Shortliffe et al. at Stanford University; we have created a MYCIN-like system (but without the ability to state numerical levels of assurance) in the "C" language under the UNIX operating system on the PDP 11. Our rule-based system is capable of either goal-driven behavior, starting with a designated goal rule, or else pattern-driven behavior, in which the pattern parts of rules are scanned until a valid match is found -- and then that cycle is repeated. 6

We have created rule sets which allow RITA to handle various ARPANET protocols, including logging into remote hosts and executing file transfers automatically. During the next six months, we expect to create sets of rules allowing RITA to handle essentially all of File Transfer Protocol, including the many error conditions which may arise. We also expect to create rule sets implementing a "trickle file" transfer process, in which very large files are automatically broken into smaller segments that are individually transmitted via ARPANET, then re-assembled into a large file at the destination host. 7

2. ARPA Program Planning Support: Intelligent Terminals - Robert H. Anderson 8

Rand has assisted ARPA-IPTO in planning a major research program in Intelligent Terminals. Rand was asked to prepare recommendations to ARPA covering such questions as: should ARPA pursue this program and 9

why? What is the role of industry? What is the unique DOD relevance, if any? What is the projected benefit to the military? What research is being pursued at this time? What critical ideas, accomplishments or technology exist now that make the program possible? What are proper goals, milestones, and program management plans? 10

We completed a report entitled: "Intelligent Terminals: DOD Requirements and Plan for an ARPA Research Program" which addresses these questions. We concluded that a five-year research and development program could lead to Intelligent Terminals having three main important attributes: 11

(1) they are capable of "intelligent" behavior, derived from the use of knowledge bases containing heuristics, assertions, and data about a limited domain. Examples of such domains are: the behavior of external systems and how to deal with them, and an individual user's preferences and characteristics; 11a

(2) they have excellent human factors in the design of the man-machine interface. They are capable of interpreting natural human input signals, such as limited voice commands and hand-printed annotations; 11b

(3) they are capable of handling common, mundane tasks -- such as text management, task management, message management, and calendar-related scheduling -- quickly and efficiently. We have recommended three application areas within DOD for testing of Intelligent Terminal prototypes: intelligence analysis; logistics and maintenance management; and tactical operations information systems. We are continuing to work with ARPA-IPTO in this program planning effort. 11c

3. ARPA Program Planning Support: Very Large Data Bases - R. Stockton Gaines 12

Rand has provided assistance to ARPA-IPTO in developing a research program in the area of Very Large Data Bases. An initial investigation of the area was conducted to determine if such a research program would be appropriate, and to identify some of the central problems the program could attack. In conjunction with this, informal meetings were held at Rand to solicit the views of nationally recognized experts in the fields of data base management, artificial intelligence and cognitive psychology. A report has been prepared as a result of these activities, entitled "very Large Data Bases: An Emerging Research Area." We have concluded that this area is one in which significant advances in the next few years are likely, and that an ARPA research program could make an important contribution. 13

pI-Write-up, Weiner

There is a substantial and rapidly increasing need within DoD to make use of very large amounts of information via computers, and an APRA program to spur developments of appropriate facets would have a high probability of producing important and useful results. Among the areas of research identified in the report, the use of AI concepts in data base systems was particularly recommended. We are continuing to work with ARPA-IPTO in this program planning effort. 14

4. System Support Activities - Peter Weiner 15

Much of Rand's computer research is being performed on a PDP-11/45 minicomputer, using the UNIX operating system developed by Bell Laboratories. We are developing several support programs to make the facilities of UNIX more useful to the ARPA research community. We have completed a working prototype version of a CRT-oriented text editor (the Rand Editor) which operates within the UNIX environment on Ann Arbor 40-line text terminals. The Rand Editor allows multiple text windows onto one or more files, and movement of two-dimensional chunks of text within a file. We have also started development of a Network Access Program (NAP) to interface PDP-11s with UNIX to the ARPANET as a host machine. We expect to complete the UNIX-NAP in May, 1975. 16

JAKE 20-MAR-75 18:53 24955

PI-Write-up, Weiner

(J24955) 20-MAR-75 18:53;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, WEINER.NLS;5, >, 23-FEB-75 22:26 JAKE ;;;; ####;

PI-Write-up, Amarel

RESEARCH ON SECURE OPERATING SYSTEMS AND DATA BASE SYSTEMS AND ON AUTOMATIC PROGRAMMING
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Saul Amarel and C. V. Srinivasan
Department of Computer Science
Rutgers University
New Brunswick, N. J. 08903

This summary covers work done during calendar year 1974 in the areas of Secure Systems and Automatic Programming.

I. SECURE SYSTEMS

1. Protection and Integrity of Data Bases (Minsky)

Our research has centered on the process of user - data base (DB) interaction, and on the structure of the DB. The conventional protection techniques - usually called access control - were found to be insufficient. The concept of intentional resolution was defined, and was shown to be an important aspect of protection. The DB is conceived as a programming system which has procedural as well as structural components. We started work on the design of a Data Base Language (DBL) for constructing such a system. In parallel, we are planning an experimental design of a DB system using the SIMULA language. This work has been documented in the following SOSAP reports by Minsky: TR-7 "On the Interaction with Data Bases", TR-8 "Comments on Privacy of Data Bases", TR-9 "On the Resolution Power of Privacy Protection in Data Base Systems" (also to appear in Comm. ACM), TR-10 "On the Formation of Abstract Data Types", TR-11 "Protection of Data-Bases, and the Process of User Data-Base Interaction", and TM-6 "Another look at Data Bases".

2. Generating Valid Implementations (Welsch)

Our approach has been to specify a kernel of an operating system, and to use a precondition generator to validate properties of the kernel. Experimental work on validation has pointed out difficulties in stating formal assertions for large programs. A critical review of this approach is now underway.

3. Resource Enhancement for Operating Systems Support (Ruschitzka) 5e

Exploratory work has begun on architectural features which would support an operating system in maintaining its integrity and privacy characteristics. 5f

II. AUTOMATIC PROGRAMMING 6

1. Studies and Systems in Specific Domains 6a

1.1 Use of Finite State Machine Theory to Synthesize Efficient Programs (Wilkens) 6a1

We have been working on the development of a program to produce the best code assignment for the states of a program described as a finite state system. The work of consolidating space and time expenditures in the state code optimization program has been largely completed. 6a2

1.2 Formulation and Manipulation of Enumeration Based Algorithms (Paull) 6a3

We are building a system which is intended to aid a user in the formulation of a recursive definition, and then to provide him with one or more "good" algorithms that implement the definition and/or suggestions about the redesign of the definition. To produce a "good" algorithm we look for certain significant properties. The search for some of these has been automated. Recent theoretical work has resulted in the identification of additional significant properties. This work is described in TM-5 by Paull, "properties which allow Optimization of Recursive Definitions ...". 6a4

1.3 Design of Non-adaptive Comparator Algorithms (Levy) 6a5

We have implemented most of the functional components of a system for synthesizing programs to find order statistics - e.g. to find the ith largest element in a list. The design of the control and evaluation subsystems is now underway. 6a6

1.4 Automatic Program Formation from Problem Specifications (Amarel) 6a7

We are continuing to explore systems that accept a problem statement in non-procedural form and form a program by using AI methods of problem solving. In addition to problems of sorting and reasoning about actions, we are working on a problem in the area of partially ordered sets which is presented in the form of input-output correspondences. 6a8

Several approaches to the latter problem have been programmed. Amarel organized a one-day session on the "Inference of Programs from Sample Computations" at the Nato Advanced Study Institute on Computer Oriented Learning Processes, Bonas, France in August. 6a9

2. General Systems for Automatic Programming (Srinivasan) 6b

Our objective is to create a meta system that can specialize itself to be an efficient problem solver in a domain based on the descriptions of knowledge in the domain. The system is called Meta Description System (MDS). The architecture of MDS is described by Srinivasan in TR-20 "A Coherent Information System" and in TR-13 "The Meta Description System". The nature of the use of MDS for Automatic Programming is discussed in TM-4 "Programming over a Knowledge Base". A major program development effort was carried out (in INTERLISP) at the ISI TENEX and at the Stanford SUMEX-AIM System throughout the year. Substantial progress was made in the implementation of the CHECKER and the INSTANTIATOR. A novel concept of a Theorem Prover for MDS was developed; it is discussed via examples in TR-25 "A New Approach to Theorem Proving by Synthesis". This effort is now moving to the program design stage. Other activities include work on data structure optimization; TR-12 by Srinivasan covers this work. 6c

PI-Write-up, Amarel

(J24957) 20-MAR-75 14:21;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, AMAREL,NLS;7, >, 23-FEB-75 19:05 JAKE ;;;; ####;

HASKINS

NIC #24958
PART OF NIC #24980

1974 ARPA - Project Summary

Prepared for: ARPA IPTO Principal Investigators' Meeting - March 12-14, 1975

Prepared by: Franklin S. Cooper, Haskins Laboratories, 270 Crown Street, New Haven, Conn. 06510

The research has focused on syllable-oriented methods for deriving phonetic and/or lexical information from the acoustic speech signal; also stressed were the development of research tools and an effective interface of our operating system to the ARPA NET.

Speech Understanding Research

A program for syllable segmentation and manner-of-production feature analysis of speech signals has been implemented and partially evaluated. The output is a tree-structure description of the syllabic unit where the nodes point to the various constituent phonetic segments. A dictionary encoded in similar form is to be used to retrieve reference items that are likely transcription hypotheses based on acoustic-phonetic information alone. Subsequent phonetic analyses are guided by the requirement to disambiguate such hypotheses.

A series of experiments was conducted to assess human visual analysis of acoustic data. Subjects were able to match reference spectrograms to spectrogram of continuous speech with up to 65% words correct, using only visual pattern information. Agreement at phonemic segment level was better than that reported for phonetic transcription of unknown utterances, without support from semantic or syntactic processing. Experiments with isolated vowel-consonant-vowel syllables gave scores for consonant identification up to 90% and marked improvement with practice.

Software development for our PDP-11/45-/10 multiprocessor system for interactive visual examination of speech spectra has been nearly completed. The system allows direct on-line comparison of spectrograms of unknown utterances with spectrograms stored or generated on demand. The human analyst can also

use the programs to examine and correct, if necessary, the structured analysis derived from the speech.

Biologically motivated signal processing techniques have been explored with the aim of transforming speech spectra so that the information that is suppressed by known perceptual processes (such as masking) will be eliminated, and the cues that are enhanced by perceptual processes will be emphasized. Cues for automatic stress location in the speech signal have been examined and compared to human stress judgments of the same speech. The observed result is that syllable-to-syllable increases in intensity, pitch and duration all contribute to perception of increased stress, and decreases in the same increase the likelihood of stress on the previous syllable.

Network Communication Software

Software interfacing of the Network Communication Program to run under the RSX-11D operating system is practically completed. The new program will allow local multiprocessing use of the PDP-11/45 in cooperation with processes running at remote sites but controlled locally.

PI-Write-up, Woods

SPEECH UNDERSTANDING RESEARCH AT BBN
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: W. A. Woods
Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Mass. 02138

This summary covers work done during calendar year 1974 on the speech understanding project at BBN.

INTRODUCTION

The BBN speech understanding system (SPEECHLIS) is a research prototype of an intelligent speech understanding system which makes use of advanced techniques of artificial intelligence, natural language processing, and acoustical and phonological analysis and signal processing in an integrated way to understand continuous speech utterances. A central issue of the project is to gain insight into the ways in which the higher level linguistic components (vocabulary, syntax, semantics, pragmatics) interact with the acoustic-phonetic and phonological components in the overall speech understanding process and to develop techniques for making this happen efficiently. The need for support from these higher level components in order to effectively understand spoken sentences has now been well established in the literature.

The BBN effort is especially concerned with discovering techniques which will be capable of dealing with a large vocabulary, a fluent English syntax, and a diversified range of semantic concepts, rather than attempting to optimize performance for small vocabularies and restricted syntax and semantics. We are concerned with finding the limits where increased vocabulary size, increased fluency of language, and increased range of semantic diversity cannot be compensated for by increased reliability in acoustic-phonetic and phonological analysis and word verification.

PI-Write-up, Woods$

PROGRESS

During 1974, work on SPEECHLIS has consisted mainly of detailed analysis of the performance of the November 1973 system, publication of our results, experiments leading to improved capabilities in the individual components, and the design of new or improved components. We have also devoted considerable effort to the design and implementation of a second topic domain, a travel budget management system. (The November 1973 system dealt with questions about the Apollo 11 moon rocks).

Specific projects during 1974 have included the implementation of a facility for performing statistical experiments leading to improved acoustic-phonetic algorithms, a series of human parameter reading experiments (in which experimenters attempt to decipher unknown utterances on the basis of acoustic parameters mechanically extracted from the signal -- an approximation of the data which computers algorithms have available), construction of a vocabulary and semantic network for the new travel budget management domain, design and partial implementation of a new lexical retrieval component using a statistically motivated scoring strategy and a new word verification component based on an analysis-by-synthesis approach, and the beginnings of a pragmatics component including a model of the speaker's intents.

A final report describing the first phase of the project (BBN Report No. 2976, Vol. I) was issued in December of 1974 and contains a detailed description of the status of the system at that time and a list of the publications resulting from the project.

PI-Write-up, Woods

(J24959) 20-MAR-75 19:04;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, WOODS.NLS;3, >, 25-FEB-75 20:53 JAKE ;;;; ####;

JAKE 20-MAR-75 14:57 24960

PI-Write-up, Burchfiel

Distributed Computation/TENEX Project
1974 ARPA Project Summary

Prepared for: ARPA IPT Principal Investigators Conference
San Diego, Mar. 12-14, 1975

Prepared by: Jerry Burchfiel
Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Mass. 02138

Our work during 1974 has resulted in significant accomplishments in the following areas:

Prototype, multi-machine, program execution environment.

We have developed and are continuing to improve a program execution environment for ARPANET TENEX hosts that includes a networkwide file system. This execution environment makes it possible for existing application programs to operate in a context that includes the file systems of the entire collection of network machines, without requiring that the programs themselves be rewritten. The initial prototype implementation of this environment was accomplished within the context of the RSEXEC (Resource Sharing Executive) system [1] and it makes use of a mechanism for intercepting selected operating system calls made by the application programs [2].

Access Control and Accounting for Terminal IMPs

Together with the Computer Systems Division at BBN we have developed a login and accounting system for ARPANET TIPs. Prior to development of this system, anyone with a terminal and data set who knew the telephone number for a TIP dial-up port had free and uncontrolled access to the ARPANET. TIP login corrects this situation by requiring that a TIP user establish his authorization to use the network by supplying a valid network user name and password. If the user successfully passes this access control check, accounting procedures for his TIP session are initiated and he is free to use the TIP in the usual manner.

TIP login and accounting was implemented by building upon the existing TIPSER-RSEXEC system which provides a means for TIPs to support what are usually thought of as "large host" functions by sharing some of the resources of ARPANET TENEX hosts. In order to satisfy reliability requirements and to achieve a degree of load leveling, the TIPSER-RSEXEC is implemented as a distributed, multi-computer system.

PI-Write-up, Burchfiel

Management of Distributed Data Bases. 10

We have developed a method for maintaining multiple, distributed copies of a data base in the presence of distributed data base updating in a manner that guarantees the mutual consistency of all copies of the data base. Although this work was motivated by the reliability and efficiency requirements of the TIP login system which dictated that the Network user data base be maintained in this manner, we believe that the method has applicability beyond the TIP login system. The method is completely distributed in the sense that it requires no centralized control nor does it require that all copies of the data base be locked simultaneously in order to accomplish the updates. 11

The method, which is described in detail in [3], consists of two parts: a reliable, data independent, update transmission and distribution mechanism which guarantees that all data base updates reach all data base sites; and, a data dependent update procedure which is activated at data base sites when update commands arrive. The update procedure makes use of a "time stamping" scheme which enables data base sites to regenerate a sufficient portion of the time sequence of update events to determine how a particular update command should be incorporated into their copies in a consistent manner. 12

Network Protocol Research 13

We have developed a prototype implementation of the Cerf-Kahn Internetwork Protocol [4] as a TENEX facility which will support communication with hosts on the ARPANET and other networks, (e.g. Packet Radio Net). We have also developed a prototype of the new TELNET RCTE (Remote-controlled echoing and transmission) option in TENEX to provide responsive terminal connections even through satellite links. 14

Security 15

We have provided new TENEX mechanisms for creating a new encapsulated capabilities domain for either a process or an entire job: servers operate by creating a new domain for each instance of service, setting the capabilities of that domain to match the qualifications of the authenticated user. 16

Resource Allocation 17

The TENEX pie-slice scheduler is a novel approach to the resource allocation problem which removes administrative policies from the regulator mechanism. It provides a guaranteed level of CPU service to each group of users based on tables maintained by the

facility administration, and it permits sale of a "slice" of the processor to each subscribing organization.

PI-Write-up, Burchfiel

[1] Thomas, R.H., "A resource sharing executive for the ARPANET", AFIPS Conference Proceedings, Vol 42, June 1973. 19

[2] Thomas, R.H., "JSYS traps - a TENEX mechanism for encapsulation of user processes", to be presented 1975 National Computer Conference. 20

[3] Johnson, P.R. and R.H. Thomas, "The maintenance of duplicate databases", ARPA Network Working Groups RFC 677, January 1975. 21

[4] Cerf, V. and R. Kahn, "A Protocol for Packet Network Intercommunication", IEEE Transaction on Communication, May 1974. 22

JAKE 20-MAR-75 14:57 24960

PI-Write-up, Burchfiel

(J24960) 20-MAR-75 14:57;;; Title: Author(s): Elizabeth J. (Jake) Feinler/JAKE; Distribution: /ACM([INFO-ONLY]) ; Sub-Collections: SRI-ARC; Clerk: JAKE; Origin: < PI, BURCHFIEL.NLS;11, >, 23-FEB-75 00:05 JAKE ;;;; ####;