DECUS U.S. CHAPTER SIGs NEWSLETTERS

FEBRUARY 1986 | **Volume 1, Number 6**

GENERAL TABLE OF CONTENTS

SECTIONS   **PAGE NO.**

LIBRARY SECTION

HOW TO SUBMIT AN ARTICLE ... ORDER FORMS SECTION

QUESTIONNAIRE SECTION

THE SPECIAL CHARACTER SET

APL SIG

DECUS

APL SIG

Chairman (RSTS)
Larry LeBlanc
Teletype Corp.
Elk Grove, IL 60007

Library Coordinator
Susan Abercrombie
Ventrex Laboratories
Portland, ME 04103

Newsletter Editor (RT, TSX)
Douglas Bohrer
Bohrer and Company
Wilmette, IL 60091

Symposia Coordinator (RSX)
Bob Awde Jr.
General Mills Inc.
Minneapolis, MN 55427

Symposia Assistant
Bob Van Keuren
UserWare International, Inc.
Escondido, CA 92025

Standards Representative
Dan Esbensen
Touch Technologies
Escondido, CA 92025

VAX APL (Contact:)
OPEN

European Contact (TOPS)
Jean-Pierre Barasz
BARTS
75008 Paris, France

Digital Counterpart
Dave Quigley
Digital Equipment Corporation
Nashua, NH 03062

ARTIFICIAL

INTELLIGENCE

DECUS

ARTIFICIAL INTELLIGENCE SIC

Chairperson
Cheryl Jalbert
JCC
Granville, OH

Ass't Chairman
Don Rosenthal
Space Telescope Science Inst.
Baltimore, MD

Symposium Coordinator
David Slater
Institute for Defense Analysis
Alexandria, VA

Ass't Symposium Coordinator
Session Note Editor
Greg Parkinson
Cognitive Systems Inc.
New Haven, CT

Newsletter Editor
Terry Shannon
Digital Review
Boston, MA 02109

Newsletter Publisher
Bob Zeek
Pfizer Inc.
Groton, CT

Membership Coordinator
Pamela Vavra
KMS Fusion Inc.
Ann Arbor, MI

PSS Scheduling
Tom Viana

Store & Buttons
Sally Townsend
Inst. Defense Analysis
Alexandria, VA

Quality Control Chair
Dick Ciero
Harris Corp.
Palm Bay, FL

Quality Control
Carol Guyermelli

Site Coordinator, Anaheim
Chris Goodard

Volunteer Coordinator, Anaheim
Peter MacDonough
Tractor Inc.
California, MD

Members-at-Large
George Humfeld

Matt Matthews IV

Evaluation Research Corp.
King George, VA

DEC Counterpart
Art Beane
Digital Equipment Corporation
Hudson, MA

(THE (LINKED LIST))

THE NEWSLETTER OF THE DECUS ARTIFICIAL INTELLIGENCE SPECIAL INTEREST GROUP

Vol.2 No. 1   February 1986

FROM THE EDITOR

Welcome to the first 1986 issue of (THE (LINKED LIST)). The AISIG
newsletter has been absent from the past three issues of THE DECUS
SIGs NEWSLETTERS for a number of reasons, but primarily because of
my recent job change and relocation to Boston. If you have ideas,
suggestions or articles to submit to (THE (LINKED LIST)), you can
contact me at either of the addresses listed below:

Work: Digital Review
       160 State Street 6th Floor
       Boston, MA 02109
       (617) 367-7190

Home: 15 Vancouver Street #201
       Boston, MA 02115

- Terry C. Shannon

FROM THE CHAIR

Join the celebration! Come gather 'round for tales of Anaheim.

First, briefly, our history: During the Spring 1985 symposium in New
Orleans, the AI working groups of three DECUS SIGs assembled to discuss
AI's place within DECUS. The result of this meeting was positive - we
voted to begin the process of becoming a formal DECUS Special Interest
Group. During the summer, we were awarded Special Interest Committee
(SIC) status, the SIG Council's new training and testing stage for
fledgling SIGs. Beginning in September, before we'd even finished
clarifying our SIC status, we began to get responses to (The (Linked List)).
By mid-October, we knew we had a good core leadership group and we'd adopted
SIC/SIG operating procedures. Sessions and PSSes were scheduled for
Anaheim. We'd begun to collect session notes. Just for fun, we'd chosen a
mascot and even had an artist's drawing of our new friend. Before we'd
left New Orleans we'd started scrambling to build a good SIG. The payoff
came in Anaheim.

The session notes were an early clue that the Anaheim symposium was going
to be a great week for us. Pam Vavra, Terry Shannon, and I had badgered
speakers all fall for their session notes. Terry had written his notes and
an important addition. Pam had formatted all the notes and designed the
cover. We made the "bold" decision to produce 300 copies. Chris had them
duplicated and bound. One look told me we had a winner, and the contents
were even better. They sold out rapidly. We'll make 400 copies of the
Dallas session notes.

Our mascot is a creature with an appealing smile called an A-i, after the

AI-1

only sound that it makes. More information on the A-i (Bradypus
Tridactylus to purists) will be presented in an upcoming issue of this
newsletter; in the meantime you can refer to the cover story in the
January 1986 edition of International Wildlife magazine.

Sally Townsend, our "shopkeeper," made a variety of clever buttons using
the A-i. They were in more limited supply than our session notes, but we'll
try to have more for Dallas.

On Saturday December 7, before the Fall 1985 symposium got underway,
DEC counterpart Art Beane and I opened the AISIC suite and sat there
talking and planning while we waited for Art's promised AI VAXstation
to materialize. During the wait, we accepted delivery of the hardware
earmarked for our "suitemates" - APL, Languages and Tools and UNISIG.
Our shared facilities provided us with a large parlor and a separate
"machine room" and proved to be an immensely successful combination.
We'll share again in Dallas.

The suite was open, generally, from 8:30 in the morning until well after
midnight. We had numerous meetings there, and, once the AI VAXstation
arrived, nonstop demos and software discussions. Our speakers were
available to answer questions and debate discuss issues, as were
representatives from DEC's AITC (Artificial Intelligence Technical
Center) and related areas. We collected new SIC members, drafted
new leadership and even did our sewing in the suite when the A-i
patches for our polo shirts finally arrived. (Who needs an alligator?)

Sunday, we offered three PSSes: Introduction to AI, LISP, and OPS5. I
visited all three and was pleased with the comments that I heard. For
Dallas, by popular request, we will offer these three plus one that serves
as a follow-on to the introduction. Complete details on each PSS will
be provided in the March issue of (THE (LINKED LIST)).

All week, we had a very positive reception for our sessions. We had to
close the doors on three of them because the fire marshall didn't seem to
approve of people hanging from the chandaliers to hear our speakers. We
repeated those three and had sellout crowds for the encore performances.
I hope that we will be able to share content from a number of the sessions
with you through this newsletter.

By Wednesday, we were definitely receiving rave reviews. Wednesday morning,
I met with our mentor and review committee. They had been appointed by the
SIG Council to watch over us for the duration of our SIC status. Their
input has been very important to our success. SIC status had been expected
to last for up to three symposia. Exactly one symposium after we voted to
request SIC status, Bill Brindley, Chairman of our Review Committee,
recommended that SIG Council schedule a vote on our SIG status.

Friday morning, I left the SIG Council meeting and arrived at the
conclusion of Terry Shannon's last session in time to begin the closing
session. My announcement that SIG Council had voted that we shall have SIG
status was met by cheers. Other DECUS units may yet vote on our status,
but when and which is unclear. Since our position is strong, I do not
anticipate any problems.

The rest of Friday was a similar joy. We had a good discussion in the
closing session about what things people want to see added to what we're
doing, and I took the opportunity to offer some much deserved honors.

Sally Townsend and I had conspired to produce a special thank you. Sally

AI-2

had buttons made featuring our mascot and the words "A-I Honors, Anaheim 85." I'd asked for three. Having come up with this concept, I had given myself the difficult job of allocating them. Pam Vavra, our new Symposia Rep, who has assisted with many of the pieces of getting us started, Chris Goddard, our new Membership Coordinator and Suite Coordinator for Anaheim, who has also picked up pieces as necessaary, and Don Rosenthal, Vice Chair, who has done many important things for the SIG, each deserve special recognition. Every member of the Steering Committee deserves great thanks for a job well done.

The buttons I saved for the people who served as lightning rods for the formation of the group. Dave Slater has been Chair of the AI Working Group of the VAX SIG since the Spring 1984 Symposium in Cincinatti, and has worked to generate interest in AI within DECUS. Terry Shannon began writing (The (Linked List)) before we even became a SIC. He has also given some of our most successful sessions and tirelessly offers more and more. Art Beane was in Cincinatti, too. Art offered the sessions that got everything started and has continued to offer sessions and respond to our calls for assistance. Art has already made significant contributions as our DEC Counterpart and as a part of our Steering Committee. It was to these last three that I gave A-i Honors this time. It is my goal to offer one A-i Honors in Dallas.

After the Closing, Chris ran for the ice cream cake that he'd ordered with our A-i on top. Everyone from the Closing was invited to the suite and we had a party. That evening, L&T and UNISIG helped us put on another party. Our mentor, Larry Jasmann, came to this one and I presented him with one of our A-i shirts as a way of saying thank you for all the guidance.

I've glossed over the technical content here to share the news with you. We're off to an exciting start and we have a great group. However, it takes a number of people to do a really good job of running a SIG. We need more input; and, as we grow, we will need even more. Write Chris or I if you would like to participate as a member or as part of the leadership. Join us in Dallas and share your views with us through the newsletter.

- Cheryl Jalbert

FEATURE ARTICLE

LISP and OPS5 have received considerable attention within our new SIG, so we felt we should give equal time to PROLOG adherents. The following article gives a brief, novice level introduction to PROLOG.

PROLOG: PROGRAMMING IN LOGIC

Until about 10 years ago, LISP, John McCarthy's 1957-vintage LISt Processing language was the only significant symbolic data manipulation language available for AI or knowledge-based applications. 1972 brought the first incarnation of PROLOG, a nonalgorithmic, logic-oriented programming language that dispenses with DO-FOR, WHILE and GOTO and implements simplified rules of predicate calculus. This article examines the history of PROLOG, discusses what it is and how it works, suggests why the Japanese chose PROLOG for their Fifth Generation project and analyzes the great LISP vs. PROLOG debate.

A BRIEF HISTORY OF PROLOG

PROLOG's roots can be traced to the University of Montreal, where a group of computer scientists was formed in 1965 to develop a French to English language translation program. Some five years later, a Frenchman named Alain Colmerauer joined the Canadian researchers. Although their efforts were not entirely successful, the translation program represented the first large scale use of symbolic logic to manipulate data. When the translation project drew to an abortive conclusion, Colmerauer returned to the University of Marseilles, taking with him the symbolic data manipulation and logic programming techniques he had developed and used in Canada. He referred to this collection of programming techniques as "System Q," and used it as a springboard to the development of a new computer language.

By 1972, System Q had evolved into a logic oriented programming language and interpreter which Colmerauer's wife named PROLOG, an acronym for PROgramming in LOGic. Implemented in ALGOL-W by the French Groupe D' Intelligence Artificialle, this early PROLOG remained in the domain of academia. In its primitive versions, PROLOG lacked the capability to perform arithmetic calculations and was supported with very few diagnostics. Little more was done with the language until 1974 when Robert Kowalski, a researcher at the University of London's Imperial College, formalized the concepts of logic programming and created a "reason" for PROLOG.

Shortly thereafter, personnel at the Department of Artificial Intelligence in Edinburgh, Scotland discovered the new language and began to use and improve upon it. Under the supervision of Dr. David Warren, the Edinburgh team added diagnostics, arithmetic capability and an improved user interface to PROLOG. Their efforts resulted in a specification for the language, often referred to as "Edinburgh syntax," and ultimately in the development of the fastest and most efficient PROLOG implementation.

Until the end of the decade, interest in PROLOG was concentrated in the European AI community at locations like Edinburgh, London and Marseilles as well as at the Institute for Coordination of Computer Techniques, or SZKI, in Budapest, Hungary. Hungarian computer scientists recognized the value of PROLOG for business and industrial as well as AI applications, so the language gained a significant following in that country. Outside of Europe, the Electrotechnical Laboratory in Tokyo was one of the few research centers to show interest in PROLOG. The involvement of this facility undoubtedly had an impact on the Japanese decision to make PROLOG the language of choice for their Fifth Generation computer program.

PROGRAMMING IN PROLOG

On an elementary level, a PROLOG program consists of problem description instead of the steps that must be taken to solve a problem. Using a conventional computer language, a programmer defines the algorithms necessary to solve a problem, and the sequence in which the algorithms are executed. A PROLOG programmer writes a program by defining the problem at hand in terms of facts, relationships and rules, allowing the PROLOG interpreter to define the problem-solving algorithm. In essence, what the programmer does is make assertions

(which tell PROLOG what is true) and allow the interpreter to draw its own conclusions.

Programming in PROLOG is a three step process in which a programmer:

1. Defines facts about objects and their relationships

2. Defines the rules that apply to the objects and their relationships

3. Asks questions about the objects and their relationships.

PROLOG uses the concepts of predicate calculus to represent and manipulate information or knowledge. Objects and the relationships between them, are represented as clauses that either may be true or false, then evaluated by an inference mechanism that relies on logic. Through its control mechanism, PROLOG can infer new facts from the ones with which you have provided it.

FACTS

Facts are represented in PROLOG by clauses whose grammar and syntax are similar to the rules that govern the English language. For example, the fact that a person named Michael lives in a town called Solvay may be expressed in English as:

Michael lives in Solvay.

The same fact, expressed as a PROLOG clause, takes the form:

```
lives_in(michael,solvay).
 ^        ^       ^
 |        |       |..... argument 2
 |        |............. argument 1
 |..................... predicate
```

When facts are entered into a PROLOG database, the name of each predicate and object must begin with a lower case letter. The predicate is written first, followed by objects which are separated by commas and enclosed in parentheses. Finally, each fact must be terminated with a period.

If a PROLOG database contains a single clause, the interpreter can only parrot back information or answer a question based on the facts contained in that clause. Additional facts and rules must be asserted before the interpreter can deal with more complicated questions. Anything that can't be proven true is considered false by PROLOG - the interpreter will respond "no" to any question that isn't supported by an assertion in the program's database, whether or not the answer actually is false.

VARIABLES

PROLOG uses variables to represent those classes of objects which have not been or can not be named explicitly. In the English language, we use terms like "someone," "everything" and "all" to represent variable data. In PROLOG, symbols that begin with a capital letter

serve the same purpose. When the interpreter is asked a question that contains a variable, it searches through the program database to find an object that matches the question. This process, known as "pattern matching," is one of the most powerful features of PROLOG.

For example, if you wanted to assert the fact that Michael likes a certain class of objects without specifying every conceivable object within that class, you could do so by defining the variable "X" as something that Michael likes. Each object equated with the variable "X" would then be recognized by PROLOG as something that Michael likes, as follows:

```
likes(michael,X).
females(X).
golf(X).
steak(X).
richard(X).
cats(X).
```

A SIMPLE EXAMPLE

The following example demonstrates how facts, rules, and questions are used to create, qualify, and query a PROLOG database.

The database listed below asserts several facts in PROLOG notation. In English, these facts are:

Two individuals named Michael and Erin like cats;
Two individuals named Amanda and Erin are female;
Amanda and Erin live with Michael;
and Amanda is younger than Erin.

In PROLOG notation, the same facts are expressed as:
```
likes(michael,cats).
female(amanda).
female(erin).
likes(erin,cats).
lives_with(michael,amanda).
lives_with(michael,erin).
younger_than(erin,amanda).
```

To ask PROLOG what Michael likes, you could enter:

?- likes(michael,X).

PROLOG will search through its database and attempt to find facts or rules that match the question. When it finds the fact "likes(michael,cats)," it responds with:

X=cats.

This example merely returns a fact that has been established by a clause in the database. By adding rules to the database, it's possible to utilize PROLOG's inference mechanism to perform more sophisticated queries.

PROLOG rules can be used to express general statements about objects and their relationships. Each rule consists of two parts, the head and the body, which are separated by the symbol ":-." To assert the fact that Michael likes any female who likes cats, you could enter the following rule into the program database:

likes(michael,X):- female(X), likes(X,cats).

This rule states that Michael likes any female who likes cats. If this rule is added to the database and PROLOG is asked who Michael likes, the interpreter searches for a female who likes cats:

?- likes(michael,X).

Instead of responding with a single fact as it did in the previous example, PROLOG attempts to answer the question by initiating a depth-first search. The interpreter starts at the beginning of the database and begins to evaluate its facts sequentially. When the interpreter finds an object in the database that matches the variable "female(X)," the first goal is satisfied or instantiated. (In this case, the fact "female(amanda)" matches the variable.) PROLOG notes the goal's location with a place marker and continues its depth-first search by scanning subsequent facts in the database in an attempt to satisfy the second goal. If the interpreter is able to match the variable "likes(X,cats)" with an object in the database, the second goal is satisfied and an answer is displayed.

In this example, the second goal cannot be satisfied. Although Amanda is female, the database contains no information to prove the assertion that Amanda likes cats. At this point, PROLOG will backtrack and attempt to resatisfy the first goal by searching the database for a second object that matches the variable "female(X)." During its initial search, PROLOG started at the beginning of the database. For each subsequent attempt, the interpreter starts its search from the location of the first satisfied goal.

Thus, PROLOG backtracks to the fact directly beneath "female(amanda)" and searches for a second occurrence of "female(X)." When the interpreter finds "female(erin)," it assigns a place marker to this fact and again attempts to satisfy the second goal. The fact "likes(erin,cats)" instantiates the second goal of the rule, so PROLOG answers the question by displaying:

X=erin.

Nowhere in the database is it stated explicitly that Michael likes Erin. Instead, PROLOG is able to arrive at this conclusion by drawing an inference from the facts that Erin is female and Erin likes cats.

Adding A Second Rule

Assume that Michael has a younger sister named Amanda. Based on the absence of supporting information, PROLOG would respond "no" to a query about Michael's siblings, as follows:

?- sister_of(michael,X). no

PROLOG cannot prove this assertion because its database does not contain a fact that matches the question "Who is a sister of Michael?"

However, we could define a sister of Michael as "a female who lives with Michael and who is younger than Erin" by adding the rule listed below to the PROLOG database:

sister_of(michael,X):-
female(X),lives_with(michael,X),younger_than(erin,X).

If we were to pose the "sister_of" question to the interpreter again, it would instantiate all three variables in the rule with the object "amanda," and respond with:

X=amanda.

Again, the database does not contain a fact or rule that tells PROLOG that Amanda is Michael's sister. As in the previous example, PROLOG uses the rules of logic to infer this fact from the other facts that have been stated.

PROLOG SEMANTICS

PROLOG supports two distinct programming styles: declarative and procedural. Procedural programming requires you to define and code a specific algorithm that explicitly directs the computer to perform each step necessary to the solution of a problem. This is the method used to write programs in traditional programming languages. Declarative programming works very differently. Instead of telling the computer how to solve a problem, you define the problem to the computer in terms of logical facts and relationships, and allow the computer to come up with a solution.

There are two reasons for the inclusion of a procedural component in PROLOG. First, input and output operations require the use of techniques that are not based in logic. Therefore, the code responsible for implementing I/O in PROLOG is of necessity procedural. Second, a program composed entirely of declarative statements would contain a knowledge base filled with statements about what is true. PROLOG's inference strategy could make deductions based on these statements, but it would do so very inefficiently. The availability of procedural features in PROLOG enables a programmer to tell the interpreter how to deal with the information in the knowledge base and in what order to resolve clauses or declarative statements.

APPROPRIATE APPLICATIONS

Although originally developed for language translation purposes, the characteristics of PROLOG make it suitable for a variety of applications, including natural language processing, relational databases, parallel processing, and expert system development.

o Natural Language Processing

PROLOG's syntax allows the rules of a natural language grammar to be expressed as English language statements, making it suitable for writing natural language processors or front ends. PROLOG automatically will parse an English language sentence and apply the appropriate rules to represent the meaning of the sentence.

No other programming language shares this capability with PROLOG; for example, a FORTRAN programmer would have to develop and write a parser and analyzer to accomplish what PROLOG does by default.

o Relational Databases

A PROLOG program is analogous to a relational database in that it is made up of facts and rules which describe the relationships between facts. In many cases, a relational database can be implemented in PROLOG without the need to include a query facility to access the database and return information from it. The readability and English-like syntax of PROLOG statements often precludes the need to develop a front end that converts English requests into a form that is acceptable to the database.

o Parallel Processing

PROLOG does not use the assignment statement, one of the stumbling blocks to parallel processing. Conventional programming languages use assignment statements to establish values and store them in specific memory locations. If two independent programs attempted to reference or modify the same memory location at the same time, a system error or crash would result. Because PROLOG dispenses with the assignment statement, it lends itself to parallel processing applications.

o Expert System Implementation

Finally, PROLOG is particularly appropriate for expert system implementation because it is able to explain the logical steps it went through to derive an answer to a problem. This built-in capability eliminates the need to design and write an explanation facility for a PROLOG-based expert system.

ORIENTAL INFLUENCE

PROLOG owes much of its recent rise in popularity to the fact that it has been designated as the kernel language for the Japanese Fifth Generation computer project. Japan based its decision on the fact that PROLOG is well suited to symbolic data manipulation and does not require the serial architecture of a Von Neumann computer. These capabilities make PROLOG a natural choice for the parallel architectures of Fifth Generation knowledge information processors.

However, the use of PROLOG as a kernel language should not be regarded as a Japanese endorsement of PROLOG as the standard programming language of the Fifth Generation. Contrary to the claims of several vendors and trade journal articles, the Japanese intend to use PROLOG as a knowledge representation and manipulation paradigm, not as the specific language in which knowledge-based programs will be written.

Perhaps taking a cue from our government's choice of ADA as the name for its standardized higher level programming language, the Japanese have dubbed their version of PROLOG "Himiko," the name of a famous woman in Japanese history. This implementation was used to write the operating system for the personal sequential inference (PSI) machine developed during the first phase of the Fifth Generation, and an

enhanced version will be used to control the parallel inference machines now under development.

PROLOG vs. LISP

LISP is an American language. It was invented and enhanced in this country and its various dialects and implementations have attracted a great following among American AI researchers. Several American firms manufacture "LISP machines," processors that are optimized for LISP programming and execution.   Because PROLOG was developed and refined outside of the United States, it often is viewed as a foreign, unproven language, especially by LISP adherents. Nationalism aside, almost anything that can be done in LISP can be done equally well in PROLOG, and vice versa. The best language for a given application depends primarily on the specific needs of that application.

One of the arguments against PROLOG is based on its apparent ease of use. In order to perform even basic operations in LISP, you must have a reasonably good understanding of the language and how it manipulates information. On a superficial level, PROLOG is much easier to work with. Its built-in inference strategy enables even a novice to do impressive things with very little effort or presupposed knowledge. After some initial investigation, you'll discover that PROLOG's inference mechanism gives no indication of how it accomplishes what it does. When you begin to use PROLOG to develop real world applications, you'll find that a familiarity with the internal workings of the language is essential.

The present unavailability of "PROLOG machines," custom processors designed for efficient PROLOG execution, is often cited as a drawback by LISP devotees. Although several prototype PSI machines have been developed as a result of the Japanese Fifth Generation program, the high performance LISP machines from companies like LMI, Symbolics and Xerox currently have no PROLOG equivalents. However, this situation may change in the near future. A fast, efficient PROLOG interpreter for LISP Machine Inc.'s Lambda processor has been written by AI researchers at the University of Uppsala in Sweden, and a VLSI-based PROLOG compiler is under development at Syracuse University in Syracuse, New York.

Standardization also must be considered. Although numerous dialects of LISP are available, an industry-wide effort to develop a de facto LISP has resulted in the adoption of Common LISP. No formal industry-wide PROLOG standard has yet been established; there are compatibility problems that must be resolved before PROLOG can enjoy widespread use as a general purpose, transportable programming language.

Finally, the backtracking strategy inherent in PROLOG's control mechanism allows for error. As discussed earlier, if a rule selected by the program does not lead to a conclusion, the program backtracks to a decision point and selects an alternative rule. Reliance on this single control mechanism makes it difficult to limit searches by determining the most likely paths to pursue. Although statements in a PROLOG program can occur in any sequence because the interpreter evaluates each statement independently, the depth-first search strategy makes an unstructured PROLOG program easy prey to the combinatorial explosion. The most efficient way to organize a PROLOG program is to structure it so that specific searches are conducted prior to generalized searches.

WHAT TO LOOK FOR IN A PROLOG INTERPRETER

PROLOG interpreters tailored to a variety of computers and operating systems are available at prices ranging from less than fifty dollars to several thousand dollars. If your primary goal is to learn more about the language or experiment with its capabilities, a PC-based PROLOG interpreter should satisfy your needs at a very moderate cost. If you plan to use PROLOG for real world applications, you'll need the facilities offered by more powerful versions of the language. These include a compiler for increased speed and performance, a built-in editor, and virtual memory management capabilities.

Although a formal PROLOG standard remains an elusive goal, most experts consider Edinburgh Syntax to be a de facto standard for the language. Edinburgh Syntax is defined by the DEC-10 PROLOG compiler and interpreter developed by AI researchers at the University of Edinburgh and is described fully in "Programming In PROLOG" by W.F. Clocksin and C.S. Mellish. A compatible superset of Edinburgh Syntax is likely to form the core of a future PROLOG standard, so it's important that the PROLOG you select conforms with the conventions set forth in this book.

EPILOG

PROLOG is an efficient, powerful language that's well suited to logic programming applications. The built-in inference strategy of the language makes it fun to work with, and its logical basis ensures that you'll learn the principles of predicate logic while you learn how to manipulate symbols instead of numbers. Furthermore, PROLOG's unique nonalgorithmic approach to problem solving will be a key element of programming languages and techniques of the future, making it worth your while to obtain a basic understanding of the language and how it works. PROLOG may not be a programming panacea, but it provides a good environment for database manipulation, expert system prototyping, natural language processing and other aspects of knowledge-based programming. If you want to learn more about symbolic programming and the concepts behind expert systems and AI applications, gaining some experience with PROLOG is a good way to start.

<-o->

OPS5 NEWS

WHO'S IN CONTROL HERE?

By Don Rosenthal, AISIC Vice Chair

One of the biggest points of confusion for new OPS programmers is the use of the built-in control strategies. Henry Ford once offered autos for sale in "any color a customer desired, as long as it was black." OPS is not quite as restrictive in the choice of strategies, it offers two: Lex and Mea. However Lex can be accurately considered a subset

of Mea.

What are control strategies? Let's answer this question before tackling the subtleties of Lex and Mea. An important characteristic of OPS5 and its direct predecessors is that it's a pure "object-level" (or "domain-level") programming language. This is considered by many to be its most attractive feature; others spell that adjective a bit differently, however. What this means is that there are no control constructs in the language: no loops, no branches, not even an implied sequencing of its basic structure - the rule.

The intent of a control-free language is to allow the programmer to deal solely with solving the problem at hand, in this case, to write rules that pertain only to the solution of the problem, with no programming overhead. The programmer writes the rules, and the "most appropriate" rule at any point during the execution of the program selects itself and fires. Obviously, there must be some mechanism for this selection, but it is imbedded within the OPS5 interpreter. This mechanism is called the control strategy. Its operation is successfully hidden from the user, with the single exception that a choice can be made between Lex and Mea. To make an intelligent choice as to which strategy to use involves understanding the selection mechanism, but it's really not as obscure as some people make it out to be (I promise...). Here goes:

o Important fact #1:

OPS5 operates in a small loop, termed variously the "Recognize-Act" cycle, or the "Match-Select-Execute" loop.

o Important fact #2:

During the match phase of the cycle, ALL rules whose IF clauses successfully match the present state of the data (which in "working memory") are copied into the "conflict set".

o Important fact #3:

When the conflict set has collected all the successfully matched rules, one is selected by the chosen control strategy. If the conflict set is empty, however, the program halts.

o Important fact #4:

During the execute phase, the THEN clauses of the selected rule are executed. These usually modify working memory so that when the match phase is re-entered, a new conflict set is generated.

Let's now investigate the selection phase more closely. The conflict set is made up of instantiated rules. These instantiations have the following structure. First comes the name of the rule. Then the time-tag of the working memory element ("wme") matched by each IF clause. What's a time-tag? It's really simple, although the name is a bit misleading. When a piece of data enters working memory, it is assigned a number. These numbers are assigned consecutively and in increasing order. Thus, a wme with a higher number than

another wme has entered working memory more recently. When a working memory element is removed from working memory, its time-tag is retired. The only trickiness is that when a wme is modified, it is given a new time tag (it actually is a "new" piece of data after all).

It is the time-tags that form provide the most important information for the selection process, but I'm getting ahead of myself. Let's look at Lex in detail. It has only 4 steps.

Step #1: Remove any instantiated rules that have fired before.

Rules can match many combinations of working memory elements, but any exact combination of matches may only be fired once. Realize that the "most appropriate" instantiation might easily become a one-rule infinite loop otherwise.

Step #2: Find the instantiation that matches the most recent set of working memory elements.

This is a simple matter of comparing time-tags. If any IF clause of any single instantiation matches a working memory element with a time tag higher than that of any other instantiation, that rule is selected, and we need not bother with the following steps.

What if there is a tie? If more than one instantiation matches to the highest time-tag of any rule in the conflict set, consider only that subset of rules, and look at their next most recent time-tag. If any one instantiation in this subset has a second most recent time tag greater than the second most recent time tag of all others, select that instantiation, else, continue to the third, fourth, etc, most recent time tag as needed.

Step #3: If any instantiations were tied all the way through step #2, make a decision based on how specific the IF clauses of those instantiations are.

Rules can match the same working memory elements, but in different ways. That is, an IF clause of one rule might look for a disease symptom, such as "fever" regardless of how high the fever was, and another might look for fever greater than 100 degrees. They both would match a working memory element representing a fever of 102 degrees, but the latter clause would be considered more specific.

Step #4: Finally, if no there were no tie-breakers due to recency or specificity, arbitrarily choose one of the tied instantiations.

Lex is a subset of Mea, as I said above, and Mea adds only one more step, between #1 and #2 above:

Mea Step #1.5: Look only at the time-tag of the working memory element matching the first clause of each instantiation.

If one is more recent than all the others, select that instantiation. Otherwise, continue as in Lex.

With that under our belts, the only point left to explore is when to use Mea and when to use Lex. Some would tell you, "Always use Mea". Not I. Stay tuned for next month's topic, "The Great Lex and Mea Controversy", or, One Man against the World."

<-o->

BOOK REVIEW

Programming Expert Systems In OPS5:
An Introduction To Rule-Based Programming

By Brownston, Farrell, Kant and Martin
Addison-Wesley Publishing, 1985
471 pages

Writing a book about OPS5 is not as easy as it might appear. A familiarity with classical programming techniques is usually all the background a programmer needs to learn a new programming language such as C or PASCAL. The underlying philosophy of subroutines, function calls and sequential execution is already established, so all the newcomer has to do is master the syntax of the new language.

Such is not the case with OPS5, a rule-based programming language that requires a new approach to programming and problem solving. As OPS5 novices soon discover, the language requires a different mindset: merely learning the syntax of OPS5 does not make you a capable production system programmer. You must understand how to develop rule-based systems and how to execute rule-based programs; activities that require you to think in terms of OPS5 rules or productions. Thus, the subtitle of the book is critical to the mastery of OPS5.

Realizing that most readers have only classical programming backgrounds, the authors first describe the basic philosophy of production system programming, then explain the ins and outs of the OPS5 language itself. The book's organization and structure make it an appropriate academic textbook, even though the exercises at the end of each chapter seem to have been added as an afterthought. The text is conveniently divided into three parts. Part One is a concise view of the nature of production systems along with a discussion of when to use them. Although the authors write in a clear and understandable style, the complexity of the subject is such that you may have to read this section several times before you have a firm grasp of the material it presents.

Part Two describes the use and syntax of OPS5 in great detail. This section includes a complete programming example and notes on programming style. Program development and organization are discussed at length, and an entire chapter is devoted to advanced OPS5 programming techniques. On the debit side, I was surprised to see that the section provided very little information about program maintenance. A production system requires non-standard maintenance techniques and this book should have covered them thoroughly.

The last section of the book discusses popular production system philosophies and provides an account of the various production system architectures and common system features. The last chapter of the book is devoted to an overview of about a dozen expert system tools related to production system tools.

Although it isn't a treatise on artificial intelligence, the book uses an in-depth study of OPS5 to present a general view of a specific area of AI research, making it more than a typical programmer's guide. I

struggled with the anemic OPS5 documentation supplied by DEC for about
two months before I got an advance copy of the first five chapters of
this textbook, and reading this material was like turning on a light
in a dark room. Despite one minor reservation about the chapter
exercises, I feel that this book is a must for OPS5 programmers.

- Reviewed by Chris Goddard, DECUS AI SIG

That's it for the February issue of (THE (LINKED LIST)). Our March
issue will feature a review of the Fall 1985 Symposium, a preview of
AISIG sessions and activities scheduled for the Spring 1986 Symposium,
a review of Golden Common LISP, and information on our four presymposium
seminars.

BUSINESS

APPLICATIONS

BUSINESS APPLICATIONS SIC STEERING COMMITTEE

Chairman
Stuart Lewis
Douglas Furniture
Bedford Park, IL

Symposium Coordinator
Steve Simek
IRT Corporation
San Diego, CA

Asst. Symposium Coordinator
Bobbie Wiley
CEI Perry Nuclear Power Plant
Euclid, OH

LRP and Marketing Coordinator
Arnold I. Epstein
D-M Computer Consultants
Rolling Meadows, IL

Marketing Asst.
George Dyer
Gallaudet College
Washington, DC

Communications Representative
OPEN

Newsletter Editor
Thomas Byrne
L. Karp and Sons
Elk Grove Village, IL

Session Notes Editor
Raymond Swartz
Goodyear Tire and Rubber Co.
Akron, OH

Library Representative
David Hittner
Projects Unlimited
Dayton, OH

CL SIG Liaison
Becky Burkes
Financial Insurance Consultants
Covington, LA

DMS SIG Liaison
Joe Sciuto
Army Research Institute
Alexandria, VA

Members-at-Large
Robert D. Lazenby
Dixie Beer Dist., Inc.
Louisville, KY

Robert Kayne
Gallaudet College
Washington, DC

Ray Evanson
Paragon Data Systems
Winona, MN

Digital Counterparts
Sue Yarger
Digital Equipment Corporation
Merrimack, NH

Ray Arsenault
Digital Equipment Corporation
Merrimack, NH

SIC Mentor
Bill Brindley
Networks SIG Chair

SIC Review Committee
Larry Jasmann
Leslie Maltz
Ted Bear
Jeff Killeen

CL SIG

SEAL of APPROVAL

Herschal

COMMERCIAL LANGUAGES SIG

Chairman
Jim Wilson
Pfizer Inc. QC Div.
Terre Haute, IN

Symposium Coordinator
Ray Strackbein
Palm Desert, CA

Library Coordinator
Philip Hunt
System Industries
Milpitas, CA

Session Note Editor
Bob Van Keuren
Userware International, Inc.
Escondido, CA

Newsletter Editor
Ted A. Bear
Ramtek
Santa Clara, CA

Ass't Newsletter Editors
Beverly Welborne
LaPorte, IN

Kevin Cullen
VITA-Mix Corp.
Holmstead Falls, OH

Daniel Cook
Userware International, Inc.
Escondido, CA

Basic Working Group Members
Mark Hartman
Jadtec Computer Group
Orange, CA

Rocky Hayden
UserWare International Inc.
Escondido, CA

Bill Tabor
Computer Productss
Pompano Beach, FL

Cobol Working Group Members
Keith Batzel
Crowe, Chizek & Co.
South Bend, IN

Mary Anne Feerick
Greyhound Temporary Personnel
Lyndhurst, NJ

Bill Leroy
The Software House, Inc.
Atlanta, GA

Herbert J. Matthews IV
ManTech International Corp.
Alexandria, VA

Kenneth Shay
Melvin Simon & Associates
Indianapolis, IN

Jim Welborne
Crowe Chizek & Co.
South Bend, IN

DIBOL Working Group Members
Neil Baldridge
CompuShare
Lubbock, TX

Becky Burkes
Financial Insurance Consultant
Covington, LA

Colin Chambers
Software Ireland Rep. Inc.
Portola Valley, CA

Mark Derrick
WAAY-TV
Huntsville, AL

Gary A.P. Kohls
Milwaukee, WI

Ken Lidster
Disc
Sacramento, CA

Kenneth M. Schilling
MCBA
Montrose, CA

Marty Schultz
Omtool Inc.
Tewksbury, MA

David L. Wyse
Professional Business Software
Dayton, OH

Marty Zergiebel
The Software Gallery
Brookfield, CT

RPG Working Group Members
Keith Batzel
Crowe Chizek & Co.
South Bend, IN

Ted Bear
Ramtek
Santa Clara, CA

Digital Counterparts
Tom Harris
Nashua, NH

Jim Totten
Nashua, NH

Joe Mulvey
Nashua, NH

Shirley Ann Stern
Nashua, NH

Standards Representatives
BASIC
Dan Esbensen
Touch Technologies, Inc.
Escondido, CA

COBOL
Bruce Gaarder
Macalester College
St. Paul, MN

DIBOL
Eli Szklanka
TEC
Newton, MA

M·D·Hubbard Spring Company, Inc.

MANUFACTURERS OF

COILED AND FLAT SPRINGS
SPRING WASHERS   WELCH EXPANSION PLUGS
WIRE FORMS   STAMPINGS

ESTABLISHED 1905

313 Δ 628-2528

BOX 425
OXFORD, MICHIGAN
48051

Nov. 19, 1985

Mr. Ted Bear
c/o RamTec
2211 Lawson Ln.
Santa Clara, CA 95050

Dear Ted,

Would you please ask your membership if anyone has access to
a reference manual for:

          C O S 300 Multi Terminal System (Mts)

                          or

          Multi Terminal Dibol

For use on a PDP-8/E or DDS 340, etc. We have the software
but need the manual.

Thank you for your help in this matter.

                          Very truly yours,

                          M. D. HUBBARD SPRING CO., INC.

                          C. D. Hubbard

3/sw
hf

DECUS
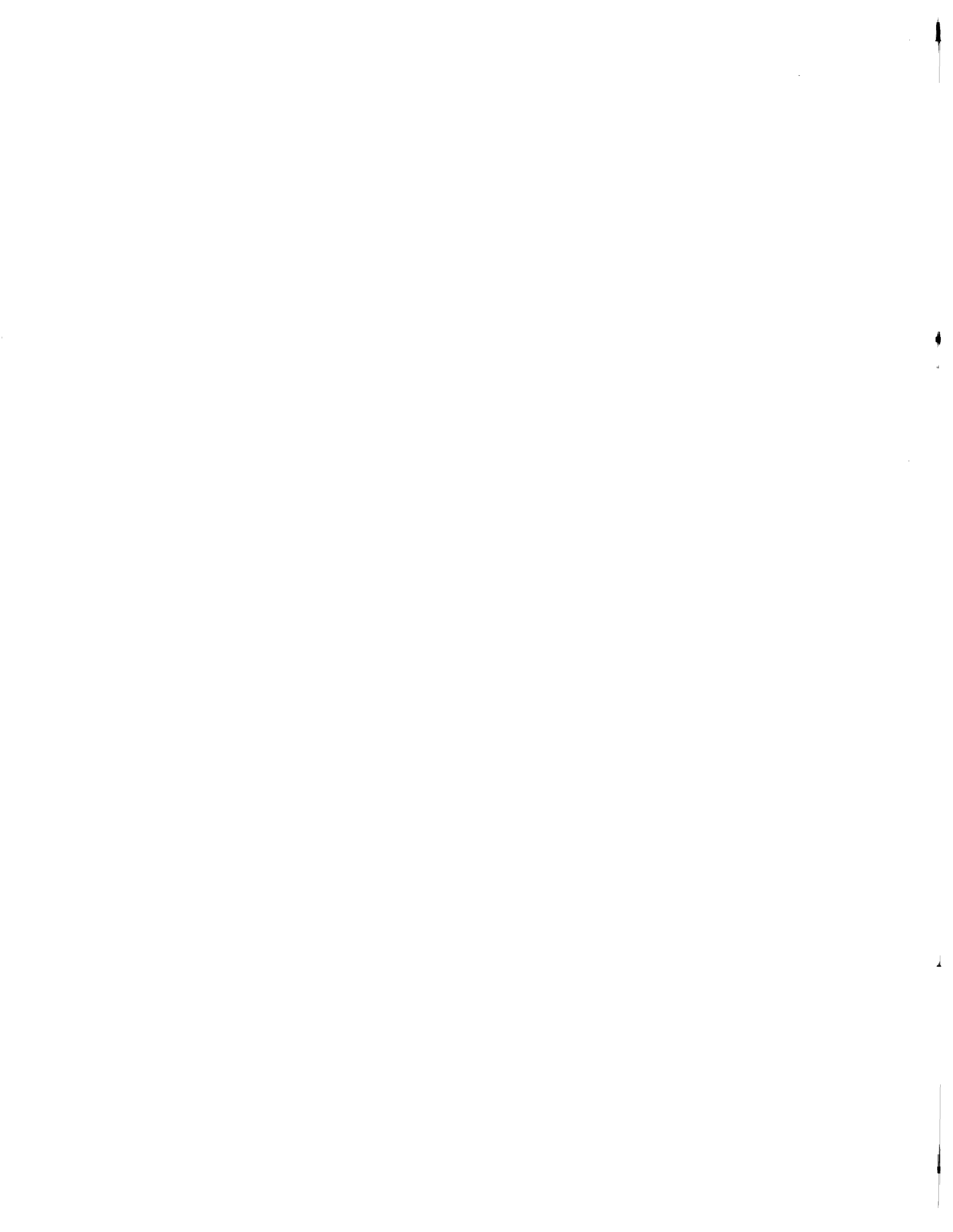
D A A R C

DAARC

Chairman
James Deck
Inland Steel Research Lab
East Chicago, IN

Symposium Coordinator
Mack Overton
FDA
Chicago, IL

Newsletter Editor
Ellen Reilly
William H. Rorer
Ft. Washington, PA

DEC Counterpart
Nancy Kilty
Digital Equipment Corporation
Marlboro, MA

Hardware & Interfacing
Peter Clout
Los Alamos National Lab
Los Alamos, NM

Math Statistics & Analysis
Herbert J. Gould
C.C.F.A. University of Illinois Med Center
Chicago, IL

Process Control - Industrial Automation
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

RS-1
George Winkler
CPC International
Argo IL

WINTER DOLDRUMS

THIS IS A GOOD TIME OF THE YEAR TO TAKE OUT PAST ISSUES

OF THE DECUS NEWSLETTER AND REALLY READ THE ARTICLES.

IN THE PAST YOU MAY HAVE JUST READ THE INFORMATION

FROM YOUR PARTICULAR SIG. IT'S NOW A GOOD CHANCE

TO PICK UP ON ALL THOSE LITTLE INSIGHTS THAT SOMEONE

TOOK THE TIME TO SHARE WITH OTHERS. THE NEXT PAGE HAS

ANOTHER BIT OF INFORMATION THAT COULD BE USEFUL TO YOU.

HAPPY READING!

DAARC EDITOR

At DECUS in Anaheim, I learned some things about MicroPower/RT that may be useful.

1. Use VBGEXE to run MPP instead of R. To do this, set V to be the command for VBGEXE by typing:

   V:==R VBGEXE ^

   Then use a command string of the form:

   V MPP OBJFIL/switches PASFIL

   For a large test program, R takes 4:58 while V takes 2:28. I vow never to call VBGEXE the vegetable executive again.

2. To convert a MicroPower MIM file to a file that resembles a SAV file for PROM burning, use the SPLIT program to remove the first two blocks. To do this, type:

   SPLIT .FILNAM.SAV=FILNAM.MIM/B:2

   This will work if ROM memory is in one contiguous section. Dump the MIM file to check to see if this is the case. If the first block of the MIM file is all zero, except for the first line of the dump, then ROM is in one complete section.

Of course, since VBGEXE and SPLIT are unsupported, then DEC will not support the suggestions made here.

John T. Davies III
Thermo Electron Instruments
524 Alpha Drive
Pittsburgh, PA 15238-2912
(412) 963-0903

DMS SIG
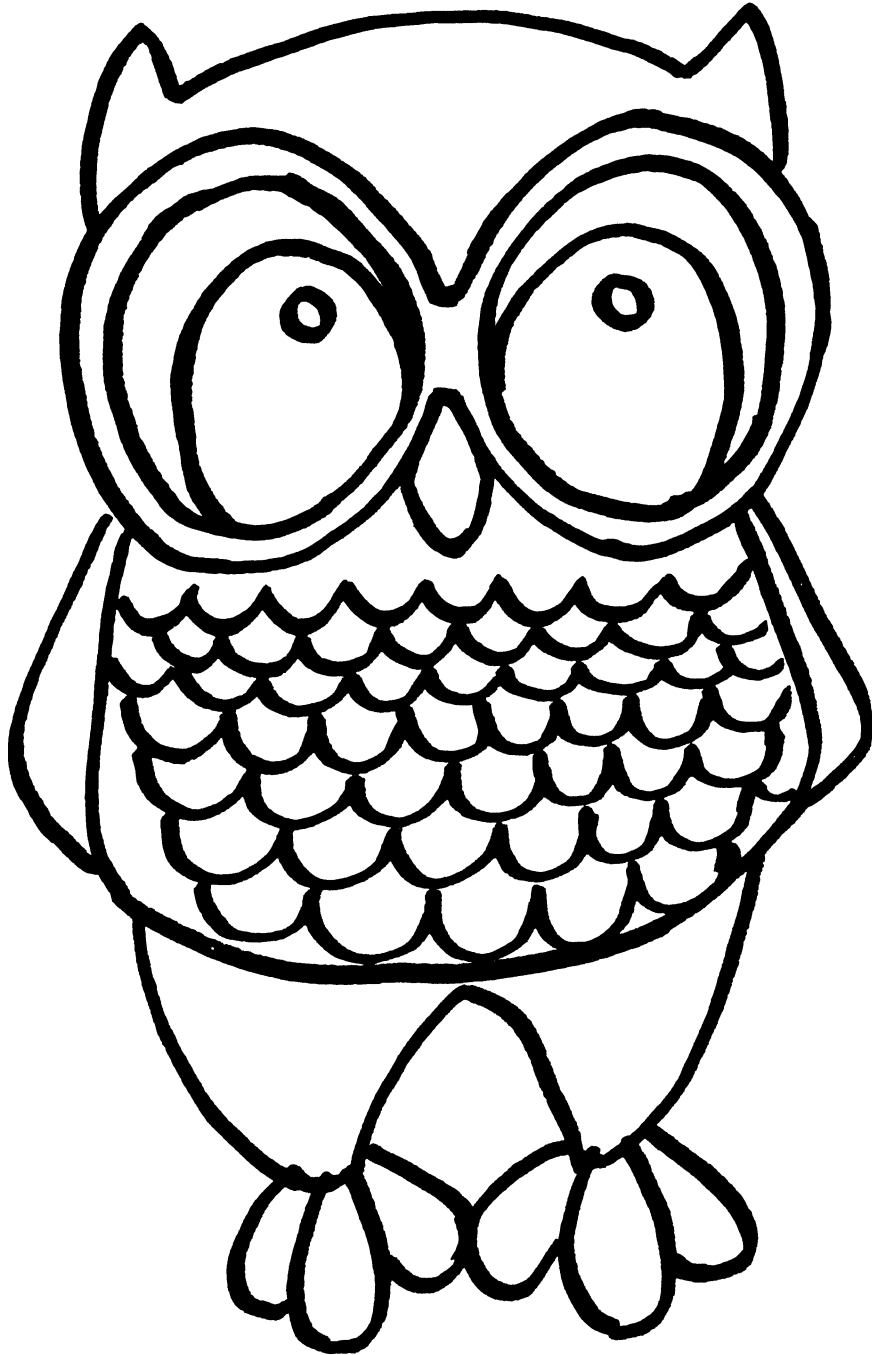
JOIN THE WISE

Data Management Systems SIG Steering Committee
February 1, 1986

SIG Chairman:
Joseph F. Sciuto
Army Research Institute
Alexandria, VA
(202) 274-9420

Comptroller:
Alan Schultz
Land Bank National DP Center
Omaha, NE
(402) 397-5040

Symposium Coordinator:
Keith Hare
JCC
Granville, OH
(614) 587-0157

Symposium Coordinator:
Barbara Mann
TRW
Redondo Beach, CA
(213) 532-2211

Communications Committee Representative/
    Newsletter Editor:
J. G. Russell Poisson
SEED Software Corporation
Alexandria, VA
(800) 428-9400

Session Note Editor:
Mark Morgan
Farm Credit Banks
Springfield, MA
(413) 732-9721

Membership Coordinator:
VACANT

MIS Working Group Coordinator (Past SIG Chairman):
Steve Pacheco
Ship Analytics
North Stonington, CT
(203) 535-3092

MIS Working Group (Past SIG Chairman):

Sandy Krueger
Key Financial Systems, Inc.
Pine Brook, NJ
(201) 299-6600

Working Group Coordinator/
    Database Working Group:
Jim Perkins
PSC, Inc.
Shelburne, VT
(802) 863-8825

Forms Working Group:
Debbie Kennedy
Land Bank National DP Center
Omaha, NE
(402) 397-5040

Non-Digital Working Group:
Doug Dickey
GTE Government Systems
Rockville, MD
(301) 294-8400

RMS Working Group Coordinator:
Allen Jay Bennett
Lear Siegler Apistan
Grand Rapids, MI
(616) 451-6429

Pre-Symposium Seminar Coordinator/
    Black Book:
David B. Turner
Korn/Kerry International
Los Angeles, CA
(408) 945-9600

ANSI Standards Coordinator:
Herman "Spence" Spencer
Army Research Institute
Alexandria, VA
(202) 294-9420

Member-At-Large:
Larry W. Hicks
Relational Database Services
Kernersville, NC
(919) 996-4882

Member-At-Large:
Richard Arndt
Cognos Corporation
Houston, TX
(713) 690-1105

AI SIC Liaison:
David Slater
Institute for Defense Analysis
Alexandria, VA
(703) 845-2200

Datatrieve Liaison:
John Schutt
J. R. Simplot Company
Boise, ID
(208) 336-2110

DEC Counterpart:
Wendy Herman
Digital Equipment Corporation
Nashua, NH
(603) 881-2494

Contributions

Contributions to the newsletter can be sent to either of
the following addresses:

Editor, DMS SIG Newsletter   Russ Poisson
c/o DECUS U.S. Chapter   DMS SIG Newsletter Editor
219 Boston Post Road, BP02   SEED Software Corporation
Marlboro, MA 01752   2121 Eisenhower Avenue
                                    Alexandria, VA 22314

Letters and articles for publication are requested from members
of the SIG. They may include helpful hints, inquiries to other
users, reports on SIG business, summaries of SPR's submitted to
DEC, etc. Machine readable input is highly desirable.
Submitters should keep in mind the DECUS policy on commercialism.

Documentation Set Winners

At the recent DECUS Symposium in Anaheim, California,
complete documentation sets for DEC data management products
were awarded to lucky DMS SIG Campground visitors:

Winners	Documentation Set
John P. Williams USDA-FAS-IAS Washington, DC	DBMS
Terry LeClair Consilium Mountain View, CA	CDD
Cathy Leonard Telic Corp Rockville, MD	FMS
Steve Thomas VIA Metro Transit San Antonio, TX	DTR
Steven Myerow Texcel Systems, Inc. Wayland, MA	ACMS
Tom Kelly Berkshire Medical Center Pittsfield, MA	TDMS
Robert Dependahl Santa Barbara City College Santa Barbara, CA	FMS-11.
Bhadra K. Patel Hughes Aircraft Co. Fullerton, CA	RDB/VMS

Coming Features

Later issues of this newsletter will feature articles on:

. Working Groups: How they assist the DMS SIG in solving
  User problems.

. Working Group Progress Reports from the following working
  groups:
                Database Working Group
                Forms Working Group
                Non-Digital Working Group
                RMS Working Group

. A written report on DEC's Response to "Wish-List" Items submitted
  at the DMS SIG Campground at the Fall '85 Symposia.

. SIG Chairman's Report on Plans for Spring '86 Symposia in
  Dallas.

. A critical review: "Is DEC's RDB a fully functional relational
  DBMS?".

. 'Relational vs CODASYL vs Hybrid' : a critical look at
  functionality in the real world of production data processing.

. "True Confessions of an RDB Novice/Is This an Undocumented
  Feature?" How RDB reacts to a delete request of a non-existent
  record.

Calling All Hands

DMS SIG is in need of a Membership Coordinator. If you are
interested in this leadership position within the DMS SIG,
please call the DMS SIG CHAIRMAN Joseph F. Sciuto at
(202) 274-9420.

Volunteers Needed - Volunteers are needed to participate in
user panels, chair sessions, and help in the DMS SIG campground
at the Spring '86 DECUS Symposia. volunteers should call the DMS
SIG Symposia Coordinator Keith Hare at (614) 587-0157.

DEC sends out conflicting signals on FMS and TDMS

Not so long ago, DEC released TDMS as the heir-apparent to
the niche carved out by its form package, FMS. Users at that time
had to consider whether to jump on the TDMS bandwagon or stick with

FMS, a product that DEC seemingly was abandoning. Since
that time, new releases of FMS have enhanced and maintained the
product while continued development of TDMS has lagged far
behind. What gives? What plans does DEC have to enhance TDMS?
Will FMS continue to be supported? Stay tuned for DEC's
response.

A HOOT FROM THE EDITOR
                        Russ Poisson, SEED Software Corporation
----

Starting a newsletter for a user group interested in
the very broad topic of data management is indeed a dif-
ficult task. The task is further complicated by the pro-
liferation of "technical" newsletters and the need to pro-
vide a new and fresh direction to those interested in data
management. I realize that such an undertaking will require
overcoming a considerable amount of negative momentum. With
this in mind, I invite all DECUS members to send any
articles, helpful hints, technical discussions, cartoons or
other material deemed suitable for publication in this news-
letter.

In particular, I challenge DMS SIG Steering Committee
to provide the necessary leadership required to accomplish
this goal.   The DMS SIG exists to provide communication in
three ways: user to user, DEC to user, and user to DEC.
Ladies and gentlemen, its time to get going! If you want
a successful newsletter, lets forget the euphoria of Disneyland
and our overblown status as "ribboned" symposia attendees and
start cranking out some articles. I hope to hear from you all
soon.

                                Thank you.
                                Russ

To
Baggage Claim
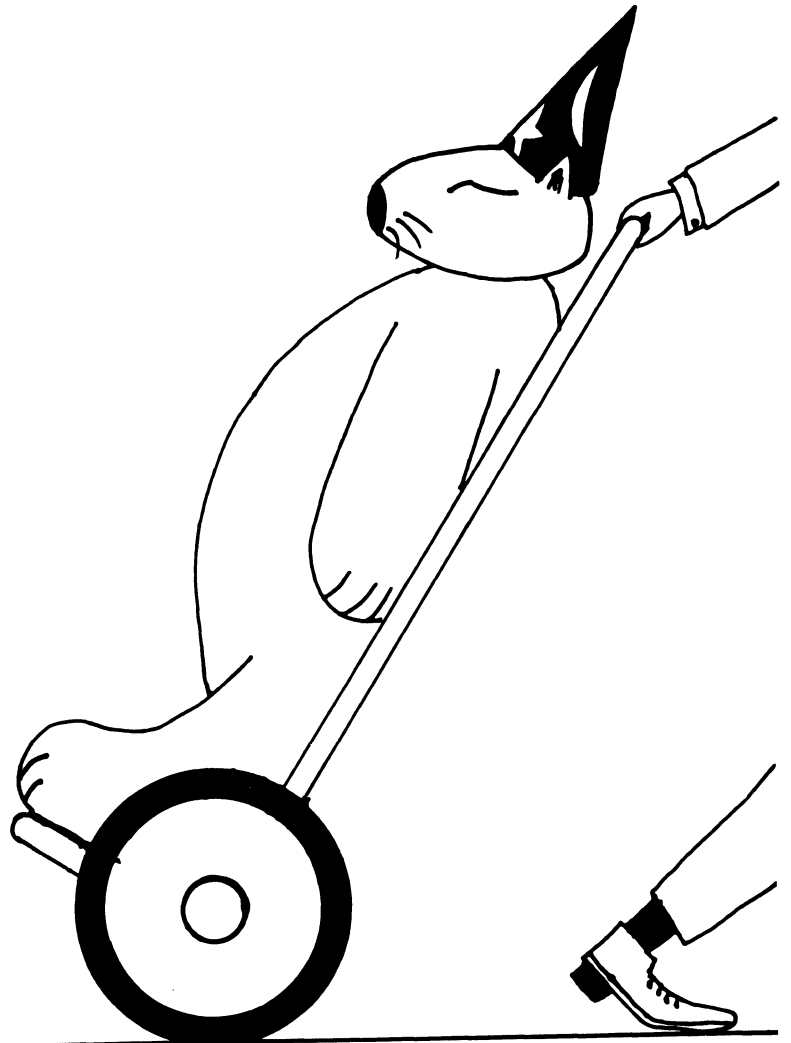
DATATRIEVE Special Interest Group - Officers

12-23-85

Role	Name	Organization	Address	City	Phone
SIG Co-Chairmen	Larry Jasmann	U. S. Coast Guard	10067 Marshall Pond Rd.	Burke, VA 22015	202-426-2344
	Joe H. Gallagher	Research Medical Center	2316 East Meyer Blvd.	Kansas City, MO 64132	
Communications					
Committee Rep.	Elaine McWilliams	Boeing Company	P. O. Box 24346 M/S 9c55	Seattle, WA 98124	206-773-0102
Editor	Donld Stern	Warner Lambert Company	10 Webster Road	Milford, CT 06460	203-878-9351
Production Editor	Steve Cordiviola	Kentucky Geological Survey	311 Breckinridge Hall	Lexington, KY 40506	606-257-5863
Artist	Bart Z. Lederman		2572 E. 22nd St.	Brooklyn, NY 11235	212-250-2300
Symposia					
Symposia Committee Rep.	Chris Wool	E.I. duPont Eng. Dept.	Louviers Bldg.	Wilmington, DE 19898	302-366-4610
Symposia Assitant	Diane Pinney	Computer Sciences Corp.	443 Inyokern Road	Ridgecrest, CA 93555	619-446-6585x284
Pre-Symposia Seminars	Dana Schwartz		15719 Millbrook Lane	Laurel, MD 20707	301-859-6277
Symposium Notes	Elaine McWilliams	Boeing Company	P.O. Box 24346 M/S 9c55	Seattle, WA 98124	206-773-0102
Product Development					
Development Coordinator	Bart Z. Lederman		2572 E. 22nd St.	Brooklyn, NY 11235	212-250-2300
New Product Focal Point	Diane Pinney	Computer Science Corp.	443 Inyokern Road	Ridgecrest, CA 93555	619-446-6585x284
Digital Counterpart	Andy Schneider	Digital Equipment Corp.	110 SpitBrook ZK02-2/N59	Naushua, NH 03062	
Library Committee Rep.	Gerri Williams	Computer Science Corp.	443 Inyokern Road	Ridgecrest, Ca 93555	619-446-6585x285
Volunteer Coordinator	Gerri Williams	Computer Science Corp.	443 Inyokern Road	Ridgecrest, Ca 93555	619-446-6585x285
Campground	Bert Roseberry	U. S. Coast Guard	500 Camp Street	New Orleans, LA 70130	504-589-4934
Special Effects	Alex L. Lamb	U. S. Army	32 Pearce Ave	Eatontown, NJ 07724	201-532-3318x1843
Special Consultant	Phil Naecker	J M Montgomery Engineers	250 N. Madison Ave	Pasadena, CA 91101	818-796-9141

DTR-i

DATATRIEVE Masters List

10-28-85

System	Name	Organization	Address	City	Phone
RSX & P/OS	Bart Z. Lederman		2572 E. 22nd Street	Brooklyn, NY 11235	212-250-2300
VMS	Joe Kelly	WymAn Gordon Co.	Worchester Road	North Grafton, MA 01536	617-839-4411x5480
VMS	Larry Jasmann	U. S. Coast Guard	10067 Marshall Pond Rd.	Burke, VA 20015	202-426-2344
VMS	James Swanger	G. D. Searle & Co.	4901 Searle Parkway	Skokie, IL 60077	312-982-7430
VMS	Chris Wool	E. I. duPont, Engineering Dept.	Louviers Bldg.	Wilmington, DE 19898	302-366-4610
VMS	Dick Azzi	Motorola SPS	Box 2953 MS: P116	Phoenix, AZ 85062	602-244-4316
VMS	Chris Hines	American Board of Family Practice	2228 Younge Drive	Lexington, KY 40505	606-269-5626
VMS	Bert Roseberry	U. S. Coast Guard	500 Camp Street	New Orleans, LA 70130	504-589-4934

Contributions

Contributions for the newsletter can be sent to either of the following addresses:

Editor, DATATRIEVE Newsletter
c/o DECUS U.S. Chapter
Company 219 Boston Post Road, BP02
Marlboro, MA 01752

Donald E. Stern, Jr
Warner   Lambert
10   Webster   Road
Milford, CT 06460

Letters and articles for publication are requested from members of the SIG. They may include helpful hints, inquiries to other users, reports on SIG business, summaries of SPRs submitted to Digital or other information for members of the DATATRIEVE SIG. Machine readable input is highly desirable and machine-to-machine transfer of material is preferred, but most anything legible will be considered. However, this newsletter is not a forum for job and/or head hunting, nor is commercialism appropriate.

Table of Contents

About the Cover

The section cover for this issue of the Wombat Examiner was drawn by Bart Lederman. Bart explains that the Wombat is exhausted after attending the Fall Symposium. After reading this issue, you will be keenly aware that a lot happened at Anaheim and the Wombat is entitled to some R and R. This issue, although published 2 months after the symposia, is being produced upon the immediate return of the volunteers.

DTR/4GL Product List

As a service to our readers, we are creating a new feature for the newsletter which will appear at least after each symposium. The following is the first of the series which attempts to provide readers current information with regard to the availability and other relevant information relating to products served by the DTR/4GL SIG.

Product Number	Version Dates	Announce/Ship System		Operating
VAX Datatrieve	3.3	11/85	12/85	VMS 4.2
	3.2	5/85	10/85	4.0
	3.0*			3.x
DATATRIEVE 11	3.1		8/84	RSTS 9.0
				RSX 4.1
				RSX+ 2.1
		12/85		Micro RSX 1.1
			1/85	VMS 4.0/RSX 1.0
	1.0		12/84	Micro RSTS 1.0
Pro DATATRIEVE	2.0		12/84	P/OS 2.0
	1.0*		.	
DATATRIEVE 20	1.0		9/84	
DECReporter	1.0	11/85		VMS 4.2
VAX Team Data		***		VMS/Rdb
VAX Rally		***		VMS/Rdb

*   unsupported
***   pre-announcement at 12/85 DECUS

Editor's note: If anyone has additional information to enhance this chart, or note an error, or can fill in the blanks, please let us know. This chart will only be as good as the information which is provided to us.

Chairman's Corner

Joe H. Gallagher, Research Medical Center, Kansas City, MO

Larry Jasmann and the rest of the SIG Steering Committee persuaded me to take over as Chair of the DATATRIEVE SIG. I will take over the full responsibilities of the Chair at the beginning of the 1986 Spring Symposium in Dallas. Until then, I will be helping Larry out with some of the duties of the Chair. My first assignment is to write this "Chairman's Corner" for the newsletter; Larry was fortunate enough to get two weeks vacation right after the 1985 Fall Symposium in Anaheim. So... here goes.

The 1985 Fall Symposium in Anaheim was a successful one. The attendance did not exceed that of the 1984 Fall Symposium (also in Anaheim), but it was more than the 1985 Spring Symposium in New Orleans. The Symposium was a financial success because very careful, almost austere, cost containment measures were put in place. The financial status of DECUS is in much better shape, and it appears that no further budget cutbacks will be required for the rest of the 1986 fiscal year.

The big news of the Symposium, at least from the point of view of the DATATRIEVE/Fourth Generation Languages SIG, was the "Pre-Announcement" of Digital's new 4GL products VAX Rally and VAX TEAMDATA. A "Pre-Announced" product means that you can not get any information in writing from Digital about the products because there is still a chance that the product names could change due to trade mark registration. TEAMDATA is for non-computer knowledgeable end-users and interacts with Digital's relational database management system providing graphics, spreadsheet, and automatic generation of default forms and reports. It is a database tool which can fit easily into the All-in-1 environment. On the other hand, Rally is an application generator which is designed to be used by computer knowledgeable end-users such as business system analysts, information center workers, and data processing professionals. Both products will increase worker productivity several times over the productivity of DATATRIEVE. We are all excited about the new products. Several full technical sessions and a pre-symposium seminar are planned for the Dallas Symposium.

Those of you who have been reading the newsletter faithfully will realize that the SIG is changing its name to DATATRIEVE/4GL and expanding its mission to provide a DECUS focal point for these two new products and other 4GL's. In addition, the DTR/4GL SIG also be the home for the recently announced product DECreporter. A presentation on the capabilities DECreporter was given by Peter Savage at the Anaheim symposium. For those of you that are familiar with DATATRIEVE, DECreporter can optionally be linked with VAX DATATRIEVE to provide a menu driven front end which generates DATATRIEVE code to produce reports. Alternatively, it can be used as a standalone product to produce reports on data contained in standard RMS files. It is a tool that will be very popular with non-computer knowledgeable end users.

There are several changes in the SIG Steering Committee. With me moving to the SIG Chair, Don Stern and Steve Cordiviola will be taking

over the Newsletter. Please submit articles for the newsletter to Don Stern. Gerri Williams will be taking over as the Library Committee Representative. Gerri will continue as Volunteer Coordinator.

The time till the next Symposium in Dallas, which will be the 25th Anniversary of DECUS, is very short. The Spring Symposium is about one month earlier than normal. So make your plans now.

Joe H. Gallagher,
SIG Chair Elect

From the Editor's Pen

Donald E. Stern, Jr., Warner Lambert Company, Milford, CT

As you may have noticed, the editor's pen is being guided by a different hand. Joe Gallagher has been elected to the position of DTR/4GL SIG Chair. He will fully assume the responsibilities of the post at the start of the Dallas Symposium. Until then, however, he will be assisting the current Chair, Larry Jasmann. Because of this, Joe is stepping down as the Editor of the DTR/4GL Newsletter. During his tenure as Newsletter Editor, Joe has made substantial contributions to the Newsletter in terms quality and content. Please join me in extending thanks to Joe for the fine job he has done and best wishes for success in his new position. Together with Steve Cordiviola, the Production Editor, we will try to fill the rather substantial void created by this move.

In order to contain the production cost of the newsletter and avoid the need to increase its price, the newsletter will be produced in a "two up landscape mode." In this way, the content of the newsletters can be maintained at a lower cost. The readership is urged to make comments and suggestions regarding the new format.

Finally, since the production of the Newsletter is totally voluntary, the quality of its content is directly related to the participation of its readership. Please consider writing an article, tech note, etc. of its readership.

Message from DTR/4GL Communications Committee Representative

Congratulations to Joe Gallagher

Joe Gallagher was elected DTR/4GL SIG Chair Elect at the DTR/4GL Steering Committee meeting held in Anaheim during symposia. Joe has been a key member of the SIG serving on the Communications Committee and as Newsletter Editor. The knowledge and dedication he brings into this new position will assure continued success for the SIG in the coming years.

Our current SIG Chair, Larry Jasmann, will continue in his roll until the Spring Symposium in Dallas. Along with his continued support to the SIG, Larry plans to participate in other areas of DECUS Leadership.

The DTR/4GL SIG would like to thank you, Larry, for the excellent leadership you have provided to the SIG the past three years and we wish you success in your new activities.

We have a new Newsletter Editor. Actually, we have TWO new Newsletter Editors. Don Stern and Steve Cordiviola have been named as co-editors of the Wombat Examiner. They will be working together to put out the same high quality newsletter that Joe Gallagher has done in the past. Like Joe, though, they will be expecting a little help from us. If you will submit an article (or articles) for publication in the Wombat Examiner, we'll publish them. We ask that they be:

- Technically accurate

- The subject be DATATRIEVE or a 4GL Product

- The article(s) may not violate DECUS commercialism policies.

If you have any questions about article publication, you can contact me, Steve, Don or Joe Gallagher and we'll be happy to answer any questions.

A lot of really exciting things happened in Anaheim at Symposia. We are looking forward to Dallas and an even more successful and productive symposium. In the meantime, DTR/4GL SIG wishes all of you the best in the new year and we hope to see you all at Dallas in the Spring.

Elaine V. McWilliams
Communications Committee Rep

News of the SIG

At a SIG Steering Committee meeting on December 10 during the 1985 Fall DECUS Symposium, new bylaws and operating procedures were approved. These new bylaws were also approved on December 13 by the SIG Council and will be taken to the Management Council for final approval. It is expected that these new bylaws will be approved during the January conference call meeting of the Management Council. A copy of the bylaws should appear in the March issue of the newsletter.

Also at the SIG Steering Committee meeting on December 10, Joe H. Gallagher was elected to be the next SIG Chair. Larry Jasmann, the current SIG chair, and Joe will be attending SIG Council activities together until the beginning of the Spring 1986 Symposium in Dallas at which time Joe will take over all of the responsibilities of the Chair. Joe will step down as Editor of the newsletter and take over some of the

duties of the Chairperson, such as the planning for the 1987 DECUS budget, immediately.

Don Stern and Steve Cordiviola will take over the newsletter activities. Don will concentrate on editorial policy and newsletter content and Steve will do the newsletter production. Of course, Bart Lederman will continue to provide the Wombat Examiner artwork.

On December 11, during the Wombat Magic Session at the 1985 Fall DECUS Symposium, Joe Gallagher was inducted into the DATATRIEVE Greybeards.

Andy Schneider (DATATRIEVE SIG Digital Counterpart) and Shirley Schneider (DMS SIG Digital Counterpart on maternity leave) are the proud parents of a baby boy. Eric William arrived October 29, 1985 weighing 10 pounds and 4 ounces. Mother and baby doing well, but those who have talked to Andy are not sure if the new daddy is going to make it.

Dan Dietterich, principle architect of VAX-DATATRIEVE and Greybeard inductee in May 1983 and Joan Hilton (long time SIG Steering Committee member and Greybeard inductee also in May 1983) were recently married.

Wayne Allen Jones (Greybeard inductee in December 1982 and the documentation specialist who started the tradition of good VAX-DATATRIEVE doc sets) is now in Digital's Terminals and Printers Engineering Program Office. His job is to try to get the software engineers to plan for and take advantage of new features in future Digital terminals.

Dave Norby (one of the founders of the DATATRIEVE SIG and Greybeard inductee in May 1982) has moved from G. D. Searle to Recycled Paper Products. Good luck to Dave in his new job.

Converting Decimal UICs to Octal UICs

Bert Roseberry Eighth Coast Guard (dt)
500 Camp Street New Orleans, LA 70130   (504) 589-4934

I read with great interest Don Stern's article (Wombat Examiner Volume 7, Number 2, October 1985) that had in it the record definition of the User Authorization File. While working on my own applications, I was bothered by the fact that the member number and the group number for the UIC were decimal values rather than Octal.

I wanted something that looked like a UIC in the form [group, member] for my application. One possible route would be to add a function to DATATRIEVE that converts decimal numbers to their octal representation (I wonder why the developers included the function FN$HEX but not FN$OCTAL?), however, due to things such as having layered products to relink if I "customized" DATATRIEVE, I decided against this.

Delving into the deep recesses of my books on partial differential equations and other obscure subjects I came up with nothing. I decided to visit one of the many fine establishments in the French Quarter and after a few drinks, it came to me and I hastily wrote down the following procedure on a napkin.

Whenever you want an output for the UIC that looks like a UIC,
first execute this procedure and then use the computed by variable UIC.

For example:

```
DTR> READY UAFV4 SHARED
DTR> :INIT_UIC
DTR> FOR UAFV4 PRINT USERNAME(-), UIC(-)


    Alabaster  [300,100]
    Cities     [300,200]
    Dream      [277,001]

DTR>
```

```
DEFINE PROCEDURE INIT_UIC
!
!      This is ONLY good for converting octal numbers up through
!      511 although there is no reason why it could not be
!      expanded to convert even higher numbers.
!
!      Get the "ggg" part of [ggg,mmm]
!
DECLARE G1 COMPUTED BY FN$MOD(GRP,8) .
DECLARE GSUB1 COMPUTED BY GRP - G1 .
DECLARE G2 COMPUTED BY FN$MOD(GSUB1, 64) / 8 .
DECLARE GSUB2 COMPUTED BY GSUB1 - (G2 * 8) .
DECLARE G3 COMPUTED BY GSUB2 / 64 .
!
!      Get the "mmm" part of [ggg,mmm]
!
DECLARE M1 COMPUTED BY FN$MOD(MEM,8) .
DECLARE MSUB1 COMPUTED BY MEM - M1 .
DECLARE M2 COMPUTED BY FN$MOD(MSUB1, 64) / 8.
DECLARE MSUB2 COMPUTED BY MSUB1 - (M2 * 8) .
DECLARE M3 COMPUTED BY MSUB2 / 64 .
!
!      Put it together in one six place number
!
DECLARE UT COMPUTED BY M1 + (10 * M2) + (100 * M3) +
        (1000 * G1) + (10000 * G2) + (100000 * G3) .
!
!      Format it properly
!
DECLARE UIC COMPUTED BY FORMAT(UT) USING "["999,999"]" .
!
END-PROCEDURE
```

Wombat Magic – Part 1 1985 Fall DECUS Symposium
Disneyland Hotel, Anaheim California

Session Chair: Bert Roseberry, U.S. Coast Guard, New Orleans, LA
Session Editor: Donald E. Stern, Jr., Warner Lambert, Milford, CT

As usual, the Wombat Magic session at Anaheim drew Wombat Wizards
from across the country. There, a great deal of powerful magic was
presented. (Magic is anything which might be interesting to other
DATATRIEVE users.) The content of the session will be shared with the
readership in several parts. An effort will be made to group the magic
by functionality rather than preserve the exact chronological order of
the session. Where appropriate, the presenter's comments are quoted
("...").

Wayne Jones, Digital Eqipment Corp.
     How to Find Your Way Around the Disneyland Hotel

"I've been facinated, and finally understood after the third time being
here, what the algorithm was for finding your way to your room. When I
applied it today, for the first time, it didn't work because I got con-
fused...but never mind."

```
DEFINE DOMAIN ROOMS USING ROOMS_REC ON ROOMS.DAT;
DEFINE RECORD ROOMS_REC USING
01 ROOMS.
    03 NAME PIC X(20).
    03 ROOM PIC X(4).
;

DECLARE ROOM_NUM PIC XX COMPUTED BY FN$STR_EXTRACT(ROOM,3,2).

DECLARE TOWER_NAME PIC X(6) COMPUTED BY CHOICE
    ROOM_NUM < 34 THEN "MARINA"
    ROOM_NUM BT 34 AND 67 THEN "SIERRA"
    ROOM_NUM > 67 THEN "BONITA"
END_CHOICE.

DECLARE BONITA_FINDER COMPUTED BY FN$STR_EXTRACT(ROOM,1,2).

READY ROOMS
FIND ROOMS WITH ...
REPORT CURRENT
    PRINT NAME,
        CHOICE
            BONITA_FINDER="57" THEN "BONITA"
        ELSE
            TOWER_NAME
        END_CHOICE,
    (FORMAT(FN$STR_EXTRACT(ROOM,1,1)) USING 9 +
    FORMAT(FN$STR_EXTRACT(ROOM,2,1)) USING 9)
        ("FLOOR"/"NUMBER"), ROOM_NUM
END_REPORT;
```

```
SUM 1 BY TOWER_NAME
```

and the result is:

NAME		ROOM NUM	FLOOR NUMBER
DON STERN	MARINA	8	21
BERT ROSEBERRY	BONITA	11	97
DTR/4GL CAMPGROUND	BONITA	11	98
LARRY JASMANN	BONITA	11	99

TOWER NAME	TOTAL	TOTAL
BONITA	3	
MARINA	1	4

Denis Haskin, Clark University
    Formatting a Social Security Number

"I'm a little ashamed, this one actually does something... For you ex-
pert DATATRIEVE people this is going to be nothing, believe me, but I
got excited enough about it when I hit on it that I literally ran around
the machine room a couple of times. Basically, we have a company giving
us a software package and it's written in BASIC so I don't touch it at
all. Luckily, they stored all of the record definitions in the CDD,
which is great because they gave us a horrible report writer which we
threw back in their face and promptly doing all our own report right out
of DATATRIEVE. Our users, in fact, even like it. They stored the so-
cial security number, for the students or applicants or whatever, as a
longword."

```
        .
        .
        .
    03 SS_NUMBER USAGE LONG.
        .
        .
        .
```

"What you want to do is to print it out in a normal format, i.e.
003-63-8199." Tried using (FORMAT SS_NUM USING XXX-XX-XXXX). What's it
do? Takes out the leading zeros; 363-81-99

"No problem, use the numeric edit string with keeps in leading zeros.
(FORMAT SS_NUM USING 999-99-9999).

"DATATRIEVE is smarter than you are; it's numeric and you can't have two
negative signs; 000 3638199

"This had me stumped, literally for a week. What you do is use a FORMAT
inside another FORMAT and it comes out right:"

    (FORMAT (FORMAT SS_NUM USING 999999999) USING XXX-XX-XXXX)

    The result: 003-63-8199

Les Hulse, The Gillette Company
    Another Way to Format Social Security Number

"In deference to my other Massachusettes collegue from Clark College,
I'd like to point out a slightly different way of doing something. Take
that social security number and stick any edit string character you want
inside the string in double quotes. It is literally printed in that
position and is ignored for the edit string character.

```
        .
        .
    03 SS_NUM PIC 9(9) EDIT_STRING 999"-"99"-"9999.
        .
        .
```
    . "This even works with date string and those things and you can
get some 'wierd' things by bypassing pieces of the date."

Phil Naecker, Consultant
    Outputting Negative Numbers for Accountants

"The problem with the EDIT_STRING paren's is that it leaves the paren's
very far out at the limits of the field you are defining. So if you
define a field with an edit string (Z(9)), it means that you want a
field with up to nine digits and, if its negative, enclosed in paren's.
Well, sometimes accountants are very picky about this and they don't
want parentheses way out there where they might miss them five whole
character spaces away. What they'd like to see is the parentheses imme-
diatly around the value. Once again we go to CHOICE, which is the way
'real programmers' work in DATATRIEVE and you compute the value."

    DECLARE VALUE USAGE LONG EDIT_STRING IS (Z(9)).

    DECLARE STRING COMPUTED BY
        CHOICE
            VALUE GE 0 THEN VALUE
        ELSE "("||VALUE||")"
        END_CHOICE.

    [Ed. example:

        VALUE = -50
        PRINT VALUE, STRING

        VALUE   STRING

    (50)   (-50)   ]

```
DECLARE ED_STRING COMPUTED BY CHOICE
    VALUE GE 0 THEN VALUE
    ELSE "("||FN$NINT(FN$ABS(VALUE))||")"
END_CHOICE.

DTR> PRINT VALUE, STRING, ED_STRING
```

| | | ED |
VALUE	STRING	STRING
(50)	(-50)	(50)	]

Kathy Wrobel, DEC Telephone Support
   Prompting for a Sort Key
   (Presented by Phil Naecker)

!!!!!!!!!!!!!! 3rd Prize Winner !!!!!!!!!!!!!!

"This one is submitted by Kathy Wrobel who works in the Telephone Sup-
port Center. Many of you may have talked to Kathy. Kathy tells me that
if she wins then I get a kiss. So, if you've seen Kathy, you know that
this is a very serious bribe time and I want all the judges to give me
their wish lists. Yet another computed by but this one's with sorts."

```
DECLARE SORT_FIELD PIC X(10).

DECLARE SORT_KEY COMPUTED BY CHOICE
    SORT_FIELD CONT "RI" THEN RIG
    SORT_FIELD CONT "LO" THEN LOA
        .
        .
        .
    ELSE ""
END_CHOICE.

FOR YACHTS SORTED BY SORT_KEY PRINT RIG, LOA, PRICE

FIND (heaven forbid) YACHTS SORTED BY SORT_KEY    also works.
```

"Does anyone know what you can't do?"

```
FOR YACHTS MODIFY USING SORT_KEY="New Name"   does not work;
```
DATATRIEVE returns the following message:

```
Cannot assign to a virtual field.
```

Bob Hoover, HLP Inc.
   Maintaining Dictionary Elements in a Captive Account

"My magic is about having users that you try to keep captive inside a
menu and you really don't want them to get to the DATATRIEVE prompt,
they're a little dangerous. What I've done is design a routine which
brings up a maintenance menu. Now a maintenance menu is not too tough
if you got a table and its a domain table but what do you do if you see
a menu something like this.

```
┌─────────────────────────────────────────────┐
│                 Maintenance                  │
│                    Menu                      │
│                                              │
│         1.  Edit TABLE_1                      │
│         2.  Edit TABLE_2                      │
│         3.  Edit TABLE_3                      │
│         4.  Edit JUNK Table                  │
│                                              │
│         F   FINISHED                         │
│                                              │
└─────────────────────────────────────────────┘
```

There are three or four <dictionary> tables and the only way to maintain
the tables is to give a user access to the table. How do get to the
table with an EDIT, keeping them captive?   It actually a very simple
trick or piece of magic."

```
DECLARE OPTION PIC X.
WHILE OPTION NE "F" BEGIN
    (display the menu)        OPTION = *."Option"
    CHOICE
        OPTION = "1" THEN PRINT FN$SPAWN("DTR EDIT TABLE_1") ON NL:
        OPTION = "2" THEN PRINT FN$SPAWN("DTR EDIT TABLE_2") ON NL:
        .
        .
        .
    END_CHOICE
END
```

[Editor's note: To use this magic you must first include the FN$SPAWN
function; it is not currently supplied standard with DATATRIEVE]

Phil Naecker, Consultant
   Solution to READY *."Domain"

"Everybody wants to ready a domain 'on the fly' and decide which domain
to ready 'on the fly'. The problem is you can't 'IF something THEN
READY this ELSE READY that' but, as pointed out earlier, you can create
logicals 'on the fly' inside an IF or a CHOICE and you can ready a do-
main via a logical. So..."

```
DECLARE WHICH_DOMAIN PIC X(32).

WHICH_DOMAIN = *."Which one on list"

CHOICE
    WHICH_DOMAIN CONT "A" THEN
        FN$CREATE_LOG("PSEUDO","DOMAIN_A")
    WHICH_DOMAIN CONT "YAC" THEN
        FN$CREATE_LOG("PSEUDO","YACHTS")
        .
        .
        .
    END_CHOICE

READY PSEUDO
```

"If your a really slick DATATRIEVEr, you'll place the smallest unique string after the CONTAINING and, by using CONTAINING instead of EQUALS, it will do case translation so whether you type in lower case or upper case you'll get exactly what you want. You'll also include a list in the 'which domain' prompt to let users know which domains can be readied."

>>>>>>>>>>>> More magic to come in future issues >>>>>>>>>>>>>>>>

Datatrieve Wishlist, Fall 1985 Symposia, Anaheim, Ca

Bart Z. Lederman, Wishlist Coordinator

The wishlist is an informal method of making the user needs known to the DEC DATATRIEVE developers. Users submit their requests during the DECUS symposia. On Thursday afternoon the items are organized and given to the developer: in this instance we were fortunate in having Suellen Harris, VAX-DTR developer, and Bill Opalka, DTR-11 Software Engineer, present to cover all areas of DATATRIEVE. Readers should note that these are _informal_ responses, and _cannot be taken as commitments by DEC_; however, in the past many suggestions have made it into the product. Readers should also note that anything marked Editor's note is my opinion, not DEC's nor DECUS's. In a few cases, a solution or work-around is presented to the problem.

1. In DTR sort order, have the hyphen ("-") take precedence over blanks and the plus sign ("+"). This would help in sorting numbers in alphanumeric strings.

Comment: This has to be answered by the Sort people. There is little likelyhood we could change the default without upsetting all current users.

2. The report writer default of 60 lines/page should actually be 60 lines/page and not 61 lines/page. (DTR-11)

Comment: This is a bug fix, a patch will be available shortly in an update.

3. When the report writer prints lists, it does not keep track of the number of lines for a page break (it will complete the list before the page break). Have it page break, complete the list, then page break again with headers. (Editor's note: this item has appeared before for DTR-11).

Comment: We will look into this.

4. We MUST have more pool space regardless of how it is done (I-and-D space, mapping, resident libraries)!!! (Editor's note: this item has appeared many times before for DTR-11).

Comment: We are aware of this problem. We are looking into these options. We cannot, however, discount the people who have DTR running on machines that do not support I-and-D and Supervisor mode (11/34, 11/23, 11/24, Micro-11 and PRO-350, and other older machines.)

5. There were a number of requests for improved graphics, such as adding titles to plots, Tektronix 4014 family and other new device support, delimiter lines on graphs, and new PLOTS, which were perceived by DEC to be different aspects of a single problem and were addressed as such.

Comment: Datatrieve provides certain plots with certain capabilities, but it's not a graphics package per-se. It is not our goal to expand the capabilities of the current plots or the current plot language. You can use DECGraph for some of the requested options like titles on graphs. As to new devices and new plot languages other than ReGIS, we are currently investigating that possibility.

6. There were also several questions about editing on the VAX, which DEC felt could be treated as one item. They included:
How about journaling for the editor a la EDT? Versions are nice, but sometimes you spend a long edit session making the last version obsolete and lose it when your process dies; and, we need the capability to select the editor used with DTR. This is important to avoid training users in multiple editors, and would gain performance.

Comment: We are currently investigating the editing capabilities in DTR and will consider this at the same time.

7. Wild cards for dictionary items. DMU provides some capability but something more like DCL would be nice. (Editor's note: this has appeared before as, for example, SHOW FA* which would show all items starting with the letters "FA", for listing selected sets of items.)

Comment: This is still on our wish list but is not one of our highest priority items. In any case, it has to be done by the CDD group, and we will take back the suggestion to them.

8. Add to the record definition an "ERROR_MESSAGE" qualifier so a specific message will be output if a storage or validation error occurs on that field.

```
10 FIELD_1 PIC X QUERY_HEADER IS "field"
         ERROR_MESSAGE IS "No hyphens".
```

Comment: This is an interesting new suggestion, we will add it to the wish list. (VAX. Not likely in DTR-11.)

9. I have laser printers and use DTR to create custom forms. I need to be able to specify where the report header and page number appear on the report.

Comment: You can do this now by suppressing the report header and page number and doing your own.   (Editor's note: try looking at some of the Wombat Magic sessions transcribed in the Wombat Examiner.)

10. I'd like LOCK_WAIT to take effect on CDD node locks.

Editor's
 note: During the closing session, it was suggested that this CDD option can be implemented by assigning the logical CDD$WAIT to "YES". I have found that an explanation of this is included in the CDD installation manual.

11. Add FN$SPAWN("text") where "text" is a DCL command or command file. Yes, you can add this yourself but it is so useful it ought to be standard. (Editor's note: the ability to spawn command lines in DTR-11 on those systems that support it would be mighty useful.)

Comment: This has been suggested before, we will take a serious look at it due to the number of requests (VAX). Functions are currently on the wish list for DTR-11.

12. Expand DROP to allow an RSE.

Comment: Interesting suggestion, will add it to the wish list (VAX). Not likely for DTR-11.

13. Can DTR/Rdb COMMIT be expanded to commit a relation name only?

Comment: Rdb doesn't support it, they must do it first. We will take this back to the Rdb people.

14. Allow COMPUTED_BY fields in VIEWS.   They would be created from fields extracted for the view.   (Editor's note: without having to have the COMPUTED_BY in the source domains.)

Comment: Interesting; without looking at the code we can't tell how hard it is. We will look at it when we get back (VAX). Nice thing to do, but not likely in DTR-11

15. COMMAND LINE EDITING (be able to store and retrieve the previous 20 (or N) commands.

Comment: Not very high on priority list: VMS only gives us the ability to retrieve the last command (VAX). Not likely within DTR-11.

16. I would like a copy facility where I can copy procedures from one dictionary to another.

Comment: You can do this now with DMU COPY or DATATRIEVE EXTRACT. (Editor's note: the general consensus was that this was either a training or perception problem on the part of the user, or that the wording of the request didn't actually represent what the user intended to do.)

17. Change the default on READY from EXCLUSIVE to SHARED. Please! Better yet, make the default mode on READY an option for the system manager during installation. (Editor's note: this has appeared at least once before.)

Comment: Fairly high on the priority list now.   We can't change defaults (that would create problems for existing customers), but it may be possible as an option.

18. Would like the ability to specify STABLE sort option.

Comment: We looked into this, and determined that it might be a performance problem. We will look again, but not at a very high priority.

19. Would like to be able to include an RSE for the first domain in a CROSS:

          FOR ALL A WITH F1 = 7 CROSS B OVER K1

instead of running off a collection. May be in a compound statement:

          FIND ALL X IN A WITH F1 = 7 FOR ALL X CROSS B OVER K1.

Comment: Interesting, but we don't know how "doable" it is. We will look at it.

20. Would like a way to concatenate collections. (Submitted twice).

          FIND A IN D1 WITH K1 = 5 FIND B IN D2 WITH K2 GT 10 (merge A and B into C)

This would use keys to retrieve A and B.   It could also do OUTER JOIN and UNION operations.

Comment: Good suggestion, but probably won't be done in DTR (VAX). Nice idea, but unlikely in DTR-11 due to pool considerations.

21. Please provide a way to let DTR recognize record I/O files as sequential files. READY for modify if possible, not ready write (no deletes).

Comment: Additional input needed from submittor.   (Editor's note: this was intended primarily for DTR-11, and as it recognizes most RMS files, the nature of the problem could not be determined.)

22. Allow FIND and SELECT in REPEAT and BEGIN-END blocks. (Editor's note: has appeared before.)

Comment: Architecturally impossible in DTR: it would have to interpret every single line, which would degrade performance.

23. Allow the option of printing headings for each record in a FOR statement not just one heading at the top. This should be an option.

Comment: This can be done now with:

    FOR domain PRINT "headers", SKIP, field(-)

    or a similar construction.   Remember to suppress the headers on the field as shown.

24. Make EDIT more compatible with EDT. (Has appeared before in one form or another, and is placed separately here as this request is intended for DTR-11.)

Comment: This is on our wish list.

25. Allow a method to delete output files during procedures. We would like to be able to prompt the user, "This file already exists... do you want to delete it (or supersede it)? (Editor's note: in some cases you can do this by re-issuing the DEFINE DOMAIN command with the appropriate qualifier to supersede, but you cannot delete a file from within DTR.)

Comment: This is a problem for DTR-11 specifically on RSTS/E. We will give this consideration in any future release.

26. Allow a method to see non-printable characters when checking files. Operators confuse the STORE and MODIFY defaults. They will sometimes enter a tab when they want a field left blank on a STORE the way they have to on a modify. This results in a TAB being stored in the file. I would like to see this confusion end.

Comment: I don't think there is anything we can do to make non-printing characters visible.   We will look into preventing these characters from being stored. During the session, it was suggested that a VALID_IF clause could be added to the record definition making TAB and other unwanted characters invalid (field VALID_IF NOT CONTAINING "<TAB>", etc.).

27. Why doesn't the SUM command use the default edit-string specified in the record definitions for printing the amounts? It's a real hassle having to specify an edit-string for each field used in the SUM report. The rounding used is not useful.

Comment: DTR scales up the edit-string to prevent overflows: it can't tell ahead of time if the summed data will fit into the default edit string.

28. "SUGGESTION": I suggest a separate session or BOF meeting to inform users about upcoming changes, new features, problems, or whatever in new/upcoming releases of CDD and DTR so that we don't have to sit through INTROs, Tutorials, or other basic information sessions since this information tends to be supplied at the end of these types of sessions.

Comment: Nice idea. We will consider this for the next symposia.

29. Would like to use the DTR-11 call interface without having DECNet.

Comment: This is on our wish list, and we are looking into this.

30. Put $$MSG and related symbols needed for the call interface in a DATATRIEVE symbol library.

Comment: This is probably a user error due to not having DECNet on the system: DECNet is currently required for remote DTR-11.

31. Please add support of accessing lists by indexing (i.e., indexing into lists). We have lists that are 1000 long, and find the retrieval time for a specific record in the list too long since lists are accesses sequentially.

Comment: It's been suggested before, and is very difficult to do. There have been magic sessions published in the Wombat Examiner to do things like this. If you have a list this long, maybe you need to normalize your data. (Editor's note: the reaction of the audience supported the recommendation that there are better ways to handle data than lists this long.)

32. Is it possible to add "conditional ready" to DTR? (Be able to ready domains inside an IF-THEN-ELSE statement?)

Comment: Not likely. Some magic to ready using a logical name giving some dynamic capability has been published. (Editor's note: I believe another example is due soon from the Anaheim Wombat Magic session.)

33. Provide "official" DTR analysis of ACCOUNTING.DAT: this is a very useful tool for system management.

Comment: We are working with VMS to provide a solution. The current format of this file is not readable by DTR.

34. We would like a better interface with TDMS including request library.

Comment: There are no plans to enhance DTR in the forms area.

35. How about DTR on the Rainbow?

Comment: Not under development at this time. We will take the sugges-
           tion back to the product managers.

36. We would like the ability to cross a domain over multiple domains
OVER 2 or more different fields. Currently we are allowed only one OVER
clause.

Comment: Architecturally difficult to do, and provides users with more
           ways to get in trouble. (Editor's note: I have solved simi-
           lar problems by using a VIEW to tie two of the domains toge-
           ther, which may be done with more then one field. The third
           domain can CROSS over the view.)

37. I would like the ability to form a summarized collection, which
could then be sorted and printed with report writer. I want the ability
to produce a summary on one field, but report sorted on the totaled
fields. For example: summarize salesman detail sales records by sales
amount, but then report the summary sorted in descending order of total
sales to show the salesman with the most sales first. (Editor's note:
there was a similar suggestion on the last wish list.)

Comment: We will add this to the wish list. As a work around, try
           writing the sums into a temporary file, then report it sorted
           in the desired order.

38. Would like to perform an Outer Join. (Editor's note: has appeared
many times before.)

Comment: This is probably not something we are going to do in DTR.

39. Provide "if member", "if owner", "if empty" boolean expressions for
DBMS.

Comment: Will add it to the list, not very high priority.

40. Need a way to suppress either the first or last (or both) form
feeds in report writer.

Comment: Has suggested before, is on the wish list, not very high
           priority.

— THE WOMBAT

EXAMINER

July 1979 Vol. 1 No. 1 INCREASES THE CIRCULATION OF ANYONE IN AMERICA

DTR V2 Will
Support Variables

Planners Reveal Secret Plans

SECRET PLANS
REVEALED

PAGE 19

| Wombat Attacks Programmer *page 87* |
| SIN Tax Error— IRS Confused *page 40* |
| DTR Flushes Input *page 42* |
| Flood in New Orleans *page 49* |
| Yacht Mysteriously Erased *page 127* |
| Programmer Kills Sex-Crazed Wombat *page 931* |

Computer Predicts Wombat Invasion

Photo Reveals Lurid Details

IN CODE
WE TRUST

edusig

EDUSIG

Chairman
Robert A. Shive, Jr.
Millsaps College
Jackson, MS

Symposium Coordinator
Sue Bates
Northwestern Michigan College
Traverse City, MI

Communications Committee Representative
Robert W. McCarley
Millsaps College
Jackson, MS

Newsletter Editor
Fred Bell
Taft College
Taft, CA

PSS Coordinator VAX Systems SIG Liaison
Donald C. Fuhr
Tuskegee Institute
Tuskegee Institute, AL

Administrative Applications Coordinator
Dave Cothrun
Taft College
Taft, CA

Courseware Coordinator
Mary Jac Reed
Off Comp Based Instruction
Newark, DE

DEC Counterpart
Gary Finerty
Digital Equipment Corporation
Marlboro, MA

GRAPHICS

DECUS

GRAPHICS

Chairman
William Kramer
University of Delaware
Newark, DE

Symposium Coordinator
Bijoy Misra
Smithsonian Institution
Cambridge, MA

Newsletter Editor
Michael P. Anton
Houston, TX

Session Note Editor
Mike McPherson
Michigan State University
East Lansing, MI

Standards Coordinator
Jim Flatten
Ames Lab
Ames, IA

Library Committee
James M. Turner
Saber Technology
San Jose, CA

DEC Counterpart
Susan Usilton
Digital Equipment Corporation
Nashua, NH

Information Officer
Mike York
Boeing Computer Services
Seattle, WA

Human Interface Working Group Coordinator
Tom Owens
Graphics Research Center
Baltimore, MD

Engineering Working Group Coordinator
Dana Smith
Wilmington, DE

H M S

HMS

Chairman VAX SIG Liaison
Thomas J. Provost
MIT/LNS Bates Linac Facility
Middletown, MA

Product Planning Coordinator
George Hamma
Synergistic Technology
Cupertino, CA

Symposium Seminar Coordinator
Mike Allen
Lawrence Livermore National Labs
Livermore, CA

Communications Coordinator
John G. Hayes
Information Systems - S. Central Bell
Birmingham, AL

Publications Coordinator (Editor)
Bill K. Walker
Monsanto Research Corp.
Miamisburg, OH

Session Notes
DAARC SIG Liaison
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

Standards Coordinator
CAMAC Working Group Coordinator
Peter Clout
Los Alamos National Lab
Los Alamos, NM

LUG Coordinator
Gregg Giesler
Los Alamos Science Lab
Los Alamos, NM

Pre-Symposium Seminar Coordinator
Mike Allen
Lawrence Livermore National Labs
Livermore, CA

TOEM (Chips % Boards)
Jack J. Peterson
Horizon Data Systems
Richmond, VA

HHK (Hardware Hints & Kinks)
Wayne Kesling
Monsanto Research Corp.
Miamisburg, OH

UNIBUS Hardware
Ron Bogue
LIV Aerospace & Defense Co.
Dallas, TX

Performance Measurement Coordinator
William Wallace
600 W. Washington St.
Peoria, IL

CAMAC Coordinator
Peter Clout
Los Alamos National Lab
Los Alamos, NM

CSS Coordinator
Pratap Gohel
E.I. Dupont
Ingleside, TX

Networks SIG Liaison
Sandra Traylor
Target Systems
Yorba Linda, CA

VAX SIG Liaison
Dave Schmidt
5100 Centre Avenue
Pittsburgh, PA

DAARC SIG Liaison
Bill Tippie
Kinetic Systems Corp.
Lockport, IL

UNISIG SIG Liaison
Jim Livingston
1 Results Way
Cupertino, CA

SITE SIG Liaison
Emily Kitchen
A.H. Robbins Co.
Richmond, VA

RT-11 SIG Liaison
Gary Sallee
Sallee Software Consulting
Yorba Linda, CA

RSX SIG Liaison
Hans Jung
Associated Press
New York, NY

Members-At-Large
Mike Rembis
American Dade
Costa Mesa, CA

Hans Dahlke
Richland, WA

Jim Cutler
EDS Tower, 26533 Evergreen
Southfield, MI

DEC Counterparts
Terminals
Nina Abramson
Digital Equipment Corporation
Maynard, MA

TOEM (Chips & Boards)
Art Bigler
Digital Equipment Corporation
Marlboro, MA

Diagnostic
George D. Cooke
Digital Equipment Corporation
Maynard, MA

Storage
Marilyn Fedele
Digital Equipment Corporation
Maynard, MA

MSD (Micro Systems Development)
Roy Rodgers
Digital Equipment Corporation
Maynard, MA

Printer Products
Frank Orlando
Digital Equipment Corporation
Maynard, MA

DECUS Europe Liaison
Hans Zoller

THE
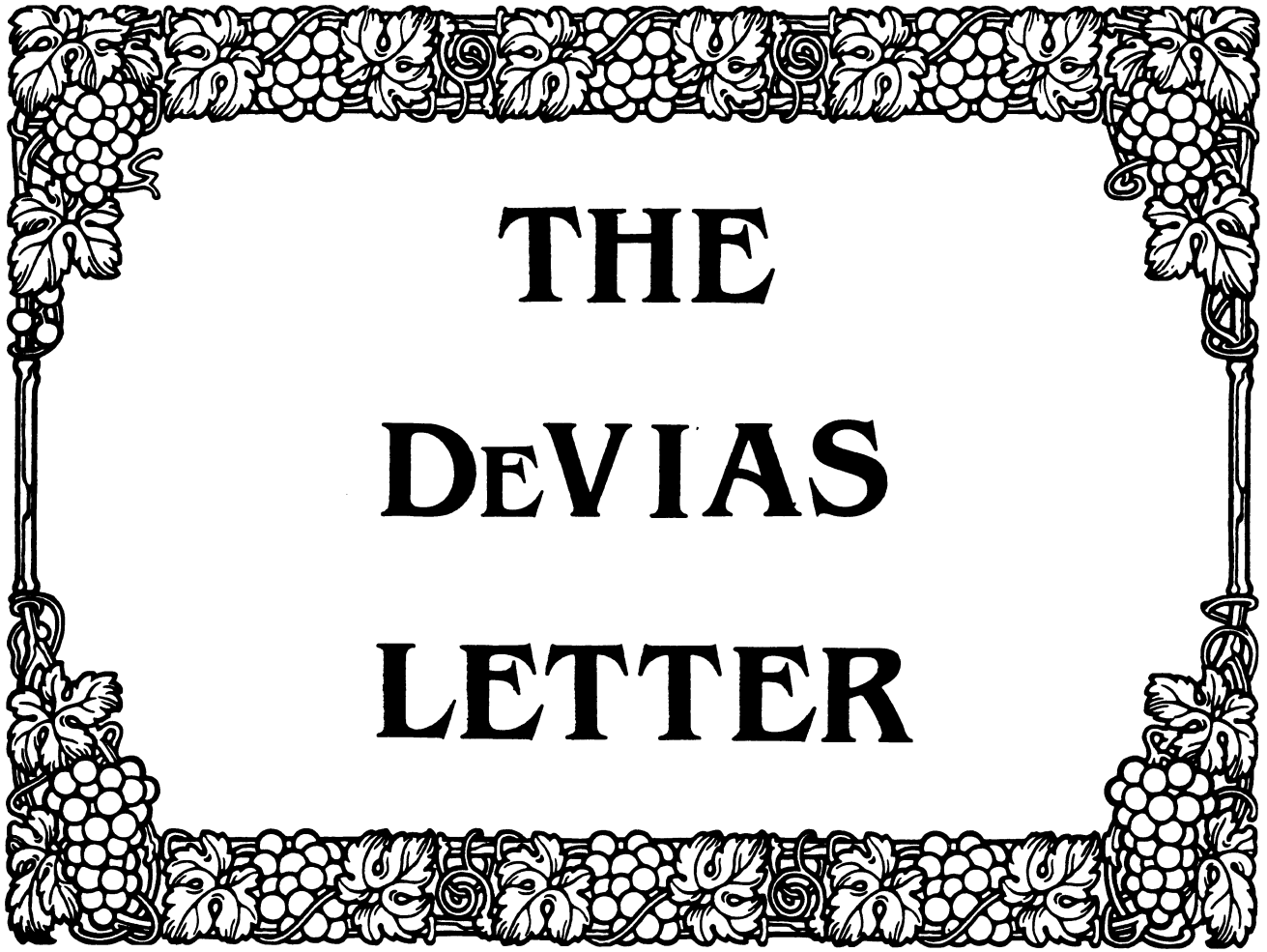
DeVIAS

LETTER

The DeVIAS Letter
Issue 35 February 1986

Curley's Corner: News from the Chairman

Letter from the Editor

IAS SIG Steering Committee

Chairman
  Bob Curley
  Division of Medical Physics
  University of Pennsylvania
  Philadelphia, PA

WHIMS Commissioner
  Kathleen Anderson
  Eaton Information Management
     Systems Division
  Hampton, VA

Library Coordinator
  Bob Schuldt
  INCO Inc.
  McLean, VA

RSX Liaison
  Ray French
  Boeing Computer Services
  Seattle, WA

Member-at-Large
  Doug Reno
  Abbott Laboratories
  North Chicago, IL

DEC Counterpart
  Mike Reilly
  Digital Equipment Corp.
  Maynard, MA

Symposium Coordinator
  Skip Stanfield
  USAF
  Washington, DC

Librarian
  Mike Robitaille
  Grumman - CTEC, Inc.
  McLean, VA

DeVIAS Letter Editor
  John Roman
  McDonnell Douglas - Dept. N436
  600 McDonnell Blvd.
  Hazelwood, MO   63042

Member-at-Large
  Kerry Wyckoff
  LDS Church
  Salt Lake City, UT

DEC Counterpart
  Tim Leisman
  Digital Equipment Corp.
  Stow, MA

DEC Counterpart
  Bob Mack
  Digital Equipment Corp.
  Landover, MD

Division of Medical Physics
Department of Radiation Therapy
University of Pennsylvania
Room 410 - 133 South 36th Street
Philadelphia, Pennsylvania 19104-3246
29 December 1985

Dear IAS Enthusiast,

Another Symposium is finished. This one was different and I can
not verbalize the strangeness. Other people mentioned it too. I
would like to hear your opinion, you who where there. Overall it
was good for IAS and good for DECUS in general.

The IAS attendance was about the same as New Orleans, but
quieter. Those who came didn't talk much. The QA (Technical
Forum) was pretty dull, it seems as if most of our questions have
been answered already. I read some of my SPRs and put everyone
to sleep. Michael Reilly and Alison Nylander were amazing (as we
have come to expect) with the breadth and depth of their
knowledge of IAS. A new member of the panel in the Technical
Forum was Ed Milhomme. He is the person at DEC Telephone Support
you are most likely to talk to. His discussion of telephone
support, and especially the differences in his perception from
that of Norm Booth (IAS Product Manager), was helpful. It was a
nice opportunity to meet the familiar voice on the other end of
the telephone. I am glad that you could come Ed, I appreciated
your presence.

The pre-symposium seminar on VAX MACRO for MACRO-11 programmers
was successful. Mike Reilly and Kerry Wyckoff (LDS Church, Salt
Lake City) were the principal lecturers and I gave a small
introduction. It was not as popular as I had expected, but we
got good reviews. Thanks to the lecturers and the attendees.
Since I feel the "MACRO-11" connection hampered our attendance,
we are offering a seminar in Dallas called "Introduction to VAX
MACRO" being taught by Kerry Wyckoff and John Roman (our editor,
McDonnell-Douglas, St. Louis). And we are again offering our
popular seminar of previous symposia, "Introduction to MACRO-11".
This time it will be taught by Bob Agnew (Beaver College,
Glenside, PA) and Mike Garcia (Digital Equipment, Maynard). Bob
has taken part in a previous seminar and Mike has given some
great presentations at previous symposia.

The last of our offerings for Dallas will be an introduction to
the C language. Mike Reilly and Alison Nylander will teach it.
It is a venture into a new area for the SIG, but it is done with
the permission of the Languages and Tools SIG, who provide the
focus for C in DECUS.

There were several good technical sessions during the Symposium.
We will try to get some of that information into the newsletter.
There are audio tapes available through DECUS of most of the IAS
sessions, but I haven't tried them to see how useful they are.
Some comments from someone who has would be appreciated.

We have a new 'counterpart' for the SIG. This is a position held
by someone from Digital who is interested in helping the SIG and
is invited by the SIG to that position. We have had one
counterpart since the formation of the SIG, Tim Leisman, who was
the IAS Product Manager at the time. Tim has held that position
since his appointment and will continue. The new position is a
result of some internal changes within Digital and the amazing
amount of support and help that we've received from Bob Mack of
the Government Systems Group. I have invited Bob to become an
'official' Counterpart of the IAS SIG, and both he and his
management have just agreed. Welcome Bob!

We also have two new people on the steering committee, Kerry
Wyckoff and Doug Reno (Abbott Laboratories, North Chicago, IL).
Both of these men have extensive experience in IAS, RSX and VMS
and will contribute some of their effort to administering the
SIG. Thank you both for responding favorably to my request that
you join in.

An interesting gentleman from Digital, Tim Moffitt, came with Bob
Mack. We got to swapping tales of the 'old days' and morsels of
PDP-11 trivia. One thing led to another and I think that he's
going to do a column for the newsletter on PDP-11 Trivia. You
may contribute. An example from Tim: "DEC made one disk
controller that did 18 bit data transfers on the Unibus. What
was it, where was it used, and what lines were used to transmit
the two high-order data bits?" "The answer is the RK11-D
controller for the RK05 disk drives. The 18-bit version was used
in conjunction with the PDP-11/10 to interface "high speed" disk
drives to a variant of the PDP-15 known as the Unichannel-15.
There were only a few hundred sold, but they worked well. The
high order data bits used the PA and PB lines on the Unibus."
"Some people will consider this a trick question as many RH11's
do essentially the same thing, but RH11's are Massbus
controllers, not disk controllers."

The DECUS Management Council has appointed a task force on
"Commercialism". I volunteered and was appointed the SIG Council
representative. I've heard many of you say, "This kind of thing
shouldn't be in the Exhibit Area!" and "They shouldn't permit
that in the newsletters!" I have too. Now is the time to
generalize and write a policy. Share your ideas with me. It is
harder to do than I thought. You needn't be formal, just put
pencil to paper. For example, "What do you think of the articles
in the PC SIG Newsletter that had the prices 'whited out'?"
"Should other companies be invited to the Exhibit Area?" "Did you
like the article in the DAARC Newsletter for January that looked
very much like an advertisement?"

Do you like the unified format of the newsletters? Or did you
like the separate issues better? A post card: "The new format
is better/worse" to me at P.O. Box 322, Flourtown, PA 19031
would be helpful. Thanks.

When all is said about the Symposium, and thanks offered to the

speakers who spent the time preparing their session to share their experience, and I get through talking about the pre-symposium seminars, it is very important to remember the invisible job of Symposium Coordinator for the SIG. For us that is Skip Stanfield (USAF, Washington). He collects all the papers that are spontaneously submitted in response to the Call for Participation. Then he tries to cajole people to give sessions that he thinks would be valuable. Then the SIG business sessions. Much of this work is done in the background at the symposia - getting ready for the next one while everyone is engrossed in this one! Then there is a lot of work on DCS (the DECUS e-mail system) to make sure the details are completed - who needs an overhead projector, who needs a slide projector, how many seats and who's going to chair the session, etc. Then, there are several days in Marlboro, Massachusetts when the actual scheduling takes place with the Symposia Representatives of the other SIGs. The actual rooms and time slots are assigned in a process that I just don't understand! Skip has done this job quietly and well and is doing it again for Dallas - Thanks, Skip - it wouldn't happen for IAS without you.

Thanks to each of you who have participated in the IAS SIG activities for the past year. May you continue to do so,

Happy New Year,
Bob Curley

I received exactly zero (as in no) entries for the "Find the Hidden Feature" contest. This was quite disapointing. Is there no one out there playing with IAS? I am sure Mike and Alison would be glad to keep the contest open longer if anyone has any entries.

In the next couple of months Decus will be holding elections for the Board of Directors. For those of you unfamiliar with the Decus hierarchy, the Board is the group which sets policy for the society. If you are unhappy (or happy) with the way the society is being run, this is your chance to put in your say. In other words, please vote.

Earlier this year we arranged to get some Cross pens made with an IAS logo, which would be given to contributors to the DeVIAS Letter. These are chrome pens with an IAS logo (reminiscent of the circular pin with red border) on the clip. The pens were made and shipped to Decus Central by Cross. Decus Central shipped them UPS to Bob Curley's PO box. Since UPS and the USPS do not talk, the pens have been lost in shipment. We hope that they will be located by the time that you read this, so that we can honor those who contribute to the DeVIAS Letter.

Speaking of contributions, you will have noticed that this issue and the previous are slim. The newsletter only has material if you contribute. So please jot down your comments, feelings, or anything that moves you and send them to me at:

John Roman
McDonnell Douglas Corp - Dept. N436
600 McDonnell Blvd.
Hazelwood, Missouri 63042

"LEVERAGE"
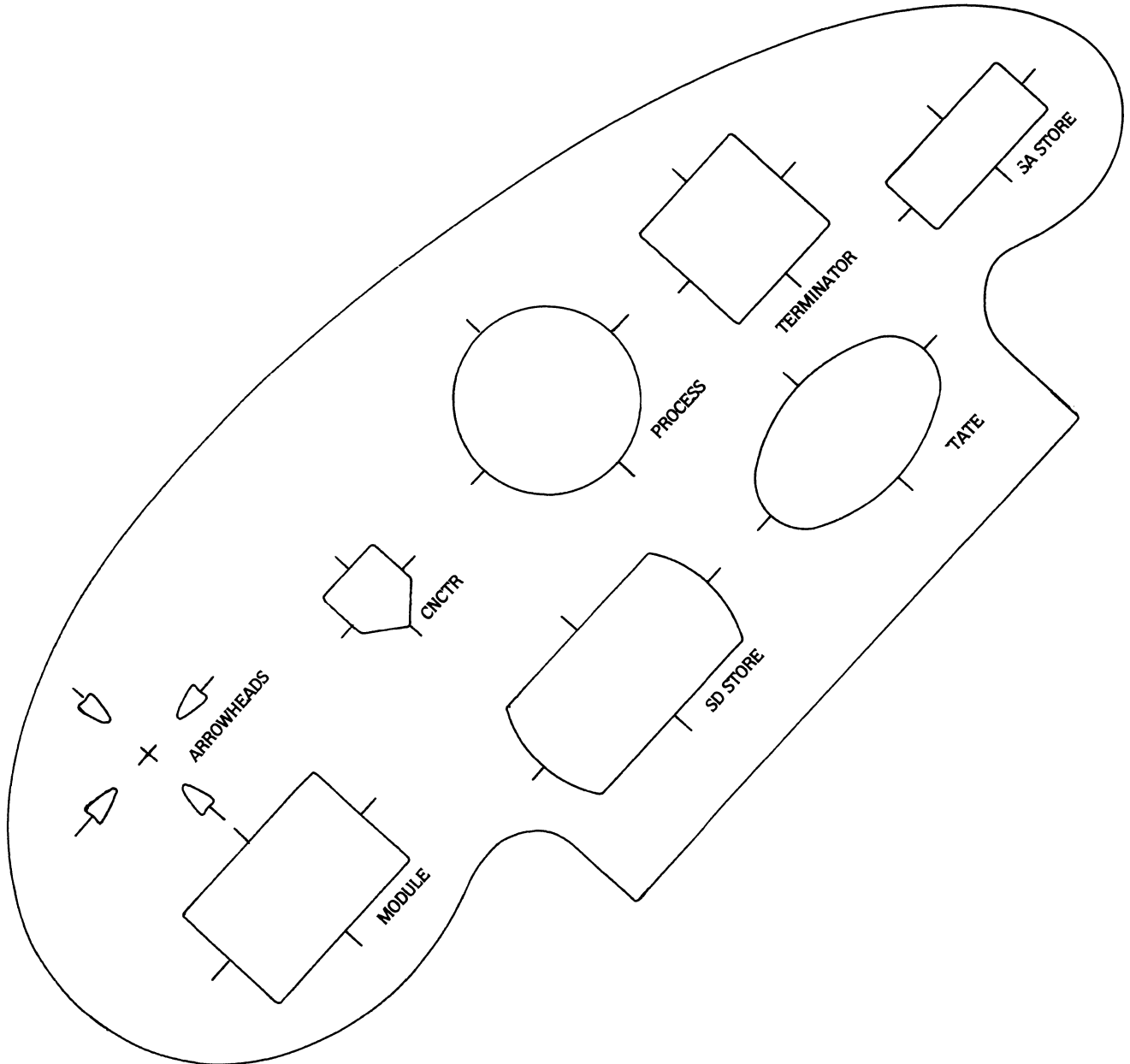
LANGUAGES AND TOOLS SIG

LANGUAGES & TOOLS

Chairperson
Katherine Hornback
Lear Siegler, Inc.
Grand Rapids, MI

Vice Chairperson
Barbara Chase
Hughes Aircraft Company
Los Angeles, CA

Symposium Coordinator
J.R. Westmoreland
Utah Power and Light
Midvale, UT

Asst. Symp. Coordinator-Campground/Suite
Melody Westmoreland
Midvale, UT

Session Chair Coordinator
Earl S. Cory
Eaton Information Management
Westlake Village, CA

Newsletter Editor
Alan L. Folsom, Jr.
Fischer & Porter Co.
Warminster, PA

Session Note Editor
Mark Katz
GTE Government Systems
Needham, MA

Tape Librarian
Dave Martin
Hughes Aircraft Co.
Los Angeles, CA

GAPSIG Standards Coordinator
Jim Flatten
Ames Lab
Ames, IA

Standards Coordinator
FORTRAN Coordinator
Jay W. Wiley
Bechtel Power Corp.
Norwalk, CA

DEC Counterpart
PDP-11
Joe Mulvey
Digital Equipment Corp.
Nashua, NH

VAX
William Segal
Digital Equipment Corp.
Nashua, NH

UNIX Coordinator
Rod Creason, Jr.
Sacramento, CA

MODULA II Coordinator
Jack R Davis
Philips Home Interactive Systems
Knoxville, TN

Ada Coordinator
Bob Gable
Lear Siegler
Grand Rapids, MI

Wishlist Coordinator
Alan Rizzuto
EMC Controls, Inc.
Cockeysville, MD

PASCAL Coordinator
Derek R. Smith
Wilmington, DE

RSTS Coordinator
George Stuart
Ogden - Weber/AVC
Ogden, UT

Methods Coordinator
PSS Committee Rep.
Pat VanMunn
Measurex, Inc.
Cupertino, CA

TeX Coordinator
Large System Liaison
Store Coordinator
Sam Whidden
American Mathematical Society
Providenc, R.I.

CMS/MMS Coordinator
Edgar Whipple
Lawrence Berkeley Lab.
Berkeley, CA

RT-11 Coordinator
Michelle C. Wong
Sacramento, CA

DMS Liaison
Keith Hare
(Home)
Granville, OH

AI SIG Liaison
Don Rosenthal
Space Telescope Science Inst.
Baltimore, MD

CL SIG Liaison
Jim Wilson
Pfizer Inc.
Terre Haute, IN

VAX/VMS Liaison
Louise A. Wholey
Measurex Corp.
Cupertino, CA

DEC Personnel Coordinator
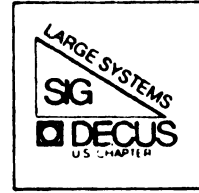Howard Holcombe
Medford, NJ

EX-Chairman
James W. Livingston, Jr.
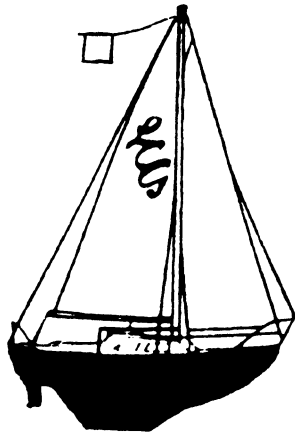Measurex Corp.
Cupertino, CA

Member-At-Large
Shava Nerad
Telematic Systems
Cambridge, MA

RSX Liaison Committee
Anthony E. Scandora, Jr.
Argonne National Laboratory
Argonne, IL

DIGITAL EQUIPMENT COMPUTER USERS SOCIETY

The Newsletter of the Large Systems SIG

LARGE SYSTEMS

SIG Chairperson
Leslie Maltz
Stevens Institute of Tech.
Hoboken, NJ

Symposium Coordinator
Robert C. McQueen
Stevens Institute of Technology
Hoboken, NJ

Newsletter Editor
Michael Joy
First Church of Christ, Scientist
Boston, MA

Menu Coordinator
Charles R.T. Bacon
National Institutes of Health
Bethesda, MD

Hardware Coordinator
Clive Dawson
MCC
Austin, TX

Languages Coordinator
David Edwards
DPEX, Inc.
Fremont, CA

TOPS-20 Coordinator
Pete Gavin
University of Texas at Austin
Austin, TX

Networks Coordinator
Richard Janick
Abbott Laboratories
Abbott Park, IL

Systems Software Coordinator
Betsy Ramsey
American Mathematical Society
Providence, RI

Special Projects Coordinator
Osman Ahmad
Assoc., of American Railroads
Chicago, IL

Special Projects Coordinator
F. Berkeley Shands
Emerson Electric Co.
St. Louis, MO

DEC Counterparts
Dave Braithwaite
Digital Equipment Corporation
Marlboro, MA

Jack Buckley
Digital Equipment Corporation
Marlboro, MA

Reed Powell
Digital Equipment Corporation
Marlboro, MA

Chairperson's Article

Now that the Fall Symposium is a thing of the past, it would probably be worthwhile to share some observations with those of you who could/did not attend. Some of these observations are what could be called readings of the pulse of those who attended, and others are specific items that might be of importance to all.

It has been two and a half years since that fateful day when we all heard that Jupiter was not to be. That means that we are roughly halfway through the promised five years of active development for our thirty-six bit systems. During the past 2 1/2 years we have all had time to figure out what course we are planning to take as our systems age. For some SIG members, the decision has been to keep their systems running as long as electrons flow. Others have started to acquire VAX systems as suggested by DEC. Still others have chosen to integrate, but not with DEC products. In any case, the emotional component of the decision seems to be mellowing. Fewer emotionally-charged comments are being made at symposia, although the needs of each segment of the community still need to be heard. We think our SIG continues to be a viable forum for each such segment, and our activities will be oriented toward meeting the needs of each group.

We will not give up supporting traditional activities for those sites who have chosen to stay with their 10s and 20s. There is still much that this group will need in the way of features in final releases, as well as commitment and delivery of adequate levels of hardware and software support. Similarly, we will support activities and services to meet the needs of those SIG members who have opted to introduce VAX systems, particularly oriented around the high end products. We know that a substantial number of you have chosen this path, and have expressed common needs based on our high end environment. Beginning with the Spring Symposium in Dallas we will begin to attempt to address some of these issues.

To summarize some of the comments made by Rich Whitman at the Product Panel session, I'll list some of the specific topics.

LCG Support:

There has been no slippage of LCG trained engineers for hardware support.
LCG engineers are being trained on 8600's.
Multi-year contracts for support are being written.

Training courses for customers are continuing. There will be 6-7 courses per quarter.

KL to 8600 Trade-in Offer:
Only LCG customers are eligible.
Trade-in is for a KL model B cpu with 512 KW and an RP06.
Allowance is $90,000 credit on an 8600, with return of the KL six months after delivery of the 8600.
1 KL cpu trade-in per 8600 cpu.
VMS, VAX FORTRAN, and VAX COBOL licenses are included.

Unbundled TOPS Licenses:
TOPS licenses will be available for sites with at least one existing
license. This will enable existing sites to run TOPS on non-DEC
processors.
No support will be provided on non-DEC systems.
Sources are included.
Non-educational license costs are $80,000 for the first license;
$40,000 thereafter.
Educational license costs are $40,000 for each license.

Based on the results of our Menu (published previously), the following
directions for VMS were noted as needs of our community:

        Operator capabilities

        Tape handling improvements

        Security oriented features
                Commercial requirements
                D.o.D. needs

        Data integrity
                Journaling
                Checkpointing

On a fun note, our VAXBUSTER shirts were a terrific hit in Anaheim.
Every single shirt was sold, with most sold the first day. We expect to
have a special item in the DECUS Store in Dallas, so plan to get yours.
See you after the snows.

-----Leslie Maltz

From the Editor

Well, I hope everyone enjoyed Anaheim. I know I loved everything except
the cold weather. I didn't plan on temperatures in the 40s and 50s for
sunny Southern California. I'm sure that Dallas will be warmer! Speaking
of Dallas, please give us your menu items as soon as possible so we'll
have a full list before the Dallas symposium. Be sure to read the article
entitled "YOUR INPUT" which appears in this same newsletter.

Back to Anaheim for a minute. During the Question and Answer period at
the opening Large Systems session, the one topic which came up more than
any other was the DECsystem10/20 field service support level. There were
numerous comments about problems not fixed for days, of parts shortages,
and the like. If there is a general problem, or even a few isolated
cases, they should be addressed. One way that can be done is by sending
your comments to the newsletter for publishing, or just a problem tally.
If you have recently had a field service problem, jot down what happened
and send us your report. We will compile a list of problems and let the
SIG members know the results via the newsletter. The list will also be
shared with DEC so that can know the extent of the problems and directly
address the issues. If there are concerns other than field service
problems let us about them too. Address your letters to the editor at the
address listed below.

Recently, we received a copy of the European 10/20 SIG menu items for
1985. It seemed like it might be interesting to compare the top ten items
from Europe compared to the U.S. menu. Here are the results:

European Menu
    1. Create a GALAXY-like environment for VMS
    2. Archiving facility for VMS
    3. Continue TOPS enhancements
    4. Include temporary NI for the KL/8600 trade-in offer
    5. Extend the KL/8600 trade-in offer to other parts of the system
    6. Offer VMS courses for those with TOPS backgrounds
    7. Extend the KL/8600 trade-in offer beyond July 1986
    8. License TOPS for third party hardware at a reasonable price
    9. Shorten the delivery time for TOPS and VMS manuals
    10. Reduce cost of ownership on multi-system sites--especially soft-
        ware

U.S.
    1. Create a GALAXY-like environment for VMS
    2. Support a foreign tape utility for VMS
    3. Create a COMND JSYS equivalent as part of VMS
    4. Provide time stamps in VMS batch log files
    5. Improve tape/disk management in VMS
    6. When TOPS-20 development ends, provide all sources to customers
    7. Implement COMPIL-class commands in VMS
    8. Extend the KL/8600 trade-in offer beyond July 1986
    9. Implement a TOPS-20-like CLI for VMS
    10. Continue providing TOPS training courses.

Note that the first item in both lists was the same. Most of the items in
the European list appeared in the top 20 or so items in the U.S. list
showing a relatively high correspondence between the two menus.

Don't forget to send in your questions to Doctor TOPS! The good Doctor
will soon be out of questions to respond to so write soon. Here's where
to address any questions or comments:

    Michael D. Joy, Editor
    The First Church of Christ, Scientist
    Christian Science Center A41
    Boston, MA 02115

Your Input

DEC Needs User Input on TOPS Enhancements

We are now two and a half years into the final five year development
period for TOPS-10/20. DEC has just released a major new version of
both operating systems: TOPS-10 V7.03 and TOPS-20 V6.1. They are now
in the planning stage for the next releases. Given the traditional
life cycle for new versions, it is a fair guess that the next release
of TOPS-10 and TOPS-20 will also be the last.

DEC is now planning what features they intend to implement in the next
release of TOPS-10 and TOPS-20, so the time is right for us to make
our "wish lists" known. This may be our last chance to get all those
little features we've been thinking about over the years into TOPS.

The Large Systems menu (wish list) is usually compiled from
suggestions users make at the fall and spring DECUS symposia. We know
that many of you cannot attend the symposia. Because it is so
important to get our requests concerning the next TOPS releases to
DEC, we'd like to start collecting menu items early this year, and ask
everyone, including those who cannot attend the spring symposium, to
participate.

If you have enhancements or fixes you'd like to see in the next
release of TOPS-10 or TOPS-20, write or call Betsy Ramsey at

        American Mathematical Society
        P.O. Box 6248
        Providence, RI 02940
        401-272-9500 x295

She will record your suggestions, and pass them on to Chuck Bacon, who
is the Large Systems SIG Menu Coordinator. Chuck will be putting
together the full menu. With your help, we hope to be able to mail
the menu to all members of the Large Systems SIG for voting very soon
after the spring DECUS.

DEC takes a strong interest in our menu, and responds directly to the
top items. Please make your needs known.

-----Betsy Ramsey   LS-4

Alternative Strategies

NEWS RELEASE

FIRST CUSTOMER INSTALLATION OF SC-30M*

November 18, 1985

SAN FRANCISCO, CA--Systems Concepts, Inc. today announced the first customer installation of its SC-30M computer system at the LOTS Computer Facility, Stanford University.

The SC-30M is a high-performance 36-bit computer system which is user-program-compatible with Digital Equipment Corp. DECsystem-10** and DECSYSTEM-20** products. Using software enhancements from Systems Concepts, the SC-30M can run the TOPS-10** and TOPS-20** operating systems with full user-mode compatibility.

The LOTS facility is Stanford's academic computer center, providing the computer support for instruction and for unsponsored research. Ralph E. Gorin, Director of LOTS, said, "The SC-30M is an important augmentation to our interactive capacity."

In its current configuration, the SC-30M system at LOTS uses an IBM*** 3380 disk storage module, with its associated control unit, for on-line data storage, and an IBM magnetic tape subsystem for backup and interchange purposes. To communicate with user terminals, the SC-30M employs the widely-used TCP/IP protocol on an Ethernet**** network. According to Peter R. Samson, Director of Marketing for Systems Concepts, "This installation demonstrates some of the major strengths of the SC-30M system: standard interfaces, higher performance, and program compatibility."

*   SC-30M is a trademark of Systems Concepts, Inc.
**   DECsystem-10, DECSYSTEM-20, TOPS-10, and TOPS-20 are
            trademarks of Digital Equipment Corp.
***   IBM is a trademark of International Business Machines
            Corp.
**** Ethernet is a trademark of Xerox Corp.   LS-5

Doctor TOPS

Dear Dr. Tops:
       I would like to allow selected users to change their UIC
codes so that they can better share files. However, I don't want any bimbo
to have SETUIC privs. Can you PLEASE help me?

                    Positive Paronoia

Dear PP,
       I highly recomend that you investigate ACL's for your VAX. If you
don't have 4.x, get it. Changing the UIC code on the VAX can lead to many
headaches, mainly because the VMS developers never thought ahead when
they wrote the code. Among other things this will break is PDP-11
compatibility mode (The VAX AME product, VAXRSX11). I will provide
this super hack to you with MANY WORDS OF WARNING; Under 4.0 of
VMS, this will CRASH YOUR VAX.

                    I disown this hack

                    Dr. Tops

```
! Command file for UGROUP.EXE which changes the left half of the UIC
!
! USER [abcd,efgh]
!
$ current_uic = f$user()
$ if P1 .eqs. "" then inquire P1 "New UIC or GROUP?"
$ if P1 .eqs. "" then P1 = current_uic
$ request_uic = P1
$ run sys:ugroup
$ P2 = f$user()
$ P1 = f$directory()
$ write sys$output "#6#3", P1, P2
$ write sys$output "#6#4", P1, P2
```

```
$ comp setuic.mar,ugroup.for
$ link/notrace_back ugroup,setuic,sys:sys.stb
$ copy ugroup.exe sys$sysroot:[hacks]
$ wv sys:ugroup.exe
$ r sys:install
   sys:ugroup/replace
$! sys:ugroup/priv=(cmkrnl,bypass)
$ purge sys:
$ purge
$ exit
```

```
        program ugroup
c
c       this program changes the left half of the UIC word for each user
c
c       a list of valid left half words (16 bits, for V4.0) is kept
c       in sys:ugroup.dat by right half of the uic
c
c       format of the file.....
c
c       UIC_Right_half,list of possible left halves, with commas
c
c       000020,000020,000001,000100
c
c       Note the 16 bit OCTAL format!!!!!
c
c
c       USER [ABCD,EFGH]
c       USER name
c
c
c       symbols used:
c       current_uic - UIC before changes
c       request_uic - UIC after changes or group ID
c       * Note: only left half changes
c
c
        implicit integer (a-z)
        character*32 myuic,newuic
        character*32 group_names(35),Group_uic(35)
        include '($SSDEF)'
c
        data group_names/
     1  'Other','Other','Other','Other,'Other',
     1  'Other','Other','Other','Other,'Other',
     1  'Other','Other','Other','Other,'Other',
     1  'Other','Other','Other','Other,'Other',
     1  'Other','Other','Other','Other,'Other',
     1  'Other','Other','Other','Other,'Other',
     1  'Other','Other','Other','Other,'Other'/
C
        data group_uic/
     1  '[1,4]','[20,100]','[100,100]','[101,100]','[102,100]',
     2  '[103,100]','[104,100]','[105,100]','[106,100]','[107,100]',
     3  '[110,100]','[111,100]','[112,100]','[113,100]','[114,100]',
     4  '[115,100]','[116,100]','[117,100]','[120,100]','[121,100]',
     5  '[122,100]','[123,100]','[124,100]','[125,100]','[377,100]',
     6  '[126,100]','[127,100]','[130,100]','[131,100]','[132,100]',
     7  '[133,100]','[134,100]','[135,100]','[136,100]','[137,100]'/
c
c       open the data file
c
        open(unit=1,name='sys:ugroup.dat',readonly,shared,
     1  status='old',err=900)
c
c       get my current UIC
c
        iret=lib$get_symbol('current_uic',myuic)          LS-7
        if(iret.ne.ss$_normal)goto 800
```

```
c
c         get requested UIC
c
          iret=lib$get_symbol('request_uic',newuic)
          if(iret.ne.ss$_normal)goto 800
c
c         check for group name before [xxx,yyy]
c
          if(newuic(1:1).eq.'[') goto 2
c
c         lookup name in table
c
          call lcuc(newuic)
c
          do 3 igc=1,35
             if(newuic.eq.group_names(igc)) then
                newuic=group_uic(igc)
                goto 2
                endif
3         continue
c
c         Uic is not correct
c
          type *,newuic, ' is not recognised'
          stop
c
2         continue
c
c         convert to integers
c
          call uic2int(oldgrp,oldusr,myuic)
          call uic2int(newgrp,newusr,newuic)
c
c         find oldusr in sys:ugroup.dat
c
          call finduser(oldusr,newgrp,ok)
          if (ok .ne. ss$_normal) goto 1000
          uic=newgrp*2**16+oldusr
          call setuic(uic)
          call exit
900       continue
          type *,' Unable to find UIC file'
          stop
800       continue
          type *,' DCL symbols not available'
          stop
1000      continue
          if(ok .eq. -1) type *,' Current UIC not on file'
          if(ok .eq. -2) type *,' Error reading UIC file'
          if(ok .eq. -3) type *,' Not a valid UIC'
          stop
          end
c
c         routine to make integers from ascii
c
          subroutine change(number,chars)
          character*(*) chars
c
c         input is in octal
c
```

```fortran
      number=0
      length=len(chars)
      do 20 i=1,length
      j=ichar(chars(i:i))-ichar('0')
      number=number*8+j
20    continue
      return
      end
c
c     subroutine to convert character uic into numbers
c
      subroutine uic2int(usr,grp,string)
      implicit integer(a-z)
      character*(*) string
c
c     isolate fields of uic [AAAA,BBBB]
c
      g1=2      !left group
      g2=2      !Right group
      u1=32     !left user
      u2=32     !right user
c
c     check for leading "["
c
      if(string(1:1) .eq. '[') goto 9
      type *,' Error: no leading bracket in UIC'
      stop
9     continue
c
c     scan for ","
c
      do 10 g2=2,32
         if(string(g2:g2).eq.',') goto 11      !stop on comma
10    continue
      type *,' Error: no comma in UIC'
      stop
11    continue
c
c     scan for right "]"
c
      g2=g2-1                    !skip over comma
      u1=g2+2                    !sam ting
      do 12 u2=u1,32             !scan it all
         if(string(u2:u2).eq.']') goto 13
12    continue
      type *,' Error: no ending bracket in UIC'
      stop
13    continue
      u2=u2-1                    !back up over "]"
c
c     convert into a real number
c
```

```fortran
                call change(usr,string(g1:g2))
                call change(grp,string(u1:u2))
                return
                end
c
c       subroutine to locate and verify a new UIC group
c       given user #
c
                subroutine finduser(olduser,newgroup,error)
                implicit integer (a-z)
                integer clist(10)
                include '($SSDEF)'
c
c       assume ok
c
                error = ss$_normal
c
c       read the file until uicgroup is found, or # too big or eof
c
                do 30 i=1,200
                read(1,100,end=200,err=201)clist
100             format(10(o6,1x))
                if(clist(1).eq.olduser) goto 400          !found it
30              continue
c
c       here on error
c
200             continue
                error = -1       !eof
                return
201             continue
                error = -2                !you lose
                return
400             continue
c
c       search for new uic group
c
                do 401 i=2,10
                if((clist(i) .eq. newgroup) .and. (clist(i) .ne. 0)) return
401             continue
                error = -3
                return
                end
c
c       routine to convert lower case into upper case
c
                subroutine lcuc(string)
                character*(*) string
                character*1 k
                i=len(string)
                do 10 j=1,i
                   k=string(j:j)
                   if((k.ge.'a').and.(k.le.'z')) then
                      string(j:j)=char(ichar(k)-(ichar('a')-ichar('A')))
                      endif
10              continue
11              continue
                return
                end
```

```
        .title   setuic set user UIC based on user #
        .subtitle Dr. TOPS @VMSLand Center

        .ident  /1-001/

;       this routine changes the user UIC string
;       Beware, this breaks AME products (PDP-11 compatibility)
;
;       requires CMKRNL priv to function
;
;       input: longword UIC
;
;       output: the changed P1 and system space user UIC
;
        .library          /sys$library:lib.mlb/
        $jibdef           ;JIB symbols
        $pcbdef

        .psect  _lib_code, pic,usr,con,rel,lcl,shr,exe,rd,nowrt


;
;       starting address of procedure
;
        .entry  setuic, ^M<r3,r4,r5,r8>

        movl    @4(ap),r8                       ;put UIC into R8
        $CMKRNL_S routin=writeuser              ;enter kernel mode
10$:    ret                                     ;thats all folks


;
;       kernel mode routine
;
        .entry  writeuser,0                     ;save nothing
        movl    ctl$gl_pcb,r0                   ;move pcb addr into r0
        movl    r8,pcb$l_uic(r0)                ;save uic in memory
        movl    #1,r0                           ;return success
        ret                                     ;thats all

        .end
```
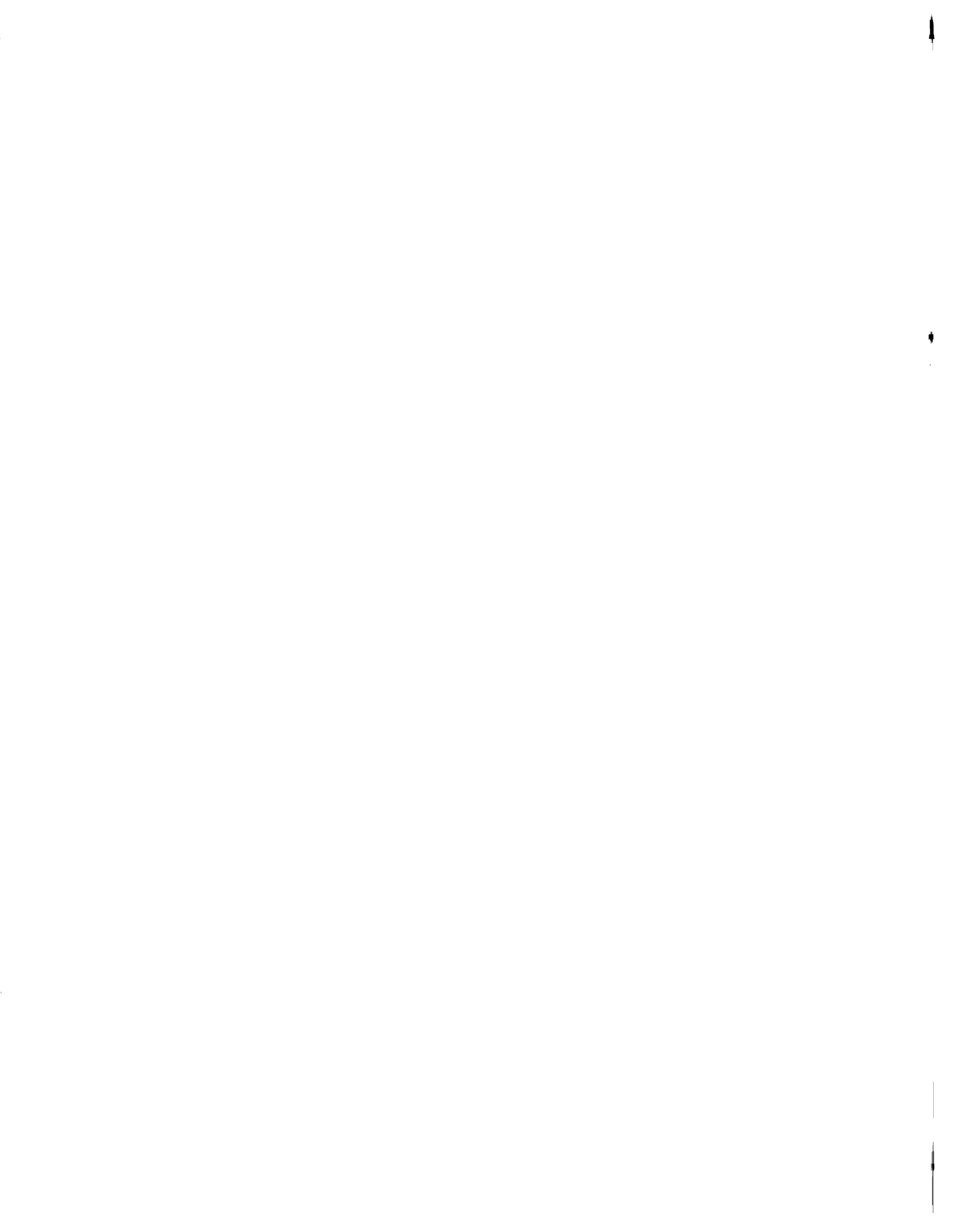
MUMPS

MUMPS SIG STEERING COMMITTEE

Chairman
Mark Berryman
Plessey Peripheral Systems
Irvine, CA

Symposium Coordinator
Chris Richardson
Computer Sciences Corp.
Ridgecrest, CA

Communications Rep.
Mark Hyde
Advanced Computing Services
DeWitt, NY

Newsletter Editor
Janet Berryman
Plessey Systems, Inc.
Irvine, Ca

VAX Liaison
Coyett A.J. Dese
VA DM&S Verification & Dev. Ctr.
San Francisco, CA

Digital Counterparts
Beatrice Walther
Digital Equipment Corp.
Marlboro, MA

Diane Brown
Digital Equipment Corporation
Marlboro, MA

NETwords

The NETWORKS SIG Newsletter
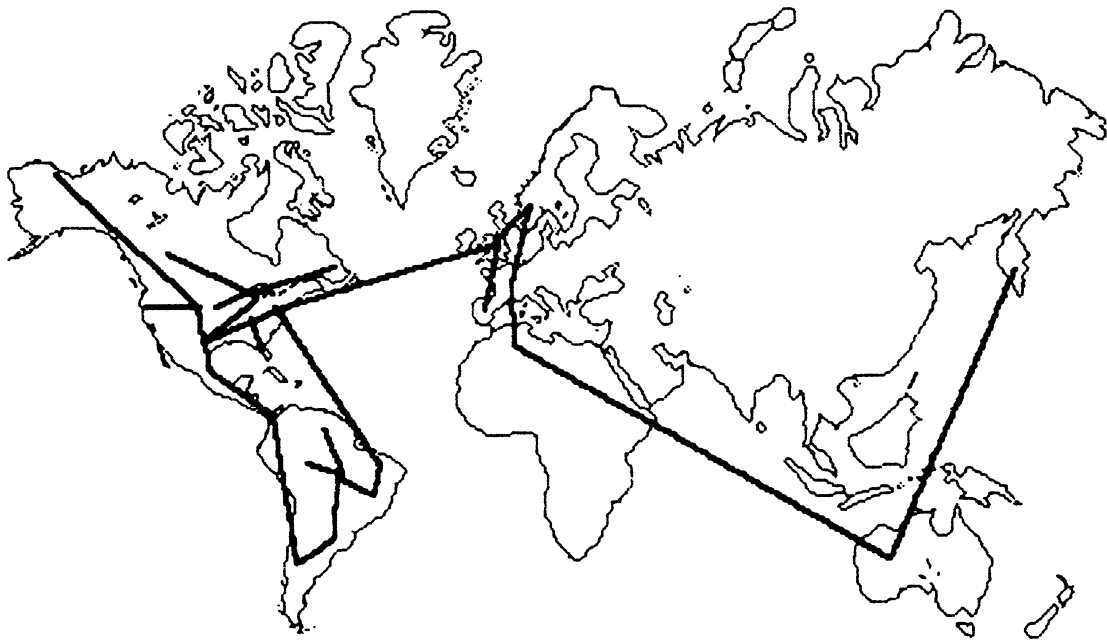
Networks SIG Steering Committee

Bill Brindley
Chairman
Naval Security Group Command
(202) 282-0527

Sandy Traylor
Symposia Coordinator
Target Systems, Inc.
(714) 921-0112

Jim Ebright
Communications Coordinator
Software Results Corporation
(614) 421-2094

Bill Hancock
Technology/
  Standards Coordinator
(214) 495-7353

Vickie Hancock
Newsletter Editor
(214) 495-7353

Carol Greenfield
DEC Counterpart
Digital Equipment Corporation

The Networks Special Interest Group (SIG) is one of 25 SIG's within in Digital Equipment Computer User's Society (DECUS). The main purpose of the Networks SIG is to promulgate information concerning the use, development, and standardization of network products that function or involve Digital Equipment Corporation systems. Additional functions of the SIG include the coordination and scheduling of symposia sessions, providing methods for free-flow communications, publication of the Networks SIG newsletter NETWords, participation in domestic and international standards committees, input to Digital for new products and corrections to existing products, promotion of working groups for special network needs and topics, and many, many other functions.
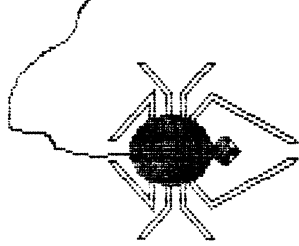
The Networks SIG Steering Committee invites you to participate in the Networks SIG. There are many ways that you can help the Networks SIG. Some of those include chairing sessions at symposium, participation in the various Networks SIG working groups, participation in special research projects, and others. If you are interested in devoting your time and expertise, contact any of the steering committee members.

DECUS is run entirely by volunteer leadership. Help us make DECUS and the Networks SIG better - take an active part in **your** SIG!

In this Issue:

The Editor's Cobwebb

By Vickie Hancock

Welcome back, to those who went, from Anaheim! I sincerely hope that you did not get as cold as I out there. I was fortunate enough, however, to purchase a nice, warm, Mickey Mouse jacket, so I managed to survive. Warm California, phooey!

In this issue we have an enlightening article on how to use Ethernet in a native manner (without DECnet) utilizing the QIO interface on VMS and P/OS. For those of you thinking about using the Ethernet as strictly a data transfer method, this may be for you. We also have an article from Bill Hancock on the basics of network design and analysis for you folks out there thinking about getting into networks. We had a lot of questions about that at Anaheim, so Bill whipped out his pen (or, should I say, Mac), and put together some thoughts for you to enjoy.

It's hard to believe, but Dallas is right around the corner! As a result, we will be putting together a special anniversary issue of NETWords, complete with useless (and useful) trivia and who knows what else. Your contributions are desired and we need them FAST. Any item(s) you would like to contribute are welcome, so get out that pen and WRITE NOW!!

See you next issue!!

VH

Native Mode ETHERNET Communications
Thomas Turano
Roman Pinsky
Laboratory Data Products
Digital Equipment Corporation
Marlborough, Massachusetts

Introduction

        Recently we were involved in testing a remote processor
which responded to ASCII commands sent over the ETHERNET.   This
processor did not use DECnet.   To simplify   our   testing
procedures,   we constructed an interactive interface which
allowed a user to type a command at a terminal on a host
computer and have that command transmitted to the remote
processor. The remote processor's ASCII response to the command
was then delivered to the user's terminal on the host computer.

        It was necessary to test the remote processor from both
a VAX and a PRO host, so an interactive interface was built for
both the VMS and the POS operating systems. Because the remote
processor did not use DECnet, communications using the ETHERNET
meant that QIO requests were necessary. QIO requests are not as
straightforward as DECnet calls, and we felt that it might be
beneficial for the readers of NETWORDS to examine how VMS and
POS use QIO requests to communicate over the ETHERNET. For the
purposes of this article, we have taken the POS and VMS
interactive interfaces and modified them to communicate with
each other rather than with a remote processor. In doing so, we
hope that the mechanisms by which native mode communications
over the ETHERNET can be accomplished will become clearer.

        A synchronous communications protocol was used in this
application.   That is, the host node always initiated
communications by transmitting a command, and always received a
reply from the remote node before sending another command. In
the example in this article, the VAX node is the initiator of
the communications with the PRO. The synchronous protocol is
used in an attempt to maintain data integrity.

        Communications protocols in general are complicated
schemes for maintaining a link between computers and detecting
when a link has been broken. If an established protocol (such
as DECnet) is not used, it is the responsibility of the programs
which are conversing to detect a communications failure and take
appropriate action.

        When the communications are synchronous, the initiator
of the conversation can detect the loss of data. If the reply
message from the remote node is lost (for example, due to
multiple collisions on the ETHERNET), the initiator will timeout
for failing to receive a reply to its command.   If the
communications protocol were asynchronous, a more complicated

scheme would be required to detect the loss of a message, since either node could have sent a message at any time. This article is meant to demonstrate the method by which systems can access the ETHERNET using QIO requests; a discussion of communications protocols is well beyond its scope.

ETHERNET Overview

The ETHERNET transmits data as datagrams. Each datagram consists of a 46- to 1500-byte message preceded by a header and followed by a trailer. The header of the datagram is made up of:

o   an eight-byte preamble

o   a six-byte destination address

o   a six-byte source address

o   a two-byte protocol code

The trailer consists of a four-byte cyclic redundancy check (CRC) field that is used to determine if the data was corrupted during transmission. The preamble and the CRC fields are inserted/removed by the hardware prior to transmission/upon reception of the message.

Each node on the ETHERNET has a unique address which is used in the header. Each node listens to the ETHERNET. When a node determines that a datagram has its address as the destination address, the node checks the CRC portion of the message. If the CRC shows the message is undamaged, the message is passed to the user.

When a node wishes to transmit a message, it first listens to the ETHERNET. If no other node is transmitting, the node can begin to transmit. If two or more nodes begin to transmit simultaneously, a collision occurs, and all the transmitting nodes immediately cease transmitting. A random amount of time is then allowed to lapse before each node again attempts to transmit.

At any given time, more than one user on one node may wish to communicate with more than one user on a second node. There must be a means of determining which user on a receiving node a given datagram is sent to. The protocol code portion of the datagram accomplishes this "demultiplexing" of datagrams. Each node on the ETHERNET creates a table of protocol type-destination address pairs, and only one user on that node can use an address pair at any given time.

For example, if user A on node ALPHA using protocol type 01 establishes a connection with node BETA, then user B on node ALPHA is not allowed to establish a connection with node BETA using protocol type 01. User B has to use another protocol

type. Protocol types have been assigned under license by XEROX to the various manufacturers of ETHERNET products. Digital Equipment Corporation has been assigned a block of protocol types for its various products (for example, DECnet, LAT), and has in turn reserved the protocol type 60-06 (HEX) for its users who wish to use native mode ETHERNET communications. This is the protocol type that the example programs use.

VMS NODE

The ETHERNET program on the VMS node consists of three routines, each of which is made up of several modules. The main program is a FORTRAN routine called NATIVEMODE, which contains two FORTRAN modules. The first is a function called DEV_TRANSLATE, and the second a subroutine called TRANSACT. DEV_TRANSLATE converts the logical device assignment given to the receiving node to its ETHERNET physical address. TRANSACT is the subroutine which controls the I/O to the ETHERNET.

A group of three FORTRAN functions make up the EIO routine. The first of these functions, ETHERNET_ATTACH, is called directly by the main program NATIVEMODE. It assigns a channel to and sets the characteristics of the ETHERNET device. The other two FORTRAN functions perform the QIO requests to WRITE/READ to/from the ETHERNET. These two functions are called by the subroutine TRANSACT.

Finally, there is a MACRO routine ESET, called by ETHERNET_ATTACH, which actually sets the ETHERNET device parameters. This routine was written in MACRO because the parameters can be easily assigned to the parameter block using assembly language.

The calltree for the VMS side of the application is:

```
                    NATIVEMODE
                    (FORTRAN)
                  [MAIN PROGRAM]
                        |
      +-----------------+----------------+
      |                 |                |
      |          DEV_TRANSLATE       TRANSACT
      |            (FORTRAN)         (FORTRAN)
      |        [located in NATIVEMODE]   [located in NATIVEMODE]
      |                                   |
ETHERNET_ATTACH                  +--------+--------+
   (FORTRAN)                     |                 |
[located in EIO]          ETHERNET_WRITE     ETHERNET_READ
      |                     (FORTRAN)           (FORTRAN)
      |                  [located in EIO]    [located in EIO]
    ESET
   (MACRO)
```

Calltree for the VMS Program NATIVEMODE

        The programs are listed in the following order,
NATIVEMODE.FOR, EIO.FOR, and ESET.MAR. Also note that the
programs which follow are documented in line.

                        NATIVEMODE.FOR


```
C
C Program NATIVEMODE: interface over an ETHERNET link
C Link with EIO,ESET
C Edits:
C


        IMPLICIT INTEGER (A-Z)

        PARAMETER MAXLEN = 1518

        CHARACTER*1  CARRIAGE_RETURN
        DATA CARRIAGE_RETURN/13/
        CHARACTER*1 cmd(maxlen)
        CHARACTER*1 reply(maxlen)
        CHARACTER*40 device        ! dev name (TTnn or XEnn)
        CHARACTER*256 errmes       ! array to hold error messages
        BYTE unit_address(6)
        INTEGER dev_length         ! length of device name
        INTEGER syschan            ! system channel for ethernet link
        INTEGER i
        INTEGER cmdlen             ! length of data being transmitted
        INTEGER cmd_lun            !
        INTEGER msglen
```

```
              INTEGER replylen          ! length of reply being received
              INTEGER status

              LOGICAL dev_translate
              EXTERNAL dev_translate

              INTEGER ethernet_attach
              EXTERNAL ethernet_attach
C
C
C

              cmd_lun = 20
              OPEN (unit=cmd_lun, FILE='SYS$INPUT:',STATUS='UNKNOWN',
         1       RECL=MAXLEN)
              OPEN (unit=6, FILE='SYS$OUTPUT:',STATUS='UNKNOWN')
C
C Set up for comm with remote node
C Parse logical NET$DEV if not defined inform user and exit
C
              IF (dev_translate ('NET$DEV',device,dev_length,
         1                       unit_address)) GO TO 2
C
C Advise user what is going on
C
              WRITE(6,988)
988           FORMAT(/,' Before running NATIVEMODE, you must assign the',
         1           ' logical NET$DEV to indicate',/,
         1           ' which ETHERNET device YOU want to talk to ',
         1           ' For ETHERNET links, you must issue the command:',//
         1           '     $ASSIGN XEnn:aaaaaaaaaaaa NET$DEV',//,
         1           ' where nn is the DEUNA/DEQNA controller and unit, and
         1               'aa... is the',/,
         1           ' hexadecimal ethernet address of the DECNA in the ',
         1               'SYSTEM unit. For',/,
         1           ' example, to connect to the SYSTEM at address ',
         1               '0100A2010020 on the',/,
         1           ' ETHERNET controller XEA0:, use XEA0:0100A2010020.',/
              CALL EXIT
C
C Announcement
C
2             ESCAPE = 27
              WRITE (6,986) ESCAPE
986           FORMAT(/,' ',A1,'#6LNATIVEMODE ETHERNET example',//,
         1           ' For HELP, type "?" at command prompt.')
C
C Attach ETHERNET device link
C Inform the user that the link is attached if successful
C otherwise inform the user of the error and exit
C
              status = ethernet_attach(device,unit_address,syschan)
              IF (.NOT. status) GO TO 9
              WRITE(6,989) device,unit_address
989           FORMAT(/,' [ETHERNET link ',A<dev_length>,6Z2.2,' Attached]')
              GO TO 10
```

```fortran
9             CALL sys$getmsg(%VAL(status),msglen,errmes,,)
              WRITE(6,996) errmes
996           FORMAT(' ?NATIVEMODE-F-OUTFAIL, Attach Failed',/,
     1              ' ',A<msglen>)
              CALL EXIT
C
C Command Loop - print prompt
C Then read the input from the terminal
C
10            WRITE(6,999)
999           FORMAT(/,' Transmit: ',$)
              cmdpos = 1
11            READ(cmd_lun,98,END=900) (CMD(I),I=cmdpos,MAXLEN)
98            FORMAT(<MAXLEN>A1)
C
C Determine length of command typed starting from
C the end and moving back
C
              cmdlen = maxlen
15            IF (cmd(cmdlen) .NE. ' ') GO TO 20
              cmdlen = cmdlen - 1
              IF (cmdlen .GT. 0) GO TO 15
C
C If continuation marker, continue getting command
C
20            cmdpos = cmdlen
C
C Empty commands are echoed, but ignored
C
              IF (cmdlen .EQ. 0) GO TO 10
C
C  Call TRANSACT subroutine to process the message
C
              CALL TRANSACT (syschan,
     1                       unit_address,cmdlen,cmd,
     2                       maxlen,reply,replylen)
C
C Display the reply
C
              WRITE(6,95) (reply(i),i=1,replylen)
95            FORMAT(' Receive:   ',70A1,40(/,10X,70A1))
              GO TO 10
900           CALL EXIT
              END
C
C*************************************************************
C
              LOGICAL FUNCTION dev_translate (logical,device,
     1                            dev_length,ethernet_address)

              INCLUDE '($IODEF)'
              INCLUDE '($SSDEF)'
C
C Args
C
```

```
        CHARACTER logical *(*)
        CHARACTER*1 device(40)
        CHARACTER*1 hchar            ! hex character from address
        CHARACTER*1 phy_device$C(100)
        CHARACTER*100 phy_device
        INTEGER dev_length
        INTEGER enet_addr_index !temps for parsing hex NI address
        INTEGER phy_dev_length
        INTEGER scan
        INTEGER*2 hbyte              ! Accumulator for hex byte
        INTEGER*2 nibble             ! place for arithmetic on hex char
        INTEGER*4 status
        BYTE acmode                  ! access mode for user logical
        BYTE ethernet_address(6)
        BYTE hbyte$B                 ! to copy byte from accumulated hex
        BYTE log_table               ! logical table where NET$DEV found
        BYTE phy_device$B(100)
C
C Equivalence arrays
C
        EQUIVALENCE (phy_device,phy_device$C,phy_device$B)
        EQUIVALENCE (hbyte,hbyte$B)
        EQUIVALENCE (nibble,hchar)
        INTEGER*4 SYS$TRNLOG
C
C Translate the logical device
C Assume failure
C
        dev_translate = .FALSE.
        status = SYS$TRNLOG(logical,phy_dev_length,
        1                   phy_device,log_table,acmode,%VAL(0))
        IF (status .NE. SS$_NORMAL) RETURN
C
C Glean NET device (physical device up through ":"
C
        scan = 0
10      scan = scan + 1
        device(scan) = phy_device$C(scan)
        dev_length = scan
        IF (device(scan) .EQ. ':') GO TO 20
        IF (scan .EQ. phy_dev_length) GO TO 900
        GO TO 10
C
C If anything remains, it must be ethernet address (12 characters)
C
20      IF (scan .EQ. phy_dev_length) GO TO 900
        IF ((phy_dev_length - scan) .NE. 12) RETURN
C
C Convert data from hex
C
        scan = scan + 1                    ! point to first hex character
        DO 100 enet_addr_index = 1,6       ! loop through NI address
        hbyte = 0                          ! init this hex byte, and accumulate..
        DO 50 np = 1,2
            hbyte = hbyte * 16
```

```
                    nibble = 0              ! clear high byte of NIBBLE
                    hchar = phy_device$C(scan) ! load char in low
      C                                         byte of NIBBLE
                    IF ((.NOT. ((hchar .GE. '0') .AND.
              1          (hchar .LE. '9'))) .AND.
              1          (.NOT.((hchar .GE. 'A') .AND.
              1          (hchar .LE. 'F')))) RETURN
                    IF ((hchar .GE. '0') .AND. (hchar .LE. '9'))
              1                 hbyte = hbyte + (nibble - 48)
                    IF ((hchar .GE. 'A') .AND. (hchar .LE. 'F'))
              1                 hbyte = hbyte + (nibble - 55)
                    scan = scan + 1
      50        CONTINUE
                    ethernet_address(enet_addr_index) = hbyte$B
      100       CONTINUE
      900       dev_translate = .TRUE.
                RETURN
                END
      C
      C*****************************************************************
      C
                SUBROUTINE transact (syschan,  drop,
              1                          cmdlen, cmd,
              2                          replysize, reply, replylen)

                INCLUDE '($IODEF)'
                INCLUDE '($SSDEF)'

                PARAMETER r_efn = 1
                PARAMETER e_efn = 2

                INTEGER cmdlen
                INTEGER replylen
                INTEGER replysize
                INTEGER status
                INTEGER syschan
                INTEGER wlen
                INTEGER msglen
                INTEGER*2 r_iosb(4)
                INTEGER*2 e_iosb(4)
                CHARACTER*1 cmd(*)
                CHARACTER*1 reply(*)
                CHARACTER*256 errmes

                INTEGER ethernet_post_read
                EXTERNAL ethernet_post_read

                INTEGER ethernet_write
                EXTERNAL ethernet_write
      C
      C Post read FIRST
      C
                CALL SYS$CLREF(%VAL(r_efn))
                status = ethernet_post_read (r_efn, syschan, reply,
              1                                      replysize, r_iosb)
```

```fortran
        IF (.NOT. status) GO TO 950
C
C Write the command
C
15      IF (cmdlen .EQ. 0) GO TO 35       ! Just post input
        wlen = cmdlen
        status = ethernet_write (syschan, drop, cmd, wlen)
        IF (.NOT. status) GO TO 930
C
C Read the reply
C
35      CALL SYS$WAITFR(%VAL(r_efn))
        status = r_iosb(1)
        replylen = r_iosb(2)
        IF (.NOT. status) GO TO 940
        RETURN
C
C Error
C
930     CALL sys$getmsg(%VAL(status),msglen,errmes,,)
        WRITE(6,99) errmes
99      FORMAT(' ?NATIVEMODE-F-OUTFAIL, Output QIO Failed',/,
        1       ' ',A<msglen>)
        CALL EXIT
940     CALL sys$getmsg(%VAL(status),msglen,errmes,,)
        WRITE(6,98) errmes
98      FORMAT(' ?NATIVEMODE-F-INFAIL, Reply READ Operation Failed',/,
        1       ' ',A<msglen>)
        CALL EXIT
950     CALL sys$getmsg(%VAL(status),msglen,errmes,,)
        WRITE(6,991) errmes
991     FORMAT(' ?NATIVEMODE-F-READFAIL, READ QIO Failed',/,
        1       ' ',A<msglen>)
        CALL EXIT
        END
```

```fortran
      INTEGER FUNCTION ethernet_attach (dev,address,syschan)
      INTEGER eset              !
      INTEGER status            ! Status returned
      INTEGER syschan           ! Channel to the driver
      CHARACTER dev *(*)        ! Device name
      BYTE address(6)
      EXTERNAL eset             ! Module ESET.MAR
      INTEGER SYS$ASSIGN
      EXTERNAL SYS$ASSIGN
C
C Assign a channel to the device
C
      status = SYS$ASSIGN(dev,syschan,,)
      IF (.NOT.status) GO TO 990       ! On error return error
C
C Set the characteristics using the channel
C
      status = eset(syschan,address)
C
C Return the status
C
990   ethernet_attach = status
      RETURN
      END
C
C*****************************************************************
C
      INTEGER FUNCTION ethernet_write (syschan,drop,buffer,length)

      INCLUDE '($IODEF)'
      INCLUDE '($SSDEF)'

      INTEGER length            ! Length of the buffer
      INTEGER status            ! Status of the call
      INTEGER syschan           ! System channel
      CHARACTER buffer *(*)     ! Transmit buffer
      BYTE drop(6)
      INTEGER SYS$QIOW
      INTEGER*2 IOSB(4)
C
C  QIO to write virtual block over the ETHERNET
C
      ethernet_write = SYS$QIOW(,%VAL(syschan),
     1                 %VAL(IO$_WRITEVBLK), %REF(IOSB),,,
     2                 %REF(buffer), %VAL(length),,,%REF(drop),)
      RETURN
      END
C
C*****************************************************************
C
      INTEGER FUNCTION ethernet_post_read (efn,syschan,buffer,size,iosb)

      IMPLICIT INTEGER (A-Z)
```

```fortran
        INCLUDE '($IODEF)'
        INCLUDE '($SSDEF)'

        INTEGER efn              ! Event flag
        INTEGER size             ! Size of buffer to be received
        INTEGER status           ! Call return status
        INTEGER syschan          ! System channel
        INTEGER*2 IOSB(4)        ! I/O status block
        CHARACTER buffer *(*)    ! Buffer to receive message

        INTEGER SYS$QIO
C
C Byte definitions for the received message
C
        BYTE RECPAR(16)          ! Info about packet and sender
                                 ! (1-6) = receiver ethernet address
                                 ! (7-12) = sender ethernet address
                                 ! (13-14) = protocol type
                                 ! (15-16) = pad field
C
C QIO for read virtual block
C
        ethernet_post_read = SYS$QIO(%VAL(EFN),%VAL(syschan),
     1             %VAL(IO$_READVBLK), %REF(IOSB),,,
     2             %REF(buffer), %VAL(size),,,%REF(RECPAR),)
        RETURN
        END
```

```
        .LIBRARY /SYS$LIBRARY:LIB.MLB/

        $IODEF                          ; VAX I/O definitions
        $NMADEF                         ; VAX definitions
        $XMDEF                          ; VAX
                                        ; Config calls
        NMA$C_LINPR_CLR = 2             ;    Symbol not defined VMS V3.
;
; Program constants
;
        bufsiz  = 1500                  ; Ethernet buffer size
        nbuff   = 2                     ; Number preallocated buffers
        maxbuf  = 1500                  ; Maximum buffer size
        protyp  = ^X0660                ; protocol type 0660  HEX
;
; Header size values in bytes
;
        preamble        = 8
        dest_address    = 6
        src_address     = 6
        protocol        = 2

        .PSECT  RWDATA,PIC,NOEXE,LONG
;
;
;
wtext:  .ASCID  'WRITE'                 ; PROMPTS
rtext:  .ASCID  'READ'
;
;
;
parblk: .WORD   NMA$C_PCLI_BUS          ;BUFFER SIZE
        .LONG   bufsiz                  ;/PORT
        .WORD   NMA$C_PCLI_BFN          ;Number of buffs
        .LONG   nbuff                   ;    value = nbuff
        .WORD   NMA$C_PCLI_BSZ          ;Buffer size
        .LONG   maxbuf                  ;    value = maxbuff
        .WORD   NMA$C_PCLI_PRM          ;Promiscuous mode
        .LONG   NMA$C_STATE_OFF         ;    value = disabled
        .WORD   NMA$C_PCLI_MLT          ;Multicast address state
        .LONG   NMA$C_STATE_OFF         ;    value = disabled
        .WORD   NMA$C_PCLI_PAD          ;Pad short buffers
        .LONG   NMA$C_STATE_ON          ;    value = padding enabled
        .WORD   NMA$C_PCLI_EKO          ;Echo mode
        .LONG   NMA$C_STATE_OFF         ;    value = noecho
        .WORD   NMA$C_PCLI_DCH          ;Data chaining
        .LONG   NMA$C_STATE_OFF         ;    value = no chaining
        .WORD   NMA$C_PCLI_CRC          ;Generate CRC
        .LONG   NMA$C_STATE_ON          ;    value = hardware CRC
        .WORD   NMA$C_PCLI_PTY          ;Protocol type
        .LONG   protyp                  ;    value = protyp
        .WORD   NMA$C_PCLI_CON          ;Controller mode
```

```
            .LONG    NMA$C_LINCN_NOR          ;  value = normal
            .WORD    NMA$C_PCLI_ACC           ;Protocol access
            .LONG    NMA$C_ACC_LIM            ;  value = limited
            .WORD    NMA$C_PCLI_DES           ;Shared protl dest addr
            .WORD    PB40-PB30                ;  length parm block following
PB30:       .WORD    NMA$C_LINMC_SET          ;Modifier word
unitad: .BLKB    6                            ;6 Byte dest ETHERNET address
PB40:
pblen=.-parblk                               ;Length of parameter block
pbdesc: .LONG              pblen             ;Descriptor points to param bl
pbaddr: .ADDRESS           parblk

eiosb:: .BLKQ    1                           ; Status
buflen: .LONG    0

;Set ETHERNET parameters
;From FORTRAN:
;   BYTE address(6) ! Filled in with destination address
;   INTEGER syschan ! Filled in with system channel from SYS$ASSIGN
;   INTEGER status  ! to receive routine status
;
;   status = ESET(syschan,address)
            .PSECT   CODE,PIC,EXE
            .ENTRY   ESET,^M<>
            MOVL     #512,buflen
            MOVL     8(AP),R3           ;Copy ADDRESS
            MOVB     (R3)+,unitad+0
            MOVB     (R3)+,unitad+1
            MOVB     (R3)+,unitad+2
            MOVB     (R3)+,unitad+3
            MOVB     (R3)+,unitad+4
            MOVB     (R3)+,unitad+5
;
; Set the mode function
;
            $QIOW_S            FUNC =#<IO$_SETMODE!IO$M_CTRL!IO$M_STARTUP>,-
                               CHAN = @4(AP),-
                               IOSB = eiosb,-
                               P2   = #pbdesc
            RET
            .END
```

NATIVE is the main FORTRAN program running on the POS node. This program makes calls to the MACRO program SUBLNK, which contains the entry points to set the characteristics of the ETHERNET device, and send and receive messages. The POS QIO requests called in the MACRO routines are collectively called DLX. In all, five DLX functions must be used:

o   IO.XOP - Open the line

o   IO.XSC - Set the line characteristics

o   IO.XRC - Receive a message

o   IO.XMT - Transmit a message

o   IO.XCL - Close the line

The IO.XOP request opens the line and assigns a network logical unit number to the ETHERNET device. The IO.XSC sets the device characteristics for the ETHERNET device. A receive request (IO.XRC) must be issued prior to issuing any transmit request. This is necessary because the PRO might not be able to post a transmit request and issue a receive request before the other system has replied to the transmitted message. Because of this, the receive request is issued with an event flag. When the event flag is triggered, the receive is completed. The transmit QIO request is designated IO.XMT, and can be issued once a read is posted. Upon completion of the program, the connection should be closed with the QIO call IO.XCL.

The calltree for the POS program is:


```
                  NATIVE
                 (FORTRAN)
               [main program]
                     |
                  SUBLNK
                  (MACRO)
                [subroutine]
            +--------+--------+
         SETLNK   RDLNK    WRTLNK
              {entry points}
```
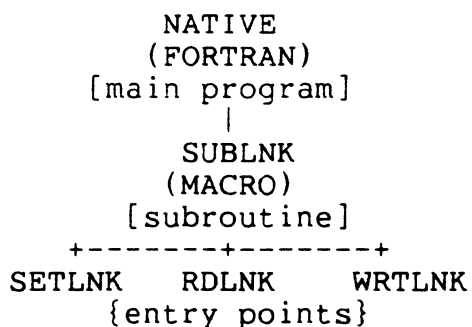

Calltree for the POS program NATIVE

The FORTRAN routine NATIVE is listed next, followed immediately by the MACRO routine SUBLNK.

```
            PROGRAM NATIVE
            IMPLICIT integer (A-Z)

C
C
            INTEGER setlnk,recmsg,sndmsg ! external macro routines
C
C       ASCII values
C
            CHARACTER*80    cmd              ! command sent
            CHARACTER*80    reply            ! reply received
            INTEGER isb(2)                   ! I/O status block
C
C       Define the constants
C

            succes = 0
            unit = 0
            netlun = 1
            ieflg = 2        ! event flag for completion of QIO read
            replen = 80      ! reply length can be changed

C
C       assign netlun NX0: (ETHERNET DECNA controller)
C
C
            CALL asnlun(netlun,'NX',unit,idsw)
            if (idsw .lt. 0) stop 'error assign lun'
C
C
C Set the ETHERNET driver characteristics and open line
C
            CALL setlnk(netlun)
            IF (status .LT. succes) GOTO 200


C Post read first
5           status = recmsg(reply,replen,netlun,isb,ieflg)
C Read the message to be sent
            WRITE(5,9)
            READ(5,10)(cmd(i:i),i=1,80)
            IF (cmd(1:1) .EQ. 'E' .OR. cmd(1:1) .EQ.'E' ) GOTO 100
9           FORMAT(/,' Transmit: ',$)
10          FORMAT  (80a1)

C Transmit message

C Determine length of command typed
            cmdlen = 80
15          IF (cmd(cmdlen:cmdlen).NE.' ') GOTO 20
            cmdlen = cmdlen - 1
            IF (cmdlen.gt.0) GOTO 15

20          status = sndmsg(cmd,cmdlen,netlun)
```

```
              IF (status .LT. succes) GOTO 200

              call waitfr(ieflg)                ! wait for event
              IF (status .LT. succes) GOTO 200

              WRITE(5,21) (reply(i:i),i=1,isb(2)) ! display message
              GOTO 5

100           CONTINUE
21            FORMAT (' ',' Receive:    ',<isb(2)>A1)
              CALL exit

200           WRITE(5,22) status
22            FORMAT(x,' QIO status return is: ',1I)
              CALL EXIT

              END



              .SBTTL MACROs and Data Definitions
;
;    System directives
;
              .MCALL DLXDF$
              .MCALL EPMDF$,EXIT$S
              .MCALL QIO$S,QIOW$S
              .MCALL WTSE$S
              DLXDF$
              EPMDF$
; -----------------------------------------------
;
;    Constants
;
              cc.app = 3140    ; Protocol (60-06)
;
;         Data storage
;
              .PSECT data
nwsb:   .BLKW    2            ; Network status block
device:.ASCII /CNA-0/  ; Device name
        devl = .-device ; Device name length
        .EVEN
netlun: .word
;
; -----------------------------------------------
;
;              device characteristics buffer
;
chrbuf:
        .WORD    CC.DST  ; Characteristics type
        .WORD    10.     ; Number of bytes of characteristics data
        .WORD    0       ; Output data byte count
        .WORD    0       ; Characteristics status
        .WORD    CC.APP  ; Protocol type
```

```
                .WORD    LF$PAD   ; LF$PAD  padding required protocol flag
;               .WORD    0        ; to turn LF$PAD off
;
;       -------------------------------------------------
;
;
;               protocol-destination address pair in HEX
;  This sets up the protocol-destination pair as in a table.
;  destination address
;               .BYTE    10       ; 08
;               .BYTE    0        ; 00
;               .BYTE    53       ; 2B
;               .BYTE    0        ; 00
;               .BYTE    7        ; 07
;               .BYTE    47       ; 27
;               chrl = .-chrbuf
;  source address
                .BYTE    252
                .BYTE    0
                .BYTE    04
                .BYTE    0
                .BYTE    312
                .BYTE    34
                chrl = .-chrbuf
;
;       -------------------------------------------------
;
                .SBTTL   Main Routine
                .PSECT   setlnk
setlnk::
;
;  open line
;
                MOV      @2(R5),netlun
;
                QIOW$S   #IO.XOP,netlun,,,#nwsb,,<#device,#devl,#0>
                TSTB     nwsb     ; Test the return status
                BMI      $err     ; Branch on error to fail set status, else
;                                        check status
;
;
setchr:
;
;  set characteristics
;
                QIOW$S   #IO.XSC,netlun,,,#nwsb,,<#chrbuf,#chrl>
                TSTB     nwsb     ; Test the return status
                BMI      $err     ; Branch on error to fail set status, else
;                                        check status
                CMP      #CS.SUC,chrbuf+6; Check characteristics status word
                BEQ      setok    ; If OK avoid the error handling
$err:
                MOV      nwsb,R0          ; return status of QIO first word
                JMP      fini
setok:
```

```
            MOV       chrbuf+6,R0       ; return characteristic status word
fini:
            RETURN
            .END


;
;   RDLNK
;
            .SBTTL MACROs and Data Definitions
;
;   System directives
;
            .MCALL DLXDF$
            .MCALL EPMDF$,EXIT$S
            .MCALL QIO$S,QIOW$S
            .MCALL WTSE$S
            DLXDF$
            EPMDF$
;
;   Constants
;
            cc.app = 3140    ; Protocol (60-06)
;
; ----------------------------------------
            .PSECT data
nwsb:   .blkw    2
recbuf:.word ; receive buffer
recbl: .word ; Length of buffer
netlun:.word ; netlun
ieflg:   .word
;
; ----------------------------------------------------
;
;        Receive characteristics buffer
;
recchr:
            .WORD     CC.DAD   ; Read destination address parm
            .WORD     6        ; Input data byte count
            .WORD     0        ; Output data byte count
            .WORD     0        ; Characteristics status
            .WORD     0        ; ETHERNET address
            .WORD     0
            .WORD     0
            .WORD     CC.ADR   ;  source address parameter
            .WORD     6        ; Input data byte count
            .WORD     0        ; Output data byte count
            .WORD     0        ; Characteristics status
            .WORD     0        ; ETHERNET address
            .WORD     0
            .WORD     0
            .WORD     CC.PRO   ; Protocol type parameter
            .WORD     2        ; Input data byte count
            .WORD     0        ; Output data byte count
            .WORD     0        ; Characteristics status
            .WORD     0        ; Protocol type
```

```
                 recl =    .-recchr
      ;
      ; ------------------------------------------------
                 .PSECT recmsg
      recmsg::
                 MOV       2(R5),recbuf        ; address of buffer to receive message
                 MOV       @4(R5),recbl        ; value buffer length
                 MOV       @6(R5),netlun       ; value of NETLUN
                 MOV       8.(R5),nwsb         ; I/O status block 2 word
      ;                    first is a status,second is a length of the message
                 MOV       @10.(R5),ieflg      ; event flag to be wait on in FORTRAN
      ;                                        program
      QIO$s     #IO.XRC,netlun,ieflg,,nwsb,,<recbuf,recbl,#recchr,#recl>
      done:     MOV $dsw,R0                    ; return status directive word
                 RETURN
                 .END


      ;
      ;          Wrtlnk MACRO subroutine
      ;
      ;          .SBTTL MACROs and Data Definitions
      ;
      ;   System directives
      ;
                 .MCALL DLXDF$
                 .MCALL EPMDF$,EXIT$S
                 .MCALL QIO$S,QIOW$S
                 .MCALL WTSE$S
                 DLXDF$
                 EPMDF$
      ;
      ;   Constants
      ;
                 cc.app = 3140     ; Protocol (60-06)
      ;
      ; ------------------------------------------------
      ;          Data storage
      ;
                 .PSECT data
      nwsb:     .BLKW     2         ; Network status block
                 .EVEN
      ;
      ;   Buffer space
      ;
      xmtbuf:  .word      ; buffer address
      xmtbl:   .word      ; buffer length
      netlun:  .word      ; net log unit number
      ;
      ; ------------------------------------------------
      ;
      ;          Transmit characteristics buffer
      ;
      senchr:
                 .WORD     CC.ADR  ; Set ETHERNET address for transmit
                 .WORD     6       ; Length of address
```

```
        .WORD    0          ; Size of data output
        .WORD    0          ; Characteristics status
;
; ----------------------------------------------
;
;                destination address in HEX and Octal
;
ethdst:
        .BYTE    252
        .BYTE    0
        .BYTE    04
        .BYTE    0
        .BYTE    312
        .BYTE    34
;
;                protocol type
;
protyp:
        .WORD    CC.PRO     ; Set protocol type
        .WORD    2          ; Length of protocol data
        .WORD    0          ; Size of data output
        .WORD    0          ; Characteristics status
        .WORD    CC.APP     ; Protocol type
        senl =   .-senchr; Length of the transmit char buffer
;
; ----------------------------------------------------
;
        .PSECT sndmsg
sndmsg::
        MOV      2(R5),xmtbuf    ; address of xmtbuf
        MOV      @4(R5),xmtbl    ; value of xmtbl
        MOV      @6(R5),netlun   ; value of netlun
;
        QIOW$S   #IO.XTM,netlun,#1,,#nwsb,,<xmtbuf,xmtbl,#senchr,#senl>
        CMP      #CS.SUC,$dsw    ; Compare the status word and success
        BNE      sfail           ; If unsuccessful go to error handle
        MOV      nwsb,R0         ; return I/O status of completion QIO
        jmp      done            ; done
sfail:
        MOV      $dsw,R0         ; return status directive word
done:
        RETURN
        .END
```

POS requires two additional files to task-build an application on the PRO. The first is the ODL file which contains the linkage relationships between the components of the program.

NATIVE.ODL

```
        .ROOT native-sublnk-OTSROT-RMSROT,OTSALL
@LB:[1,5]PROF77
@LB:[1,5]RMSRLX
        .END
```

The second file is the command file NATIVE.CMD, which tells the OS which LUNs to assign to which devices. The priority switch /PR:0 must be included. Failure to do this causes the OS to return the error condition indicating insufficient privilege.

NATIVE.CMD

```
native/fp/pr:0,native/-sp=native/mp
ASG=TI:5
ASG=NK0:1
ASG=DW1:2
GBLDEF=TT$LUN:5
GBLDEF=MS$LUN:6
GBLDEF=WC$LUN:0 ; oldfile - newfile services
GBLDEF=HL$LUN:0 ; help
GBLDEF=MN$LUN:0 ; menu
GBLDEF=TT$EFN:1 ; system event flag terminal I/O
CLSTR=PROF77,POSRES,RMSRES:RO
//
```

To build this application on the PRO, the main routine NATIVE is first compiled using the command:

```
FOR/LIST native
```

Then, the MACRO routine SUBLNK is assembled using the command:

```
MACRO/LIS/OBJ [1,5]netlib/LIB,[directory]sublnk
```

Finally, the objects are linked using the Program Application Builder:

```
RUN $PAB
PAB>@NATIVE.CMD
```

Using the Communications Routines

Both the VMS and POS programs assume that the other node's program is running before transmitting a message. The VMS program, NATIVEMODE, also requires that the POS node it is to communicate with is identified. Before running NATIVEMODE, the DCL command ASSIGN must be issued to assign the logical NET$DEV to the address of the POS node with which NATIVEMODE is to connect. From DCL, issue the command:

$ASSIGN XEnn:aaaaaaaaaaaa NET$DEV

where nn is the ETHERNET controller on the host system, and aa... is the hexadecimal ETHERNET address of the POS node.

For example, to connect to the POS node at address 0100A2010020 on the ETHERNET to which the controller XEA0: is connected, use:

$ASSIGN XEA0:0100A2010020 NET$DEV

The use of logicals allows the NATIVEMODE program to connect to different POS nodes simply by reassigning the logical designation NET$DEV.

Conclusions

Native mode ETHERNET provides a simple means of communicating across the ETHERNET without using the DECnet protocol. Although the development of a protocol requires extensive work, there are applications which might benefit from such an effort. The use of QIO requests on the VMS system and DLX requests on the POS system provides a straightforward means of accomplishing native mode ETHERNET communications. The demonstration programs shown should provide the reader with a sufficient template to use DLX/QIO requests for ETHERNET communications in both the POS and VMS systems.

Acknowledgments

The authors would like to express their appreciation to Bob Lordi who wrote the original version of the VAX program and Jenny Simich and Susan Brown who expended a great amount of effort in turning our engineering notes into this article.

Pay Now or Pay Later: Network Design and Analysis

Bill Hancock, CSP
2510 Limestone Ln.
Garland, Texas 75040
(214) 495-7353

"Now, we don't need to waste our time analyzing the network configuration. All we needed to do was buy the hardware we needed and a DECnet license. DECnet is installed everywhere, so I'm sure that we made the right choice. After all, it works for everyone else, does'nt it?"

Right.

"We thought about doing a thorough design. We even looked into using some consultants to help us, but the corporate types thought that it was wasted time and the consulants cost too much. Frankly, I'm a little concerned; I've never been involved with a network before."

Nice try.

"I think that we'll do it ourselves. Network design can't be that much more different than applications programming. Besides, I've been in computer science for a long time and I can learn anything out of a book, so I'll just get some book, read up on it, and design the network."

Medic! This one's gonna die!

What you have just read are actual statements that I have had the personal pleasure of hearing for the excuse of NOT going through the proper motions to design a network. Yuck! For those of you who read this and had an experience of deja vu, read on. This article is about network design and analysis - what it is, why you need it, and what happens if you don't do it.

Network design and analysis is a term we networking types apply to the basic methods necessary to PROPERLY design a network. A properly generated network design can provide a company with the following benefits:

- o Proper analysis of existing equipment for network installation
- o List of requirements for network installation
- o Proper configuration of network components for optimum cost savings
- o A network topology that is flexible and adaptable
- o Correct selection of network hardware for the network function
- o Correct selection of network software for the network function
- o Documentation of the network for future enhancements and modifications
- o Migration path into future network technologies without re-design
- o A long network life-cycle (reducing the costs of potential replacement)
- o Interconnect paths and methods for multiple network architectures
- o User analysis and configuration of network resources for optimal use

o Network management plan and methodology to reduce downtime and allow
   for maximum use of available resources
o Expectations for performance, reliability, and usability
o Optimal programming environment for network(ed) applications
o Training needs for programmers, users, and network managers
o Recurring expense forecasting and budgeting methods
o Network support needs (programming, management, user support)
o Use of mathematical modeling tools to help insure the sucess of the
   network design and topology
o Optimal design to prevent network congestion, queueing delay, and
   proper placement of routing and management resources on the network

So, for all you would-be network designers, take a hard look at the above list and tell yourself that your network design encompasses all the above issues and needs. And for those of you about to embark on the network trail, ask yourself if you have adequately answered this list of items. If you have, super! If you have not, you have'nt properly designed your network.

Network design is much more than ordering the parts and pieces from a vendor. It is much more than the suggestions the vendor gives you for configuration of your network. As a potential user of network components, you have the final decision on any network configuration and no matter what the vendor tells you to buy, the final decision to buy rests upon your shoulders. What that means, folks, is that if the network doesn't work as promoted to management, you can blame the vendor, but the ultimate person responsible is you, the person who recommended to management the network components to buy. And, if you think for one second that the smiling vendor across the desk from you is going to recommend some other vendor than himself for your network, you are either naive or you have been listening to too much vendor hype ("We're not here to make money – we're here to be your friend!"). So, remember, the person that will catch the blame from your company in the end is not the vendor – it's you! So, if you want to trust your vendor, great. Personally, there are few people I would consider qualified network designers and you can bet that you are not going to get access to them from a vendor for free. Remember, you get what you pay for.

The first step of network design is identification of the need for a network. While this may seem obvious, few companies sit down and spend some time logically defining the reasons for installing a network. Going through this exercise tells you whether or not a network is necessary to accomplish the desired function or whether there is a more cost-effective method to solve the problem at hand. I was asked to design a network for a large financial company one time and after looking over the needs very carefully, I told them that they didn't need a network. At first, the management of the company thought that I was nuts because their vendor had been hammering on them for months that they needed a network. They simply took it for gospel that they needed one and eventually the vendor got to the upper management and convinced them that they needed a network to solve their "problems." I was called in because the customer didn't know anything about networking and did not have the ultimate confidence in the vendor's efforts to find the "right" solution, regardless of what the vendor solutions were. After working on the project for three weeks, I found that the methodology adopted by the company's management for distributing workload and the reporting hierarchy involved was functioning very well and there was less than 5% out-flow of work to other company entities. What this meant was that 95% of the work being done in the respective branches stayed within the branch and did not require corporate intervention to get work accomplished. Also, all work was done in a reasonable manner and placing a computer in the middle of the paperwork effort would do nothing but slow things

down (yes, Virgina, computers are not always better). I went back to the customer's management and explained all of this to them and they immediately called the vendor and demanded an explanation. The vendor told them I was wrong and proceeded to do the one thing that a vendor should never do - cut down the competition. Since I was their "competition," it was obvious to them that I was trying to deprive them of a sale and they felt that they were right and I was wrong. Now I was mad! I spent another two weeks (at the customer's request) thoroughly documenting the lack of need for the network and also fighting some of the irrational claims of the vendor (I asked the vendor once why it was that they were pushing the network so hard when the customer didn't need it; their answer was "Because."). At the end, the vendor backed off their claims as the vendor had not done a thorough (or even partial) job of looking at the customer's needs and how the customer conducted business. The vendor had no idea as to what the customer's plans were for the next fiscal year nor did the vendor bother to look into the budgetary constraints that the customer was under. All the vendor cared about was making the sale - no matter how much the customer did not need it or how much it cost.

The entire hassle could have been avoided if the customer had thought, carefully, about why they "needed" a network. Instead, the customer was heavily influenced by the vendor's sales tactics and got swept up in the buying frenzy that usually accompanies a great many sales of networks. Also, the customer should have looked first to the company business plan - it will tell you whether a network is necessary to achieve the business goals of the company or not, based upon expected market penetration, growth factors, profitability requirements, and personnel requirements. So, rule number one is make sure that you need a network - don't go out and buy one due to unjustified internal pressure, vendor pressure, or peer pressure (yes, we all wish that we had a network just like company X down the street).

After a need for a network has been established, rule number two in network design is applied: what is it supposed to do and how much is it going to cost?

What it is supposed to do it a matter of defining, very carefully, what functionality the network is to offer. If it is electronic mail, file transfer, or task-to-task communications, great, but WRITE IT DOWN! Also, keep the base functionality of the network clear, consise, and simple. Too many good intentions get shot down because the base rationale was too complex for technical personnel to understand, much less the management personnel who have to approve and budget for it. Remember that your company's management is the signing authority for technical purchases and direction, regardless of what you have been told. If they can't understand what the needs are, you can bet that they will be more than a little apprehensive about installing technology they do not understand. I once told a company that a network does nothing and then explained that if it did anything at all, they should be glad. Setting expectations is very important and this is accomplished by carefully defining the network functionality.

So far, identification of need and identification of functionality have been defined. Now comes the problem of cost. Networks are just like systems in many ways: they have a life cycle, they require periodic upgrading and expansion, there are recurring costs such as software and hardware maintenance, telco service, packet services, modems, etc., they require personnel to management and maintain the network components, software may need to be developed so there may be costs for software engineering or applications programming, etc... The point is this: if you think that because network components are less expensive than a given system, think again. The overall cost of services and expansion will show that over a period of time, the network may turn out to be the most costly portion of your overall computing plan. Why? Simple. Networks, for all the high-tech bruhaha they have generated, are very expensive to install and operate

over a period of time because they are "service-intensive." What this refers to is the fact that networks require the use of vendor services more than a typical computer system might due to their inherent complexity and lack of a wide understanding of network technology by users, programmers, and managers. Networks are used for communications and communications services are expensive. Yes, networks can save a company a lot of money IF THEY ARE USED CONSISTENTLY AND PROPERLY. The sad thing is that without proper design, neither consistency or proper use of a network is achieved by most network users. To illustrate how expenses can creep up on you, here are some things that affect the cost of networking:

o Cost of hardware components (modems, cabling, channel interfaces, controllers, cabinetry, protocol converters, line conditioners, protocol analyzers, time domain reflectometers, frequency spectrum analyzers, breakout boxes, bit error rate testers, multiplexers, packet assembler/disassembler boxes, traffic analyzers response time analyzers, phone set tester, line testers, manual and automatic switching units, autodialers, protocol simulators, converters, data encryption equipment, auto callback units, data compression units, junction panels, line drivers, protocol converters, repeaters, bridges, voice frequency testers, front-end processors, servers, and many others)

o Cost of software components (networking architecture packages (such as DECnet, SNA, TCP/IP, and others), protocol emulators, protocol conversion, data compression, data analysis, network management, network troubleshooting, network statistics, network security, network applications (such as electronic mail, distributed data base applications, office systems, etc...), operating system interfacing software, etc...)

o Cost of operational services (leased-line cost, building conduit space costs, packet-switch network hookups, packet-switched network kilopacket charges, equipment leases, cable installation and add-ons, earth station channel charges, transponder channel charges, dial-up charges (digital service), dial-up charges (analog service), service surcharges for exceeding pre-agreed usage levels of shared services, general equipment maintenance, software maintenance, pickup/delivery and destination charges, line conditioning, per-call maintenance, per-call consulting services, administrative charges, etc...)

o Cost of consulting (network design, data collection, data reduction and analysis, network topology, traffic matrix, routing matrix, performance models, applications design, applications programming, queueing delay analysis, network technology assessment, network implementation, network installation, network management, network user training, network programmer training, network manager training, network project management, network troubleshooting and fault finding, network enhancements and add-ons, network interconnect design and implementation, interconnect training, network planning, network facilities survey, and many more)

o Cost of replacement due to improper initial design (all of the above plus the original cost to implement the current network)

While this may look like an extensive list, it isn't. That means that there are plenty of other costs that can come out of nowhere that were not expected or not properly planned for. You may look at this list and say to yourself that you don't need all the stuff listed above. This may be true, but I feel that with the influx of network technology and the price of the hardware dropping, you will find yourself involved in networking in the near future if you are not already involved. This also means that although you may not use some of the equipment and components listed above, what's to say that you will not later on in your current company or in some other computing life? By the way, for all you MicroVAX II buyers out there, do you honestly think that you can do without an Ethernet between them for too long? Digital doesn't think so. For all you companies buying MV II's and not thinking about networking them, dream on. Many companies that I deal with are just now starting to see the problem of purchasing MV II's. It's not the cost of the hardware or the functionality (which are superb, by the way) - its the distributed management, support, and logistics costs that kill you.

Now that we have identified a need for a network, we know what it is supposed to do, and we know that there are a lot of things that can affect the cost, the next thing to do is rule number three of network design: the site survey.

The site survey is not a trivial thing. Site surveys involve the careful examination of company facilities, building architecture, phone facilities (if you are using phone lines), existing computer hardware and software components, examination of existing contracts (to see if some already cover the needs for the network), power facilities, HVAC facilities, wireways and wire centers, electromagnetic interference possibilities, radio frequency interference possibilities, safety issues, security issues, building wiring and fire codes, electrical codes, reception and shipping facilities, building maintenance capabilities, on-site or vendor maintenance capabilities, and other related items. While this may initially seem to be not necessary, consider what happened to a friend of mine when he was designing the cable layout for a large electric company. He carefully measured the cable length needs and used a building diagram given to him by the customer as the basis for layout of the wire plan. What he did not know was that he was using an old plan and that many of the walls and wireways had changed. As a result, he planned wire runs directly through the women's restroom, which was not on the building diagram. Fortunately, since a proper site survey involves the customer, software, hardware, sales, service, and other selected personnel, the mis-layout of the wire plan was caught before the plan was finalized and corrections were made. Site surveys involve many people and require quite a bit of time to properly lay out the network in the environment in which it will function and to insure that all the "players" are where they are supposed to be when they are supposed to be there to insure a smooth installation of the network.

Rule number four is the basic network design, data collection/reduction, and data analysis. Network design, as I said before, is not as simple as throwing the wire down, hooking it up, and tossing some software on the machine. Network design is a science that has grown quite complicated as more and more sophisticated networks have evolved. A network designer starts out by actively and aggressively investigating all the needs, wants, hopes, and aspirations for the network that a company wishes to implement. He then takes the justifications that a company has written up, the functionality statement, and the site survey and identifies missing parts and pieces necessary to the network design. Following collection of data to satisfy the parts and pieces that are missing, the designer sets out to investigate the proper type of technology that the company requires now and to achieve their future goals for the network. Isolation of the proper technology is a critical step in solid network design. By providing alternative technologies, the network designer can give the customer a few good options by which to

implement the network, which can result in time and cost savings as well as reducing the risk of a single network technology causing network failures due to flaws, bugs, or other problems. Once a series of technologies have been defined, the designer then uses mathematical modeling tools (manual and computer-based) to figure out data flow ratios, probabilities of error, queueing delays, interconnect problems, least-cost network topological layout, routing paths, redundancy paths, and many more necessary items essential to solid network design. The modeled data is collected and reduced to meaningful facts and figures about the design and compared to network requirements dictated by the customer. If the results fit the requirements window, the network design being analyzed may be useful in the customer's environment (provided it meets physical needs, support needs, etc.). This process is repeated for every reasonable network technology until all the potential technologies are completely modeled. Following the modeling of network data, a financial analysis is done to determine how much the network is going to cost to implement, start-up, maintain, and expand. This is another exhaustive analysis that requires thought on the future of the network as well as applying practical experience with the theoretical network model. Finally, an assessment analysis is performed to identify networks that are "most" useful (closest to the desired functionality) and "least" useful (on the right track, but not closest to the desired combination of price/performance/ease of use, etc...). Once all of these items have been done, the network designer takes the results back to the site survey team and works with them to iron out any particular problems with the network designs as well as help isolate which design best suits the needs of the customer.

Another document the network designer will typically generate is one defining personnel needs and operational considerations. This document typically describes the type of personnel necessary to get the job done and what kind of personnel will be necessary for the day-to-day support of the network and its related components. In addition to the base personnel needs, a breakdown of costing for such personnel might also be included.

Once a particular network design has been identified by the network designer and the site survey team, a formal design document is drafted which documents the rationale for the design, a description of the components, a network topology, a wiring diagram, expansion capabilities, expected life cycle, applications support environment (and package descriptions, if applicable), network management environment, potential problems, data throughput analysis, testing and verification procedures, identification of network installation resources, an implementation timetable, personnel and training needs, cost analysis, and risks. The formal design document is the backbone to the network design and serves as a guideline for implementation and expansion. Following generation of the design document, a presentation is also created for the customer's management so that all parties involved thoroughly understand what the network looks like, what it is capable of doing, what resources are required, how long it will take to implement it and how much it will cost to implement, support, and maintain.

By now you have probably realized that there is not a network in place, yet, and still there have been quite a few people involved and an obvious amount of work has been done. Why go through all this grief just for a network?

The answer is simple and yet complex (the yin and yang of networks): proper business procedure and reduction of potential risks. I had a management friend of mine come up to me once and asked me why all of this was necessary for the sake of installing a wire, some controllers, and some software. I told him that it is like playing the stock market. There are people who buy a stock because it "looks" good; they may not have qualified the prospect, but they have a good feeling about it, so they buy the stock. This is the "gut feel" approach. Sometimes it works,

sometimes it doesn't, but studies have proven that it does not work more often than not (about 78% failures). Granted, there are some that seem to know how to use the gut feel approach very well and are very good at it, but these people are very few and far between. The second type try to play the stock market on their own. They read up on it for a while, read some analysis work on given stock types, and proceed to use a discount broker to invest their money in stocks that they select. This approach is usually not sucessful for a very long time due to the long learning curve necessary to play the stock game and the need to watch stocks over a pretty fair piece of time. The self-broker stock player is usually dismally profitable at first and may improve later on if he does not get frustrated and quit first. The third type of stock player is the high-risk options player. This type can be a gambler type and can make a killing or go bust in a matter of a day. Options players have to really understand the market to play well and profit. The fourth type is the stock player that uses a broker to invest his money in stocks in hopes that the broker can pick the right stocks and make the right decisions to generate a profit for the stock player. This is somewhat akin to using a consultant: there are very, very good stock brokers, but there are a lot of mediocre or poor stock brokers who are not overly cautious with their customer's investments and can ruin potentially reasonable deals. The final type of stock player is one who is a stock expert and can play the game himself with confidence due to his in-depth and expert understanding of the stock market. I then told my friend that most people are very leery about playing the stocks by themselves. I asked him how many options players he knew and how many expert stock players he knew. He answered that he didn't know any. Most people interested in the stock market try to find a good broker and the amount they pay the broker is worth it for the lower risk they are taking, the lack of need to become a "guru" in the stock market methodology, not to mention the reduction in time that it takes to monitor their investment.

When looking at network design and analysis, the main mistake that many companies make is that they approach a network in the same manner that they might approach the self-broker methodology. Nothing could be worse. Networks have some fairly serious restrictions on them that many systems do not as well as the fact that there are many more systems "experts" than there are network "experts" due to the complexity of network design as well as the lack of general network design education and information. Most systems have training and documents available for learning the hows and whys of systems hardware and software. Networks, unfortunately, are subject to the whims of multiple types of systems trying to talk to one another, frequently on differing technology, and takes on dimensions that most systems never have to worry about. Compound that with a severe lack of good, clear user documentation, technology information, and design documentation, network designers capable of designing superior networks are few and far between and usually rely on heavy experience and learning networking "the hard way."

If you have been scared to death about network design and analysis as a result of reading this treatise, good! It also means that you may now realize that the proper design of a network is critical to making the network cost effective over the long run as well as understanding that just because you have sharp systems people, they may not be able to design a network properly due to lack of information and experience. Using consultants in the network design phase can help drastically reduce the risk factor of the network and a good consultant can tell you what he can do and what you can do. This will save money in the short and long term as well as provide you a solid network design. While it is true that you can design your network yourself, it is not necessarily true that the network will survive over the long haul nor can you feel comfortable that it will perform as expected if you have not done a performance analysis before the network is in place. One of my partners calls network design the "Fram-Oil Principle," named after the Fram-Oil commercials on TV where a greasy mechanic is showing off a blown engine and says

"You can pay me now, or pay me later." Networks are much the same. Proper network design can save a ton of money down the road and is cost-effective up-front if done correctly. If you are penny-wise and dollar-foolish, you will, indeed, end up paying later.

The next time you look at designing a network for your company or your friend, go down the list of benefits at the begining of this article and ask yourself if you have received all of them. If not, you may have missed something somewhere. If you can claim that you have them all, you have benefitted from a good network design and can expect years of cost-effective service from your network.

OFFICE

AUTOMATION

OFFICE AUTOMATION SIG STEERING COMMITTEE

Chairman
Katherine 'Kit' Trimm
Pivotal, Inc.
Tucson, AZ

Vice Chairman
Ralph Bradshaw
Johnson and Johnson
Raritan, NJ

Communications Committee Representative
E. Catherine Ditamore
ARA Services
Philadelphia, PA

Symposium Coordinator
Mitch Brown
Gen Rad, Inc.
Waltham, MA

Special Projects
Gene LeClair
HQ Dept. of Army
Washington, DC

BOF Coordinator
Ray Kaplan
University of Arizona
Tucson, AZ

Newsletter Editor
Therese LeBlanc
Wheeling, IL

Library
Bob Hassinger
Liberty Mutual Research Center
Hopkington, MA

Tape Copy Coordinator
Randall Buck
Columbia Savings
Irvine, CA

ALL-IN-1 Working Group
Leon E. Ottley
Evans and Sutherland
Salt Lake City, UT

Symposia Assistant
Sal Gianni
Northeast Utilities
Hartford, CT

Store Coordinator
Mike Jackson
Air Force Operational
Test and Evaluation Center
Kirtland AFB, NM

Personal Computer SIG Liaison
Cheryl Johnson
Grinnell College
Grinnell, IA

Networks SIG Liaison
Gene LeClair
HQ Dept. of Army
Washington, DC

DECUS Europe OA SIG
Andreas Verbay
Telinco AG
Spiegelstrasse 20

Digital Counterparts
Les Agigian
Digital Equipment Corporation
Merrimack, NH

Geof Bock
Digital Equipment Corporation
Merrimack, NH

Session Notes
Martha Rudkin
GMF Robotics
Troy, MI

FROM THE EDITOR

Welcome to an exciting new year of sharing information through the OA SIG Newsletter. The OA SIG Steering Committee has been growing and the Fall Symposium in Anaheim, CA marked the several changes and additions to our ranks. The new list of committee members, and the positions they hold is on page OA-i.

One of the changes you will notice is that I have volunteered (or was I volunteered?) as the Newsletter Editor for the OA SIG, but I need your help! Our newsletter is the place for you to share technical information, software fixes, customization ideas and a host of other information regarding Office Automation with each other. So why not send me something to publish? Send your submissions to me at:

    Therese LeBlanc
    275 London
    Wheeling, IL 60090
    (312) 459-1784

Please follow these simple guidelines when submitting material:

- Make sure the information is clearly, darkly printed

- Check your spelling...please

- If you send multiple pages, number them

- Include your name, address and phone number so I can contact you if I have any questions

Wishing you all health, happiness and success in 1986.

Therese

NEXT OA SIG CHAIR NAMED

Katherine "Kit" Trimm was named Office Automation SIG Chair Elect at the OA SIG Managment meeting held December 8, 1985. Kit has been a key member of the OA SIG Steering Committee and Symposium Coordination team for the past two years. She has demonstrated her leadership and organizational skills in all phases of Symposia support, particularly in preparing, scheduling and running the OA sessions.

Founder and current OA SIG Chair Thomas Orlowski, will continue in his roll as Chair until the Spring Symposium in Dallas. He will then provide ongoing support to the SIG Steering Committee. The members of the OA SIG would like to thank Tom for his vision and persistance in founding the SIG, and for the excellent leadership he has provided them with for the past three years.

As Chair Elect Kit will be working closely with Tom for the next four months developing her knowledge of this position and preparing for the Spring Symposium.

We congratulate Kit and wish her continued success as a DECUS leader.

NOTES FROM THE FALL SYMPOSIUM

From the opening Roadmap session Monday morning to the Wrapup Friday afternoon, the attendance and participation in the OA SIG symposium sessions was outstanding.

We had five full days of sessions covering a variety of topics: OA concepts, new products, system management, message router, communications, work stations, user panels, VTX and much more. The user presentations were excellent and we had several highly technical sessions which were well received.

Strong involvement by our DEC counterparts helped make two of the week's highlights possible; the OA Question & Answer session, and the OA Wishlist (things users would like to see done or added to DEC products).

We're not certain, but we may have broken a record at the Q&A session on Wednesday evening. We had close to 20 DEC representatives in the same room for two hours! Eleven of them were the official 'answer' panel and another six or seven were in the audience to help out if needed. The best part of all this was that they were there to listen and respond to user questions.

The OA Wishlist session on Friday morning began with DEC representatives responding to the wishlist items presented to them last May in New Orleans (a reprint of those responses will appear in the February newsletter). A steady stream of new wishes gave DEC reps. another list to take home with them and respond to at the Spring Symposium.

There were also many informal Birds Of a Feather (BOF) meetings held by OA users with similar interests and concerns. The Executive Track day (on Thursday) for senior management helped to provide high level management from many different companies with the dynamic concepts of Office Automation and some real life case studies.

This is just a sampling of the sessions and activities sponsored by the Office Automation SIG in Anaheim. Our SIG is growing and evolving at a rapid pace and we look forward to more exciting programs at the next symposium. We invite each of you to attend and participate in the Spring Symposium, April 28 - May 2, in Dallas Texas.

OA-4

ALL-IN-1

NEWSLETTER
December 1985
DECUS Edition

digital™

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The following are trademarks of Digital Equipment Corporation.

digital™	DECsystem-20	RSTS
	DECtalk	RSX
ALL-IN-1	DECUS	UNIBUS
DATATRIEVE	DECwriter	VAX
DEC	DIBOL	VMS
DECmail	MASSBUS	VT
DECmate	PDP	Work Processor
DECnet	P/OS	WPS-PLUS
DECspell	Professional	
DECsystem-10	Rainbow	

CONTENTS

OA-5

The DECUS 1985 ALL-IN-1 Newsletter

This is the ALL-IN-1 Newsletter for DECUS 1985. It tells you about tuning ALL-IN-1 to improve performance and how to best use and manage your ALL-IN-1 Version 2 system.

Do You Need Support In Running ALL-IN-1 Version 2?

DIGITAL offers a variety of services that can make it easier to run ALL-IN-1 Version 2 systems.

1 Startup Services

Startup Service Packages are available to get ALL-IN-1 operational and accelerate your productive use of ALL-IN-1.

2 Software Product Service Agreements

* DECsupport Service is the most comprehensive level of service, offering critical problem on-site assistance and scheduled preventative maintenance. You receive telephone support that gives you timely answers and solves most software problems. In addition, you get revised versions of the software and documentation, and system newsletters or dispatches.

* BASIC Service is ideal for customers who have a staff whose experience and expertise enables them to analyze and communicate a software problem to DIGITAL remote support centres. You receive telephone support that gives you timely answers and solves most software problems. In addition, you get revised versions of the software and documentation, and system newsletters and dispatches.

* Self-Maintenance Service is designed for customers who require revised versions of the software and documentation from DIGITAL. In addition, you get system newsletters or dispatches and may submit software performance questions.

A variety of service options may be added to an existing Software Product Service Agreement, such as service for multiple-like systems.

3 Training From Educational Services

To ensure customer success with DIGITAL products, Educational Services sells training for the installation, maintenance and/or management of DIGITAL software. The training available for ALL-IN-1 users is:

* ALL-IN-1 Version 1 to Version 2 Migration Planning and Training

* ALL-IN-1 Version 2 User Update Training

* ALL-IN-1 Version 2 System Manager Training

If you need these or any other services, please call your DIGITAL Account Manager.

4 ALL-IN-1 Release Notes

What do you use your ALL-IN-1 Release Notes for?
Do you ...

1. Use them when you install ALL-IN-1 ? |‾|
2. Refer to them when you have a problem with ALL-IN-1? |‾|
3. Pass on relevant information in them to your users? |‾|

You ticked every box, of course!

You didn't?

Why not read through them? As well as giving you extra information for when you install ALL-IN-1, they contain extra information about the day-to-day running of ALL-IN-1 that is not found in any other documentation.

Sections 4 and 5 describe late changes to ALL-IN-1, particularly aspects of ALL-IN-1 that changed after the documentation was typeset, and details about the use and management of ALL-IN-1 Version 2 that you should be aware of.

In Section 4 you will find information on the day-to-day running of ALL-IN-1, for instance Electronic Messaging, printing, using the Scratch Pad and Time Management.

Section 5 gives information about the optional products that can be used with ALL-IN-1, for instance, DATATRIEVE and WPS-PLUS/ALL-IN-1.

Some of the information in sections 4 and 5 is for you, the system manager, but there is other information which your users will need to know.

For instance, in section 4 there is a description of how to subtract from the result of a calculation. The guide describes where to place the equals sign in the chain of calculations.
For example: Instead of entering 562 * 25 = 1450 - 50 = ...
                    enter 562 * 25 - 50 = ...

In Section 4 you will also find an addition to the ALL-IN-1 documentation concerning WPS and EDT which DECmate users should know about. It tells you that if you transfer a document from a DECmate using CX (character transfer) to ALL-IN-1 on VMS Version 4.0 or later, the transferred document may lose some characters. To avoid this, users should change their VMS terminal set-up by putting the command $ SET TERMINAL /HOSTSYNC into their LOGIN.COM 'le.

So, make good use of your Release Notes.

Pass useful information on to your users to save them from unnecessary problems.

And next time you have a problem with ALL-IN-1, have a look through your Release Notes. They might have the answer to the problem.

5 Sending Mail to VMSmail Recipients

Do all the systems to which you send mail, have Message Router and the Message Router VMSmail Gateway (MRGATE) installed?

If not, you may be having problems sending ALL-IN-1 mail to VMSmail recipients.

If Message Router and MRGATE are not installed on every system in the network, use the address format:

TO: username @nodename @MRGATE @gateway_node

where gateway_node is the name of the node that is running MRGATE, and can be omitted if it is the local node name.

The address format given in the ALL-IN-1 documentation is:

TO: username @MRGATE @nodename

This format is the correct way to address mail to VMSmail recipients only if every system in the network to which you send VMS mail has Message Router and MRGATE installed.

6 VAX/VMS 4.2 Mail Fetcher

If you are running VAX/VMS Version 4.2 on your system, the Electronic Messaging fetching routine will not finish if a mail message containing an attachment is received by the Fetcher from a remote system. The fetching routine has the process name of "Al Fetcher" and is queued as a batch job called OAMTIMAIL. This problem causes excessive use of CPU time.

Workaround: If you see that the batch job OAMTIMAIL is running continuously, stop the Fetcher using the DCL command STOP "process_name". You can then use the RF option in the Electronic Messaging System Management menu in ALL-IN-1 to interactively fetch the message. The batch job OAMTIMAIL will start again when another message is received.

This problem is fixed in ALL-IN-1 Version 2.1.

7 Converting DECmail Messages To ALL-IN-1

o Converting DECmail messages to ALL-IN-1 takes approximately 40 minutes for 100 memos. The more memos there are, the longer it takes to transfer each memo. Therefore it is advisable to ask users to delete any unwanted memos before transferring DECmail accounts to ALL-IN-1.

o Even if only one account from a save file is being transferred into ALL-IN-1, all the accounts in the save file are transferred, and the unwanted ones are later deleted by ALL-IN-1. For this reason, it is recommended that you do not put too many accounts into one save file.

o The conversion program runs as a batch job. Do not run the conversion program in a batch queue which has time limits set.

If a conversion for a user fails, you must complete the following steps before attempting the conversion again:

1 Log into the ALL-IN-1 system manager's VMS account (or the account where the conversion was initiated) and run ALL-IN-1. Delete the folder called Decmail Conversion, if it exists.

2 Log into the user's ALL-IN-1 account and run ALL-IN-1. Delete the folder called Odm Master, if it exists. Enter the following ALL-IN-1 command while in the user's account:

    <CAB GET_PENDING "DMCONV","WASTEBASKET"

In ALL-IN-1 Version 2.1 these tasks will be done automatically.

Converting DECmail messages to ALL-IN-1 is described in the ALL-IN-1 Installation Guide. The ALL-IN-1 Release Notes contain additional information that you should read when converting DECmail messages to ALL-IN-1.

8 Upgrading ALL-IN-1 Version 2.0 from VAX/VMS Version
3.n to VAX/VMS Version 4.n 4.n

If you are, or have been, running ALL-IN-1 Version 2.0 on VMS
Version 3.n, then ALL-IN-1's indexed data files will have a
Prolog of 1. If you upgrade to Version 4.n of VMS, you must
convert the indexed data files to Prolog 3 to allow ALL-IN-1 to
access the files correctly.

You can see the Prolog of a file using the DCL command
DIRECTORY/FULL.

When you have upgraded your VAX/VMS system to Version 4.n, carry
out the following procedure to convert the relevant ALL-IN-1
files to Prolog 3:

```
$ SET DEFAULT OA$DATA
$ CONVERT/PROLOG=3/FDL=OA$LIB:NETWORK.FDL NETWORK.DAT NETWORK.DAT
$ PURGE NETWORK.DAT
$ CONVERT/PROLOG=3/FDL=OA$LIB:ATTENDEE.FDL ATTENDEE.DAT ATTENDEE.DAT
$ PURGE ATTENDEE.DAT
$ CONVERT/PROLOG=3/FDL=OA$LIB:MEETING.FDL MEETING.DAT MEETING.DAT
$ PURGE MEETING.DAT
```

and for each ALL-IN-1 user:

```
$ SET DEFAULT <user's ALL-IN-1 directory>
$ CONVERT/PROLOG=3/FDL=OA$LIB:DOCDB.FDL DOCDB.DAT DOCDB.DAT
$ PURGE DOCDB.DAT
$ CONVERT/PROLOG=3/FDL=OA$LIB:ACTITEM.FDL ACTITEM.DAT ACTITEM.DAT
$ PURGE ACTITEM.DAT
```

9 Time Management

You should be aware of the following attributes of Time
Management which are not described in your Release Notes. These
are all corrected in ALL-IN-1 Version 2.1.

o When an apostrophe (') is typed in the meeting location or
   description field, the meeting is scheduled but when an
   attendee tries to read the message they will cause an access
   violation.

   Solution: Do not enter an apostrophe (') in these fields.

o Someone whose ALL-IN-1 user name contains a comma can delete
   a meeting that he or she schedules, but attendees cannot
   delete the meeting from their calendars.

   Solution: If possible, avoid creating user accounts with
   names containing commas.

o When the Advanced Calendar (AC) option on the Reminders (RE)
   menu is used, the file ACTITEM.DAT is left open if a date
   without any reminders scheduled is selected. Eventually the
   limit for the number of files that may be left open is
   reached and an error message is displayed.

   Solution: Exit from ALL-IN-1 when the open file limit is
   reached. ALL-IN-1 closes all open files when this is done.
   You can then enter ALL-IN-1 again.

o If you send a meeting notice to an attendee who has the same
   name as yourself but is on another node, you will never
   receive the attendee's reply.

   Solution: If possible, create user accounts with names that
   are as unique as possible, for example FTSMITH instead of
   SMITH.

o If you send a meeting notice to an attendee whose User
   Profile language is different from yours, the meeting will
   not be scheduled.

   Solution: Under development.

o When scheduling a meeting, ALL-IN-1 does not find all of the
   attendee's free time and sometimes states that attendees are
   free when they are not.

   Solution: Under development.

o Time Management does not find all conflicting meetings and appointments.

Solution: None currently available.

10   Performance Management

Your ALL-IN-1 system must be carefully managed to maintain a consistently high level of performance. Effective performance management of your ALL-IN-1 system involves:

o Reviewing the hardware and software configuration

o Reviewing and monitoring system performance and workload

o Tuning VMS

o Keeping the ALL-IN-1 File Cabinet and Electronic Messaging system well maintained

These four activities are essential for maintaining good performance and avoiding performance problems.

There is a complete discussion of ALL-IN-1 performance management in the ALL-IN-1 Performance Management Guide. A summary of the key issues is given below:

10.1   System Set-up

It is important that your system has adequate hardware resources to meet its workload. Make sure your system been configured according to DIGITAL recommendations for processor power and memory size particularly as your system requirements change, for instance, when the number of users increases.

Review the user working set parameters (WSDEF, WSQUOTA and WSEXTENT) in the system authorization file.

When ALL-IN-1 is installed on your system, you will almost certainly need to increase the size of your PAGE and SWAP files and you will probably need to allocate secondary PAGE and SWAP files to different disk devices. Refer to your ALL-IN-1 Performance Management Guide for further details.

10.2   Monitoring and Reviewing the System

Monitoring system behavior and usage is the most important aspect of performance management. A good understanding of the performance characteristics of your system gives you a starting point for system tuning and performance troubleshooting and provides you with the information needed to plan ahead and avoid problems.

It is recommended that you:

o Monitor system resource usage on a daily basis and keep summary records of day-to-day resource usage in these areas:

   - Average and peak processor load

   - Avearge and peak page fault rates

   - Average and peak disk I/O traffic for each disk device

   The VMS supplied MONITOR utility provides the facilities to do this.

o Periodically review memory usage characteristics, particularly:

   _ Paged and non-paged pool usage against availability

   _ Request packet (SRP, IRP and LRP) usage against availability

   _ PAGE and SWAP file space in use

   The DCL SHOW MEMORY command provides this information.

o Run process level accounting on a daily basis to keep track of which users or groups of users are using system resources. Summary statistics on a user-by-user basis should be kept for analyzing trends which can be used to assist with growth prediction.

   Refer to the VAX/VMS Accounting Utility Reference Manual for details of how to use the ACCOUNTING utility.

10.3   Tuning VMS

There are a small number of VMS SYSGEN parameters which should be checked and, if necessary, adjusted to get the best out of your ALL-IN-1 system. The most important SYSGEN parameters to review are:

NPAGEDYN and NPAGEVIR
PAGEDYN
LRPCOUNT and LRPCOUNTV
SRPCOUNT and SRPCOUNTV
IRPCOUNT and IRPCOUNTV
MPW_HILIMIT
MPW_LOLIMIT
MPW_THRESH
MPW_WAITLIM

Please refer to the following additional sources of information:

o ALL-IN-1 Performance Management Guide

o Guide to VAX/VMS Performance Management

o VAX/VMS System Generation Utility Reference Manual

You should not adjust any VMS system parameters unless you know
exactly what you are doing and why. It is easy to degrade the
performance of any VMS system by careless or uninformed
adjustment of system tuning parameters. Only make changes to the
system if you have an informed reason for doing so.

If you are unfamiliar with VAX/VMS performance characteristics
and the effects of the SYSGEN parameters listed above, contact
your DIGITAL Account Manager.

10.4 Housekeeping and Maintenance

10.4.1 File Cabinet Reorganization

From time to time, the ALL-IN-1 File Cabinet must be reorganized
using the File Cabinet Reorganization option. How frequently you
reorganize the File Cabinet varies from system to system
depending on the workload. It is recommended that you reorganize
at least once each week, but it may be necessary to do it more
frequently if you notice that performance is becoming degraded.

The most visible indication that the performance of the File
Cabinet has been degraded is the response time to perform File
Cabinet access operations such as:

o Selecting a document

o Deleting a document

o Creating and editing a document

o Sending mail messages

o Indexing the file cabinet

10.4.2 File Cabinet Index

OA$DATA:DAF.DAT is an indexed file that holds reference data on
every document in the File Cabinet. It can grow very rapidly and
thereby degrade system performance. It is important to make sure
that unwanted documents are deleted when they are no longer
required.

If a user keeps a large number of unwanted files in the File
Cabinet, it can degrade system performance. Encourage your users
to delete unwanted documents regularly. In particular,
Electronic Messaging can lead to large numbers of documents being
kept in the File Cabinet unnecessarily.

10.4.3 Housekeeping the Electronic Mail System

The ALL-IN-1 Electronic Messaging system maintains two log files
defined by the logical names OA$MTI_LOG and OA$MTI_ERR.

o OA$MTI_LOG keeps a record of each mail message sent on the
   system.

o OA$MTI_ERR keeps a record of any error messages caused by
   sending and receiving mail messages to and from remote
   addressees.

As these files increase in size, it takes longer to append new
records to them and the speed at which mail is sent is degraded.
It is recommended that you periodically close, archive and
replace these files to prevent them from becoming too big.

10.4.4 Removing Unused Facilities

If you customize ALL-IN-1, you may have forms and scripts that
you do not use. You can save memory and improve performance by
removing unwanted forms and scripts from the installed memory
shareable libraries, OAFORM.FLB and A1TXL.

Refer to Chapters 15 and 16 of the ALL-IN-1 Application
Programmer's Reference for details.

DECUS

PERSONAL COMPUTER

SIG NEWSLETTER
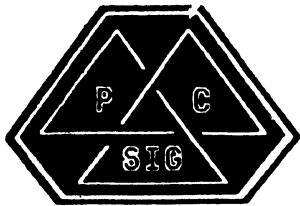
```
=====================================================================
PERSONAL COMPUTER SIG STEERING COMMITTEE
=====================================================================
```

Chairman

Barbara Maaskant
UT Health Science Center
7703 Floyd Curl Drive
San Antonio, TX 78284
(512) 691-7351

Symposia Coordinator

Rick Eliopoulis
5258 Vickie Drive
San Diego, CA 92109
(619) 225-7867

DECmate Working Group Chairman

Cheryl Johnson
Grinnell College
P. O. Box 805
Grinnell, IA 50112-0810
(515) 236-2570

Pro Working Group Chairman

Thomas R. Hintz
U. of Florida
IFAS Computer Network,
1022 McCarty Hall
Gainesville, FL 32611
(904) 392-5181

Rainbow Working Group Chairman

Lynn Jarrett
Union Tribune Pub. Co.
P. O. Box 191
San Diego, CA 92108
(619) 299-3131 (x1130)

Newsletter Editor

Caroline M. Mack
9007 Mears Street
Fairfax, VA 22031
(703) 280-4404

National LUG Organization Rep

Pierre Hahn (acting)
SUNY HSC-T10-02808101
Stony Brook, NY 11794
(516) 444-1362

Library Comm. Rep/PC Librarian

Kurt Reisler
Hadron, Inc.
9990 Lee Highway
Fairfax, VA 11030
(703) 359-6100

PC/RSX Liason

Peter Flack
Computer Sciences Corp.
P. O. Box 12233
Research Triangle Park, N
27709
(919) 541-4669

PC/Graphics Liason

Dr. Khin Maung Yin
Kent State University
1411 Clarindon-Troy Road
Burton, OH 44201
(216) 951-1447

Pre-Symposia Session Coordinato

Frederick G. Howard
Eastman Kodak Company
901 Elmgrove Road D345-L
Rochester, NY 14650
(716) 724-5331

Session Notes Editor

Alan Bruns
Allied Electronics
401 E. 8th Street
Fort Worth, TX 76102
(817) 336-5401

Members-at-Large

Rockey Salwinski
UT Health Science Center
7703 Floyd Curl Drive
San Antonio, TX 78284
(512) 691-7351

Jim Christine
SPSS, Inc.
1815 S. Cuyler
Berwyn, IL 60402
(312) 329-3580

Russ Wertenberg
Sandia National Labs
Div 8352
Livermore, CA 94550
(415) 422-2663

Michael Bowers
University of California
Animal Science Department
Davis, CA 95616
(916) 752-6136

Digital Counterparts

DECmate:

Ron Gemma
Digital Equipment Corp.

PRO:

John E. Bennett
Digital Equipment Corp.
ML01-2/C30
146 Main Street
Maynard, MA 01754-2571

Rainbow:

Katrina Holman
Digital Equipment Corp.
LJ02/I3
30 Porter Road
Littleton, MA 01460
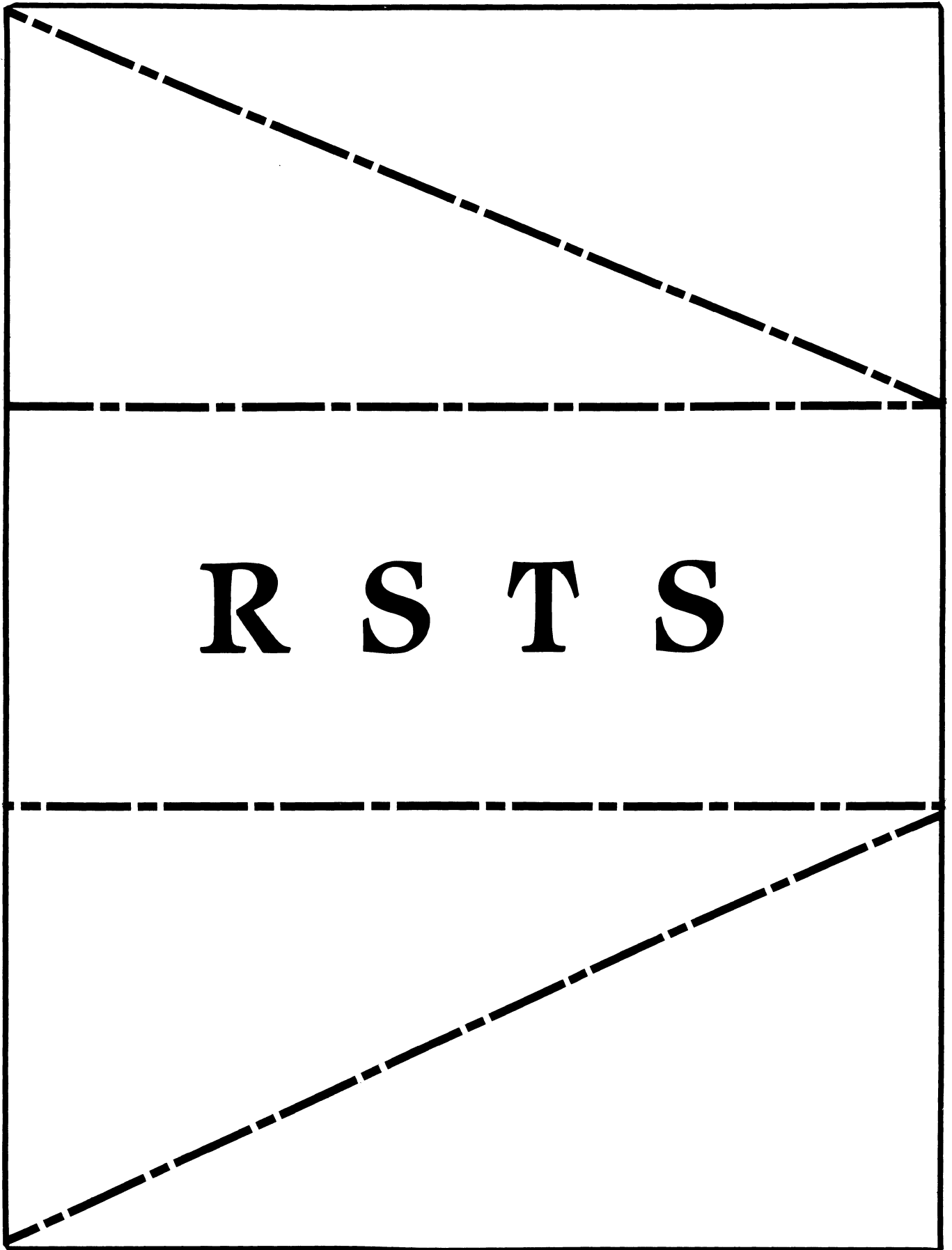
R S T S

RSTS

Chairman
Charles Mustain
Stark County School System
Louisville, OH

Symposium Coordinator
Scott W. Pandorf
Kittle's Home Furnishings
Indianapolis, IN

Assistant Symposium Coordinator
Wef Fleischman
Software Techniques
Cypress, CA

Newsletter Editor

Open

Library Representative
Susan Abercrombie
Ventrex Laboratories Inc.
Portland, ME

DEC Counterpart
Joel Arker
Digital Equipment Corporation
Merrimack, NH

Pre-Symposium Seminar Coordinator
Bruce Gaarder
Macalester College
St. Paul, MN

Wish Lists Coordinator
Neal E. Goldsmith
Software Techniques, Inc.
Cypress, CA

Vice SIG Chairman
Wish Lists & Tape Copy Coordinator
Philip Hunt
System Industries
Milpitas, CA

EDUSIG Liaison
George Wyncott
Purdue University Computing Center
W. Lafayette, IN

RSTS Product Planning Coordinator
Errol E. Ethier
Shrewsbury, MA

Members-At-Large
Ed Beadel
Instructional Computer Center
Oswego, NY

Scott Daily
Great Lakes Chemical Corp.
W. Lafayette, IN

Mark Gilmore
Cal State University
Long Beach, CA

Mark Hartman
Jadtec Computer Group
Orange, CA

Jeff Killeen
Information Design & Management
Hopedale, MA

Newton J. Munson
Rochester Institute of Technology
Rochester, NY

RSX

MULTI-TASKER

DECUS

Chairman
Dan Eisner
Perkin-Elmer Corp.
Garden Grove, CA

Vice-Chairperson
Elizabeth Bailey
Tennessee Valley Authority
Muscle Shoals, AL

Symposium Coordinator
Rick Sharpe
Toledo Edison
Toledo, OH

Pre-Symposium Seminar Coordinator
Hans Jung
Associated Press
New York, NY

Communications Committee Representative
Allen Bennett
Lear Siegler Rapistan
Grand Rapids, MI

Newsletter Editor
Dominic J. DiNollo
Loral Electronics
Yonkers, NY

Store Coordinator
Bob Freeborn
Savin Corporation
Binghamton, NY

Session Note Editor
Burt Janz
Northern Telecom Inc.
Concord, NH

Librarian
Glenn Everhart
Mt. Holly, NJ

Campground Coordinator
Jerry Ethington
Prolifix Inc.
Frankfort, KY

DEC Counterparts
Tim Martin
Digital Equipment Corporation
Maynard, MA

Dick Day
Digital Equipment Corporation
Nashua, NH

Bruce Webster
Digital Equipment Corporation
Nashua, NH

Working Group Coordinator
Ed Cetron
Center for Biomedical Design
Salt Lake City, UT

Working Group Chair
Evan Kudlajev
Philadelphia Electric Co.
Philadelphia, PA

RSX Group Chair Software Clinic Coord.
Roy S. Maull
U.S. Air Force
Offutt AFB, NE

Software Clinic Coordinator
Bruce Zielinski
RCA
Moorestown, NJ

Volunteer Coordinator
Gary Maxwell
U.S. Geological Survey
Menlo Park, CA

Multi-Processors Working Group Coordinator
Bruce Mitchell
Machine Intelligence & Indus. Magic
Hudson, WI

Networks Working Group Coordinator
Mark Podany
Case Western Reserve University
Cleveland, OH

SRD Working Group Coordinator
Bob Turkelson
Goddard Space Flight Center
Greenbelt, MD

Accounting & Performance Working Group Coord.
Denny Walthers
American McGaw
Irvine, CA

Menu Coordinator
Ed Cetron
Center for Biomedical Design
Salt Lake City, UT

Members-At-Large
Jim McGlinchey
Warrenton, PA

Jim Neeland
Hughes Research Labs.
Malibu, CA

Anthony E. Scandora, Jr.
Argonne National Laboratory
Argonne, IL

Ralph Stamerjohn
Creve Coeur, MO

Jim Hopp
Carleton Financial Comp.
South Bend, IN

SJ RTMON FILEX RMON CSI ODT

PIP LD SYSMAC FB

DIR KMON

QUEUE QUEMAN TECO

RT-11
SIG
DECUS
U.S. CHAPTER

PAT DUP

VM

KED K52

BINCOM DUMP BUP

LIBR SIPP

LINK

SRCCOM RESORC FORMAT

SLP

TTYSET IND

MACRO XM JSW HELP
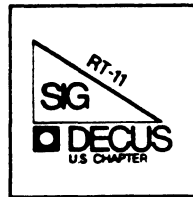
RT SIG Steering Committee

Chairman
John T. Rasted
JTR Associates
Meriden, CT

Symposium Coordinator
Ned Rhodes
Software Systems Group
Rockville, MD

Communications Committee Representative
FMS Contact
Susan Rasted
Software Dynamics, INc.
Wallingford, CT

Newsletter Editor
COBOL Contact
Bill Leroy
The Software House, Inc.
Atlanta, GA

Pre-Symposium Seminar Coordinator
Suite Manager
Bruce D. Sidlinger
Sidlinger Computer Corp.
San Antonio, TX

Standards Coordinator
Kenneth L. Aydlott
Teledyne Hastings-Raydist
Hampton, VA

Digital Counterpart
Diana L. Miller
Digital Equipment Corporation
Maynard, MA

Tape Copy Generation Contact
Ralston Barnard
Sandia Laboratories
Albuquerque, NM

APL Contact
Doug Bohrer
Bohrer & Company
Wilmette, IL

MACRO Contact
Nick Bourgeois
NAB Software Services, Inc.
Albuquerque, NM

TECO Contact
Product Planning Contact
John Crowell
Crowell Ltd.
Los Alamos, NM

DECNET Contact
Ken Demers
Adaptive Automation
New Haven, CT

UNIX Contact
Wish List Contact
Bradford Lubell
L.A. Heart Lab
Los Angeles, CA

RT-11 Hardware Contact
Carl Lowenstein
University of Calif., San Diego
San Diego, CA

TSX Contact
C Contact
Jack Peterson
Horizon Data Systems
Richmond, VA

Library Contact
Tape Copy Distribution
Tom Shinal
General Scientific Corp.
Rockville, MD

BASIC Contact
Ed Stevens
EMDA Inc.
Pasadena, CA

CAMAC Contact
J.W. Tippee
Kinetic Systems Inc.
Lockport, IL

Personal Computer Contact
LUG Contact
William K. Walker
Monsanto Research Corp.
Miamisburg, OH

FORTRAN Contact
RUNOFF Contact
Robert Walraven
Multiware, Inc.
Davis, CA

DECUS
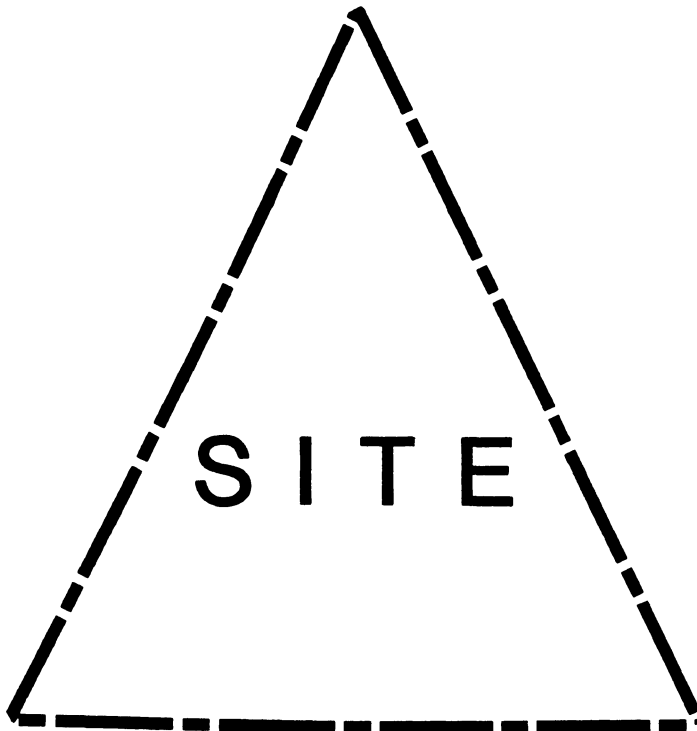
SITE

SITE

Chairman
David Hunt
Lawrence Livermore National Lab
Livermore, CA

Symposium Coordinator
Michael Weaver
OJCS/JAD/TSD
Washington, DC

Communications Committee Representative
Terry C. Shannon
Digital Review
160 State Street
6th Floor
Boston, MA 02109

Newsletter Editor
Gregory N. Brooks
Washington University Beh. Res. Lab
St Louis, MO

LUG Coordinator
Doug Bickford
UVM Academic Computing Center
Burlington, VT

Session Note Editor
Gary Bremer
Emerson Electric Co.
St. Louis, MO

HMS SIG Liaison
Jim Corrigan
TI Research
Richardson, TX

Assistant Librarian
Timothy Frazer
Specialized Bicycle Components
Morgan Hill, CA

Librarian
Larry W. Hicks
Relational Database Services
Kernersville, NC

Hardware Management
Emily Kitchen
A.H. Robins Co.
Richmond, VA

Staff Management
Adam Zavitski
Tulsa, OK

Members-At-Large
George Hilton
Compucard International
Stanford, CT

Debbie Boole
Texas Instruments
Dallas, TX

DEC Counterparts
Rogers Bent
Digital Equipment Corporation
Stow, MA

Lil Holloway
Digital Equipment Corporation
Bedford, MA

Susan Porada
Digital Equipment Corporation
Marlboro, MA

Gary Siftar
Digital Equipment Corporation
Tulsa, OK

UNISIG

DECUS

UNISIG

UISIG Chair
Kurt L. Reisler
Hadron Inc.
9990 Lee Highway
Fairfax, VA 22030
703-359-6100

UNISIG Symposia Coordinator
Stephen M. Lazarus
Ford Aerospace
Palo Alto, CA 94303
415-852-4203

UNISIG Newsletter Co-editor
James W. Livingston, Jr.
Measurex Corporation
1 Results Way
Cupertino CA 95014
408-255-1500 X4468

UNISIG Newsletter Co-editor
Bruce A. Bergman
2235 Meyers Avenue
Escondido, CA 92025-1070
619-741-8825

UNISIG Administrative Daemon
Dorothy Geiger
49 Showers Drive, #451
Mountain View, CA 94040
415-948-1003

UNISIG Tape Librarian
Carl Lowenstein
Marine Physical Laboratory
Scripps Institute
of Oceanography, P-004
LaJolla, CA 92093
619-294-3678

UNISIG Usenet Liason
Joe Kelsey
John Fluke Mfg. Co, Inc.
P.O. Box C9090, M/S 223B
Everett, WA 98206
206-356-5933

UNISIG Standards Coordinator
Jeff Gilliam
National Semiconductor
2900 Semiconductor Drive, MS C2303
Santa Clara, CA 95051
408-721-3801

UNISIG Minister Without Portfolio
Norman Wilson
Bell Laboratories, 2C-529
600 Mountain Avenue
Murray Hill, NJ 07974
201-582-2842

UNISIG DEC Counterpart
Roseann Maclean
DEC, MD02-1/H10
Continental Blvd.
Merrimack, NH 03054
603-884-5702

NEWSLETTER OF THE VAX SYSTEMS SIG

Pageswapper

Our Mascot

General material for publication in the Pageswapper should be
sent (US mail only -- no "express" services please) to:

Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station
Cambridge, MA 02139-0901
USA

Preference is given to material submitted as machine-readable
text (best is Runoff source). Line length should not exceed 64
characters. Please do not submit program source, as that is
better distributed on the VAX SIG tape.

Change of address, reports of non-receipt, and other circulation
correspondence should be sent to:

DECUS U.S. Chapter
Attention: Publications Department
249 Northboro Road (BP02)
Marlborough, MA 01752
USA

Only if discrepancies of the mailing system are reported can
they be analyzed and corrected.

Editor's Workfile

by Larry Kilgallen, Pageswapper Editor

DOD Security Rating for VMS -

Product Evaluation Bulletin number CSC-PB-01-85 dated October
28, 1985 states that the US government security evaluation of
VAX/VMS Version 4.2 for the C2 class is scheduled for completion
"by the end of the second quarter of FY1986". I think that
means the end of March, but I am sure those to whom this matters
already have good information as to how US government fiscal
years run.

DEC Electronic Store update -

On December 6, to my great surprise, I received something I had
ordered from the Electronic Store (I had been invoiced for it in
August). With that as a model, I am hoping to receive my July
order of Pascal (invoiced in November) sometime in the next
three months.

Enhancing the VAXTPU EDT Keypad Emulator

by
Michael L. Penix
and
Richard D. Piccard

Kalamazoo College
Kalamazoo, Michigan
December 1, 1985

Abstract

We report the development of customizations to
the VAXTPU EDT Keypad emulator that closely
parallel those reported in the past for EDT
itself: word processing enhancements,
multi-buffer operations, etc. In addition, the
reported customizations include dual-window
operation and overstrike mode text entry. We
discuss the choices made in coding, some basic
techniques for debugging with TPU, and several
portions of the code itself. This article, the
code, and a companion article describing the
customized environment are being submitted for
inclusion in the VAX SIG Symposium tape for the
December, 1985 DECUS Symposium.

Introduction

VMS Version 4.2 includes the new VAXTPU Text Processing Utility,
together with two user interfaces, EVE (the Extensible VAX
Editor) and the EDT Keypad Emulator. For those users who have
been working with "plain-vanilla" EDT, the latter is likely to
prove quite adequate, requires very little re-learning, and will
therefore be chosen on the basis of its far superior
performance: in one benchmark it finished in half the elapsed
time while consuming CPU time at one quarter the rate that "real
EDT" consumed it. For those of us, however, who have customized
EDT with extensive EDTINI.EDT files, and especially for managers
who have provided standard EDTINI.EDT files to new users, the
choice is not immediately obvious. Our experience at Kalamazoo
College should prove quite encouraging to others, since we made
the transition in about a week, for over 95% of our usage, and
we had extensive EDTINI.EDT files in place for all of our users.
This paper discusses the choices made in coding the enhancement

files, the techniques that proved effective for us in making the
transition, and several portions of the TPU code itself.
Listing 1 is the EDTINI.EDT file we started with. As can be
seen, we have a variety of commands, one making EDT even more
pleasant to work with for programmers by selecting alternate
delimiters for "WORD" motion and deletion commands, and many
enhancing EDT for word processing (defining keys for sentence
and paragraph motions, for example). This file had evolved from
one presented several years ago in The DEC Professional.

Choices

TPU provides two layers of enhancement, "section" and "command"
files. Section files are pre-compiled, and therefore should be
chosen for the stable bulk of the enhancements. The
DIGITAL-supplied section file for the EDT Keypad Emulator is
SYS$LIBRARY:EDTSECINI. (The filename extension is .TPU for the
source code and .GBL for the binary code; the default when used
by TPU as a section file is .GBL and when used as a command file
is .TPU.) TPU looks for a section file in SYS$LIBRARY: named
TPUSECINI.GBL unless the command line includes a /NOSECTION or a
/SECTION=filename option. Command files are read-in and
compiled at edit session startup, and should therefore be kept
short and used for the volatile portion of the enhancements.
TPU looks for a command file named TPUINI.TPU in the current
default directory, unless the command line includes a
/COMMAND=filename option, or a /NOCOMMAND option. (If you
specify /NOSECTION/NOCOMMAND, then the only action possible is
to exit by CTRL/Y.) A simple method to put a section file to
work is to define a DCL symbol by a line such as the following
at some point in a login command file:

```
$ ED*IT  :==  EDIT/TPU/SECTION=filename
```

Since no /COMMAND option is specified, it will search the
current default directory, as mentioned above, for TPUINI.TPU.
Although our ultimate goal was of course the creation of a
custom section file, we worked for quite a while with a (rapidly
growing!) command file. The time to turn the nascent section
file into a real one is when the start-up delay becomes
irritating, or when you are far enough along that you want
others to start trying to use the product of your efforts. We
will discuss below the modifications needed to change a working
command file into a working section file. The most reasonable
items to leave in the command file are the two items most likely
to be modified by individuals: the word wrap margin and the
first tab size. These should be initialized in the section
file, also, so that new converts need not have a copy of
TPUINI.TPU in their directories. In order to assist ambitious

users by providing examples, a few stable routines and key
definitions should also be placed in the version of TPUINI.TPU
that is made available for users to copy.

Techniques

Probably the key technique to rapid implementation is the prompt
switching over to the partly customized editor as soon as
possible. As you work with it to implement further changes or
to refine those already in place, the lack of a customization
you are used to in real EDT will act as a goad to keep working.
It will also ensure that you implement the most important (i.e.,
heavily used) features first. We have followed the method
discribed in the VAXTPU Users Guide for creating new section
files layered on top of existing ones. In particular, our file
"KAZSECINI" is layered on their "EDTSECINI". During the early
development phase, then, we dealt with our code as a command
file and used theirs as the section file. Later on, the "old"
portion of our code, together with their code, served as section
file and the "new" portion of our code served as command file or
was the file being edited. The advantages of layering deserve
some discussion: we can use their code when it serves our
purpose and replace it if we want to. The design of TPU as a
programming language includes named procedures as well as local
and global variables, and DIGITAL's code in EDTSECINI.TPU is
careful to start the names of all procedures and global
variables with "edt$", so by observing reasonable naming
conventions we can keep track of which portions of our code are
vulnerable to changes in theirs. We chose to name all of our
procedures and global variables with names starting "KAZ$".
During the first development phase, when all customizations are
in the form of a command file, the structure is as shown in Fig.
1: procedure declarations and code, followed by key definitions
to implement those procedures, and finally assignment statements
initializing the global variables. During this phase the file
is used as the command file, either by naming it TPUINI.TPU or
by defining a symbol for DCL that includes specific designation
of the command file. In the later phase, when the initial
customizations are in the form of a section file, the structure
is as shown in Fig. 2: first, a procedure named TPU$LOCAL_INIT
(see below), second, all the other procedure declarations and
code, third, a procedure (in our case named KAZ$DEFINE_KEYS)
that, as you might suspect from its name, includes all of the
key definitions, and finally, the commands to SAVE the current
context in a named file and to QUIT. The filename given should
be different, either in directory or extension or both, from the
current production section file. The DIGITAL file EDTSECINI
ends with a call to TPU$LOCAL_INIT, as well as including an
empty definition of that procedure. By including a procedure

with that name in our file, we replace their empty one. This procedure includes initialization of our global variables, re-initialization of some of the EDT$ global variables (changing the default first tab size and word wrapping margin, for example), and a call to our procedure KAZ$DEFINE_KEYS. Although the following sequence may seem cumbersome in writing, it works well in practice. The main problem is the obscurity of the error messages that TPU provides, a traditional problem with the first release of any compiler which is compounded here by the limited size of the message buffer and the consequent default terseness of those messages. It is therefore essential to maintain efficiency by <u>incremental</u> enhancements, checking each new procedure and key definition as you go along. As few changes as possible should be made at one time! For the sake of discussion, assume that these logical names have been defined: WORK: for the device and directory containing the evolving section file, and GOOD: for the device and directory containing the section file and standard command file most recently released for public use. Several DCL symbols are useful, as well; the following lines were included in the author's LOGIN.COM:

```
$ NEWTPU  :==  EDIT/TPU/SECTION=SYS$LIBRARY:EDTSECINI-
                /COMMAND=WORK:KAZSECINI
$!
$ NED     :==  EDIT/TPU/SECTION=SYS$LOGIN:KAZSECINI.NEW-
                /COMMAND=GOOD:TPUINI
$!
$ WED     :==  EDIT/TPU/SECTION=SYS$LOGIN:KAZSECINI.GOOD-
                /COMMAND=GOOD:TPUINI
```

These definitions were used with a SAVE command that specified the new binary file as SYS$LOGIN:KAZSECINI.NEW. Thus, each time we thought the code was correct, we gave the command NEWTPU and watched to see if any error messages were displayed. If all seemed well, then we used the command NED to use the new editor and tried out the modified or newly introduced routines. If all was well, we RENAMED the .NEW file to .GOOD, thereby setting aside a working enhancement. At the same time a copy of the source code was set aside with a suitably distinctive name in WORK:. From then on routine editing was done with the command WED, using those working enhancements. As each round of enhancement is completed, WORK:KAZSECINI.TPU is copied to GOOD:* and SYSLOGIN:KAZSECINI.GOOD is copied to GOOD:*.NEW. Then the protection of GOOD:KAZSECINI.NEW is set correctly and finally GOOD:KAZSECINI.NEW is RENAMED to *.GBL, thereby replacing the one in use. If there is a mistake in the code (even as simple as a missing ";" termination for a statement), the only message usually visible is "compilation aborted at line 1." In order to debug at all efficiently, we took the advice offered us by the Customer Support Center and used the following command sequence from the "TPU Command: " prompt elicited by <PF1> + <KP7>:

```
SET (INFORMATIONAL,ON)
COMPILE(CURRENT_BUFFER)
```

and then, if errors are reported, type CTRL/Z and at the "*" prompt, "= message", to inspect the message buffer (our code includes a less clumsy way to switch buffers, <PF1> + B). If the buffer being compiled includes routines with names the same as any already in place (from DIGITAL's EDTSECINI, or the local customizations, or previous compilations during the same edit session), an informational message that the old procedure was replaced will be generated. Since the message buffer is limited in size, the unit being compiled should ordinarily be limited to as few procedures as possible. For enhancements, our routine is to write the new procedure and the key-binding command in a separate file, compiling and re-compiling as we go. When it seems logically complete and generates no compiler errors, we cut it out of that buffer and paste it into the growing section in the main buffer, placing it at the end of the regular procedure declarations and finally cutting and pasting the key-binding statement into the procedure KAZ$DEFINE_KEYS. In the case of a flawed procedure that we seek to correct, once the full file is in place, we create a separate file with the suspect procedure as its only contents. Then we edit that file. After modifying it we give the TPU command "COMPILE(MAIN_BUFFER)", and then (if no compiler errors were generated) select an alternate buffer and include text from a file known to elicit the misbehavior whose correction we seek. Because the revised procedure has the standard name, the compilation of the main buffer re-defined the procedure that will be executed upon typing of the standard key sequence bound to that procedure.

Code

Search Patterns

A search pattern may be as simple as the string being sought itself, or it may include TPU builtins and operators to <u>describe</u> that string. In search patterns the symbol '&' <u>is the</u> CONCATENATION operator. Thus the pattern defined by

        'a' _& 'b' _& 'c'

is identical to that defined by 'abc'. The symbol '|' is the OR or ALTERNATION operator. Thus the pattern defined by

        'a' | 'A'

will be matched by any a, regardless of the case. These two
symbols may be used to form more complex search patterns. The
pattern

('a'|'A') _& ('b'|'B')

will match any sequence 'ab' regardless of the case of either
letter. The VAXTPU Reference Manual, section 2.8.4, "Modes of
Searching", discusses incremental and seek searching. The
latter is the default because it is more efficient if it will
produce the desired result. They differ on the handling of
alternate parts of the pattern. This will matter only if there
may be more than one location in the buffer that matches the
pattern. Seek searching essentially "multiplies out" the
pattern, like the distributive law of algebra, producing a set
of pure concatenation possibilities. A search is then
performed, through to the end of the buffer for each of those
possibilities in turn. The overall search terminates
successfully upon encountering of the first instance of any of
these possibilities. Thus, an earlier instance of a later
possibility will be skipped over by a seek mode search if there
is any instance of the earlier possibility. Incremental mode
searching, on the other hand, will terminate with a successful
match at the first instance of a string matching any of the
possiblities. Any alternation that is enclosed in parentheses
and preceded by a concatenation will be searched using the
incremental mode. This is why we have on occasion started our
patterns with the string ''&.

Bug Work-arounds

With help from William White of the Colorado Customer Support
Center, we have implemented a work-around for a bug in the FILL
command. The problem manifests itself whenever the beginning of
the SELECTED region is at the start of or within a word that
ends beyond the margin for wrapping. The result of executing
the FILL command under these circumstances is that a line break
is inserted at the start of the select region, whether it is
beyond the margin for wrapping or not, and whether it is in the
middle of a word or not! The concept of the work-around is
simply to ensure that the beginning of the SELECT region is
moved to the start of the line before executing the FILL
command. The work-around is implemented by copying DIGITAL's
procedure EDT$PRESERVE_BLANKS(flag), creating a procedure named
KAZ$FIND_BEG_OF_LINE(b_mark), and then modifying
EDT$PRESERVE_BLANKS, by adding a call to KAZ$FIND_BEG_OF_LINE,
at the third line of the main sequence of the code, just prior
to the call to EDT$SKIP_LEADING_SPACES(b_mark).
KAZ$FIND_BEG_OF_LINE moves b_mark from its original location to

the beginning of the line that it had been in.

Replacement Customizations

The following procedures are among those that implemented
equivalent functions to those that we already had in place
through EDTINI.EDT.

Word Delimiter Changes

PROCEDURE KAZ$SWAP_DELIM The procedure KAZ$SWAP_DELIM allows the
user to toggle back and forth between two sets of characters to
be taken as the delimiters between words. The two sets are
optimized for word processing and for programming. The word
delimiters in word processing are a space, a tab, a form-feed, a
line-feed, a carriage return, and a vertical tab. The word
delimiters for programming include all the above as well as the
characters "/<>[]{},.:*&!;+-=()\|'" which constitute most of the
"algebraic punctuation" that is used in FORTRAN, DCL, and PASCAL
programs. DIGITAL's code includes a global variable EDT$X_WORD
that is the set of characters used to identify word breaks. We
maintain a variable called KAZ$WORD_DELIM with the value of
either 'text' or 'program' to indicate the current editing
context: word processing or programming. Procedure
KAZ$SWAP_DELIM uses an if-then-else statement to determine the
current editing context from the variable KAZ$WORD_DELIM and
then toggles the two variables, EDT$X_WORD and KAZ$WORD_DELIM.
For instance, if upon entering this procedure KAZ$SWAP_DELIM has
the value 'text', then the old context is word processing. The
programming delimiters are assigned to EDT$X_WORD and the value
'program' is assigned to KAZ$WORD_DELIM, indicating the new
editing context.

Word Processing Conveniences

DEFINE__KEY ('READ__FILE ('' STANDARD.RNO'')',
            KEY__NAME('R',SHIFT__KEY),
            "Insert a copy of the file STANDARD.RNO.");

This defines <PF1> + R̂ to include a file named 'STANDARD.RNO' on
the lines preceding the line containing the current position of
the cursor. This filename results from the routine usage of
this command to include a user's standard RUNOFF
initializations. Obviously, any sort of contents could be
maintained in that file. This definition can be easily modified
to include any frequently used file into the current editing
buffer by substituting the appropriate name for STANDARD.RNO.
Logical name assignments would provide yet another mechanism for
generalizing the use of this command. PROCEDURE KAZ$GET_OUT
This procedure eliminates the following command sequence:
"CTRL/Z + EXIT + <RETURN>". This command sequence does a normal
exit out of TPU, saves the current buffer under the filename
that was used at the starting of the editing session and deletes
the journal file. We have chosen <PF1> + CTRL/Z to cause the
execution of this procedure. The first line,
"write_file(main_buffer)", writes out the main buffer. The
second line, "set(no_write,main_buffer,on)", indicates that upon
the termination of the editing session the main_buffer is not to
be written out, since we just performed that function in the
previous line. The last line, "exit", simply tells TPU that the
editing session is to be terminated and the journal file is to
be deleted. The TPU command "quit" would have worked equally
well here because we took care of writing out the buffer
ourselves. KAZ$FILL_PARAG This permits an entire paragraph to
be re-filled at once. We take real EDT's default definition of
a paragraph: a block of text bounded by a line containing no
characters, or by the beginning or end of the buffer. The right
margin used is the same as the current EDT "wrap" emulation.
The cursor is returned to the original location in the buffer.
The entire operation is performed without ongoing screen
updating. Our procedure to move to the beginning of the current
paragraph is used to establish the start of the range to be
filled. Then a search is conducted forwards for the end of the
paragraph. If it fails, the paragraph ends at the end of the
buffer. If it succeeds, we move back one character to avoid
swallowing the blank line between paragraphs. The word
separators used for the fill operation are the current word
delimiters, EDT$X_WORD. KAZ$END_SENT First a short discussion
on the search command is warranted. Search uses three
parameters: the first parameter is the search pattern. The
second parameter is the direction of the search. The third
parameter is a qualifier on the search. In this proceedure the
direction is always FORWARD and the qualifier is always EXACT.
The search command, as used in this procedure, returns the range
of the first exact pattern match that is forward of the cursor
position. The returned range will be 0 if the pattern was not
found. In KAZ$END_SENT, the search command is used three times.
The first search looks for the end-of-sentence delimiters that
are defined by the search pattern KAZ$SENT_DELIM. One character
string that will match that pattern is ". ", a period followed
by a space character. The initialization of KAZ$END_SENT will
be described later. In this procedure, if the pattern is not

found, i.e., the search range returned to end_sent_range was 0,
then the procedure is exited. If the pattern is found, then the
cursor is positioned to the beginning of the delimiter. The
second search looks for either a space character or the end of
the line. The cursor is then positioned to the beginning of
that pattern if it was found (thereby skipping over the printing
characters of the sentence delimiter) or the procedure is
exited. The third search looks for either line_begin or the
first non-space character (thereby finding the beginning of the
next sentence). If the next sentence is found, the cursor is
positioned there, otherwise the procedure is exited.
KAZ$SENT_DELIM is initialized in procedure TPU$LOCAL_INIT. The
search pattern consists of the concatenation of the empty string
(to force incremental searching) with two alternations: first,
the punctuation, '.', '?', or '!', and second the conclusion
which may include a "concealing" character such as ']', '}',
')', or '"', and terminates with a space or line_end. The
second alternation is rather extended, since the various
combinations of concealment and termination are all explicitly
given. This implementation is significantly more versatile than
true EDT's +SEN and -SEN commands, since those do not recognize
any "concealed" sentences.

New and Wonderful Options

KAZ$WINDOWS This procedure permits selection of single- or
double-window editing. After interactively specifying the
number of windows, the user is asked for the name of the second
buffer. If the buffer is new, the user is asked for the name of
the file to edit and whether that file should be written to disk
upon exit. The windows and buffers are established and mapped,
and keys are suitably re-defined for moving between buffers.
Global variables are established containing the names of the
first and second windows and buffers. In normal usage, before
executing KAZ$WINDOWS, we define PF1 + M to execute a procedure
that moves the cursor to the main buffer and map that buffer to
the main window. That procedure was created by copying the
support routine for the EDT line mode "= buffer" command from
page 22 of EDTSECINI, and then specializing it to go for the
main buffer. In normal usage, before executing KAZ$WINDOWS, we
define PF1 + B to execute a procedure (even more closely modeled
on the "= buffer" support from EDTSECINI) that prompts for the
name of the buffer, moves the cursor to it, and maps it to the
main window. Thus, those keys are routinely used to return to
the main buffer or to shift to a named buffer. KAZ$WINDOWS
restores these definitions whenever single-window editing is
specified. When dual-window operation is to commence, PF 1 + M
is defined to move to the first (upper) window, and PF1 + B is
defined to move to the second (lower) window. We have defined

PF1 + <UP ARROW> to jump back a screen, and PF1 + <DOWN ARROR>
to jump forward a screen. This is implemented using a global
variable, KAZ$WINDOWSIZE, whose value is initialized to 21 for
single-window editing and is set to 10 for dual-window editing.
The action of the screen jump command is thereby adjusted to
avoid either overlap or skipped lines as one moves through a
buffer. The sizes of the windows are adjusted appropriately to
allow for status lines at the bottom of each window, clarifying
the boundary between them. The scrolling limits are set so that
a reasonable amount of context is presented before and after the
cursor, while still permitting some range of motion without
stimulating excessive output to the terminal. KAZ$OVERSTRIKE
This procedure is used to swap between the overstrike and insert
modes using one key definition rather than using a key
definition for each mode. The procedure KAZ$OVERSTRIKE uses a
global variable called KAZ$ENTRY_MODE to indicate the current
editing context. This allows the defined key to take on a dual
function depending on which 'state' the variable KAZ$ENTRY_MODE
is in. For example: if KAZ$ENTRY_MODE is 'insert', the current
editing context is insert mode and the procedure should switch
into overstrike mode. In the procedure an if statement
implements the above logic; if the mode is not 'insert' then the
else clause is executed to switch to 'insert' mode. The
commands

```
        SET(OVERSTRIKE, CURRENT__BUFFER); and
        SET(INSERT, CURRENT__BUFFER);
```

switch the current editing context of the current buffer to
overstrike and insert respectively. After the editing context
is set, then the variable KAZ$ENTRY_MODE is updated to
'overstrike' or 'insert', respectively. Conclusion The VAX Text
Processing Utility has certainly demonstrated its utility! It
provides virtually the full functionality of the EDT Keypad
environment, and can be readily enhanced both in ways that EDT
can and in ways that EDT cannot. While providing this superb
editing capability, it consumes significantly less CPU time than
EDT itself to do the same job. Thus users see a more friendly
system that responds faster, too. Acknowledgement We wish to
express our thanks to the staff of the Customer Support Center
in Colorado Springs, several of whom have provided quite useful
suggestions. The file EDTSECINI.TPU provided with VMS 4.2
contained many useful examples. In fact, some portions of the
code we used were simply copied from there and modified
slightly. A variety of useful examples from the VAXTPU Users
Guide have also found their way into our code. The users of
Kalamazoo College's Educational Computing VAX system tolerated
being forced from EDT onto an early version of the
customizations reported here, and have survived the various
stages of refinement or brutalization it has gone through since,
so to them we owe especial thanks.

Enhancing VAXTPU EDT   M.L. Penix and R.D. Piccard

```
+--------------------------------------+
|                                      |
|              Procedure               |
|                 .                    |
|                 .                    |
|                 .                    |
|            Endprocedure              |
|                                      |
+--------------------------------------+

+--------------------------------------+
|                                      |
|           Key-definitions            |
|                                      |
+--------------------------------------+

+--------------------------------------+
|                                      |
|         Assignment statements        |
|                                      |
+--------------------------------------+
```
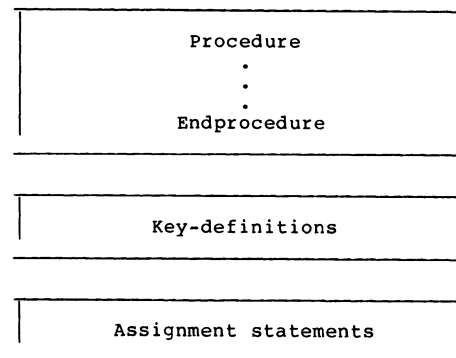
Figure 1:   TPU Command File Organization

Enhancing VAXTPU EDT   M.L. Penix and R.D. Piccard

```
┌─────────────────────────────────────┐
│                                      │
│      Procedure TPU$LOCAL_INIT        │
│                                      │
└─────────────────────────────────────┘

┌─────────────────────────────────────┐
│                                      │
│          other procedures            │
│                                      │
└─────────────────────────────────────┘

┌─────────────────────────────────────┐
│                                      │
│      Procedure KAZ$DEFINE_KEYS       │
│                                      │
└─────────────────────────────────────┘

┌─────────────────────────────────────┐
│                                      │
│    SAVE SYS$LOGIN:filename.NEW       │
│  QUIT                                │
└─────────────────────────────────────┘
```
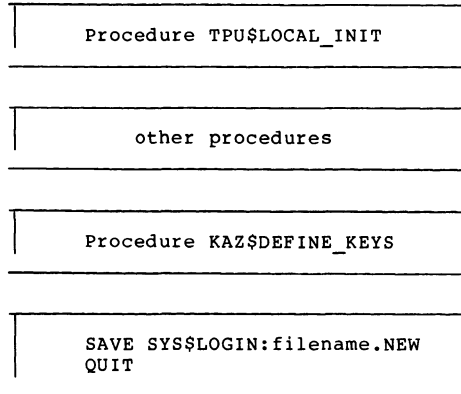
Figure 2:   TPU Section File Organization

Spring 1986 System Improvement Requests

```
*** VOTE *** VOTE *** VOTE *** VOTE *** VOTE *** VOTE ***
*** VOTE *** VOTE *** VOTE *** VOTE *** VOTE *** VOTE ***
*** VOTE *** VOTE *** VOTE *** VOTE *** VOTE *** VOTE ***
```

Gary L. Grebus

HOLD IT! DON'T PUT THIS OFF! THE DEADLINE IS APRIL 1.   You
have an opinion about what is right or wrong with VAX. Here is
your chance to influence the directions of future   DEC
development.   The VAX Systems SIG System Improvement Request
(SIR) program is an important method for the VAX user community
to provide input to Digital. Your opinion is important, and
every ballot adds to the influence of the SIR program.

On the following pages, you will find the current collection of
System Improvement Requests.   Please take the time to review
these SIR's and assess their effect on your use of VAX's.   Then
indicate   your   preferences as described below.   THE NEW,
SIMPLIFIED BALLOT FORM APPEARS THE "QUESTIONNAIRE" SECTION OF
THIS NEWSLETTER.   Also, please fill out the questionnaire
portion of the ballot. This information is important to DEC, as
it points out which requests are important to a particular
segment of the VAX community.

The returns from this ballot will be totalled, and Digital will
provide a formal response to the 10 items which receive the most
votes. The results and DEC's responses will be given at the VAX
SYSTEM SIG SYSTEM IMPROVEMENT REQUESTS session of the Spring
1986 DECUS Symposium in Dallas. At the same session, we also
hope to have an update on the status of past top-10 items.

Instructions For Voting

The ballot form contains two sections, a "support" section and an "oppose" section. To indicate your support for an SIR, enter its number in the "support" section of the ballot. You may list from zero to fifteen SIR's in this section. To indicate your opposition to an SIR you consider detrimental, enter its number in the "oppose" section. You may list from zero to five SIR's in this section.

Please return your ballot IMMEDIATELY. The very tight schedule before the Dallas Symposium makes this a tough ballot to do. To allow time for DEC to respond, **BALLOTS RECEIVED AFTER APRIL 1 MAY NOT BE COUNTED.**

Any ballot not specifying a DECUS membership number will not be counted. Only one ballot per member will be accepted.

SIG members who have voted in past ballots may notice a radical change in the method of voting on this ballot. This is an experiment to (hopefully) make it less time-consuming to participate in the SIR ballot. The continued decline in the number of ballots returned has prompted such changes. Your comments are welcomed.

Finding the Ballot

As with all "forms" in the combined newsletter package, the ballot may be found at the back of the entire volume you are now reading on the QU- pages. Nominally it goes in alphabetic order by SIGs, putting VAX at the end, but there have been exceptions.

VMS Internals

SIR: S86-1

Abstract: Provide routines for accessing the DECnet configuration

Description: There should be some way of obtaining information about the current configuration of the network: which nodes are available, number of hops, next hop to node, etc. through runtime calls. The only currently available method is to direct the output of SHOW NETWORK to a file and parse the file.

SIR: S86-2

Abstract: Provide system wide event flag clusters.

Description: UIC group based event flags cause considerable inconvenience when an existing event-flag based application must be expanded to serve users in multiple UIC groups. It should be possible to create and manipulate system-wide common event flag clusters. Use of this feature should be controlled by a privilege or quota independent of existing controls on system-wide resources.

SIR: S86-3

Abstract: Provide better support for hardcopy terminals through SMG routines.

Description: In release 4.0 of VMS, the TERMTABLE entries for DEC hardcopy terminals' do not contain all of the applicable escape sequences for the devices. For instance, many hardcopy terminals support automatic underlining and boldfacing, similar to the DEC CRT's. Even if local modifications are made to the TERMTABLE entries, the SMG routines do not use the additional information (say when SMG$SNAPSHOT is called). It would be nice to support the VT100 line drawing character set (or that kind of line drawing) when the device supports it.

SIR: S86-4

Abstract: Allow DUMP files to be moved off the system disk.

Description: A cluster of fully configured 8600's sharing the system generates a need for most of the disk to be dedicated to system DUMP files. Recently announced increases in memory size make this problem worse. The current mechanism of combining dump and paging files is unacceptable, since a shared system disk cannot tolerate

the I/O load generated by a large page file.

SIR: S86-5

Abstract: Support TCP/IP on VMS.

Description: The TCP/IP protocol is probably the most widely
    supported protocol for networking among workstations and
    mid-sized computers. It is frequently necessary to connect
    VAX's into such a mixed-vendor network. The TCP/IP
    protocol should be supported under VMS to allow such
    connections.

SIR: S86-6

Abstract: Provide for wildcard $GETDVI.

Description: The $GETDVI service should provide a wildcard
    facility similar to that available with the other $GETxxx
    services. There is currently no acceptable way to find out
    the device configuration of a system. This is particularly
    true since VMS now has many devices which dynamically
    appear and disappear (virtual terminals, MSCP disks, etc).
    In a VAXcluster, this facility should ideally be able to
    show all devices connected to the cluster, including
    node-local devices.

SIR: S86-7

Abstract: Eliminate the necessity for trailing commas in
    system service argument lists.

Description: Even when one or more arguments at the end of a
    system service argument list are optional, it is still
    necessary (according to the system services manual) to
    include trailing commas in the system service call. (This
    is unlike the run-time library routines, where it is not
    necessary to include the commas.) This is annoying, and
    error prone. Could this be changed?

SIR: S86-8

Abstract: Provide a terminal attribute to request filtering of
    BELL characters.

Description: It would be useful to provide a terminal
    attribute which supressed sending of the BELL character to
    the terminal. Having many terminals beeping in a small
    area can be distracting. Ideally, it should be possible to
    specify an alternate sequence to be sent in place of the
    BELL so that CRT's for example could display some visual
    equivalent.

SIR: S86-9

Abstract: Allow a privileged user the ability to link one
    terminal to another.

Description: VMS should support a facility which would allow a
    privileged user to link his or her terminal to another
    terminal. This link would at minimum allow the privileged
    user to issue commands as if they were typed from the other
    keyboard. This capability would be useful to "cleanup"
    whatever was running on the remote terminal before its
    process was deleted. Ideally, the facility should also
    allow the privileged user to see all output directed to the
    target terminal. This would allow for fully interactive
    "hand-holding" or consulting for user problems.

SIR: S86-10

Abstract: Provide a facility to list the VMS timer queue.

Description: VMS should provide a service and a DCL command to
    list the entries in the timer queue. Appropriate privilege
    (e.g. WORLD) would be needed to list other users' queue
    entries. This would be useful to verify that periodic
    tasks were correctly scheduled, or that an idle process was
    in fact waiting for a timed event.

SIR: S86-11

Abstract: CREATE/DIR should set owner protection to RWED, not
    only RWE.

Description: A VAX directory file cannot be deleted until all
    files have been removed from it. Therefore, it is not a
    hazard to have OWNER:DELETE access to the file as it was in
    earlier versions of VMS. The CREATE/DIRECTORY command
    should, by default, grant OWNER:DELETE access. This
    simplifies cleanup of files and directories, since an extra
    step (protection-change on only directories) is avoided.

SIR: S86-12

Abstract: Provide a supported virtual memory disk driver for
    VMS.

Description: VMS should provide a supported device driver
    which implements a disk using pages of virtual memory.
    Optionally, some or all of the virtual disk could be locked
    into physical memory. Such a facility could provide very
    fast access for frequently used files. Note that this
    facility is orthogonal to any file system or RMS caches --
    the benefit would apply for any access method or file
    structure, even simple sequential, text files. Suitable

privileges and quotas should be required to create and
destroy such a device (the scope seems similar to permanent
mailboxes). The falling prices of semiconductor memory
make this facility attractive when short access times are
vital.

SIR: S86-13

Abstract: SYSGEN should flag autoconfigured devices in
SHOW/CONFIG.

Description: It would be potentially useful if the SYSGEN
SHOW/CONFIG command could indicate if a device was
autoconfigured.

SIR: S86-14

Abstract: All utilities should use a standard format for
printable output.

Description: Printable output generated by VAX utilities and
compilers comes in a great variety of record formats and
carriage control conventions. A particularly awkward
convention is the use of embedded ASCII control characters
to generate multiple print lines from a single record.
There appears to be no standard for this or any other
mechanism. As a result it is very difficult to print
"printable" output on non-DEC printers or transmit it
through heterogeneous networks. Digital should document a
standard record format and carriage control convention and
modify all facilities to conform to this convention. As a
alternative, Digital should provide a utility which
converts all currently used formats into a standard format.
It seems that this functionality currently exists,
distributed between the print symbiont, device driver, and
"DEC standard" printers.

SIR: S86-15

Abstract: Proposed changes to SMG$READ_STRING and
SMG$READ_COMPOSED_LINE screen management routines.

Description: When using the SMG$READ_STRING and
SMG$READ_COMPOSED_LINE screen management routines, when the
display-id argument is included, there is a restriction
that the cursor for the virtual display associated with the
routine(s) must be in column 1, and also that this virtual
displays to its right. It would be very useful if these
restrictions did not exist.

SIR: S86-16

Abstract: Provide a call interface that parallels the DCL show
queue functionality.

Description: The interface to the symbiont manager is useful
as long as you already have information (i.e., the job
i.d.). There should be a call interface that permits an
application to "look up" what jobs are in the queue(s).

SIR: S86-17

Abstract: The VMS source kit should be cheaper.

Description: We need to have parts of the source kit available
for SEARCHing to quickly find out how VMS does certain
things. We are a DEC OEM and third-party supplier; we have
a non-disclosure agreement for a new type of hardware for
which we will be building compatible products with DEC's
cooperation and assistance. Unfortunately this cooperation
and assistance does not extend to helping us to write the
system software (device drivers) which are necessary to use
our product. $25,000 seems excessive for the source kit;
it seems designed to discourage people from buying it (it
certainly has discouraged us) rather than as a reasonable
price based on the cost of production and support (since
DEC doesn't support it, or anyone's use of it). I
understand that DEC does not want versions of VMS to
proliferate the way Unix systems have. This could be
addressed by leaving out something that was necessary (or
very, very, very helpful) to actually use the kit to build
a system. The system and module build command procedures
come to mind. A price of $5,000 would surely result in
more than five times as many sales of the source kit as DEC
currently gets. Thus the suggestion makes sense
financially.

From a philosophy standpoint it makes sense too. DEC has
been successful partly because it has been relatively easy
for third-party suppliers to get information on how to do
things that complement DEC hardware and software, filling
market niches that DEC is too big or too slow to respond to
and thereby enabling sales of DEC systems to customers who
would otherwise have to look elsewhere. It seems
unreasonable for DEC not to follow this philosophy in VMS,
their flagship operating system and the system they are
trying to sell to almost everyone. To say "you can look at
the microfiche" is not responsive; it's hard to issue a
SEARCH command to a fiche reader. To "market" a source kit
that costs more than the low-end configuration of the
MicroVAX II, which will surely put VMS into more sites than
all the other VAXes to date combined, seems to be
completely contrary to DEC's (former?) way of doing

business.

SIR: S86-18

Abstract: Make SWAPPER multi-threaded.

Description: Given increasingly large VAX systems with lots of
    memory and disks it appears as if SWAPPER could become a
    bottleneck when performing large writes of working sets and
    modified pages.   It would be nice if SWAPPER could format
    writes to all of the disks it accesses in parallel.

SIR: S86-19

Abstract: Add description of file to header.

Description: Long file names aren't good for elaborate
    descriptions of files because you have to type them. What
    if
        $SET FILE file/DESCRIPTION="text"
    could put up to 80 or 90 chars in the headers, which could
    be displayed by:
        $DIR/DESCRIP
    or
        $DIR/FULL
    This facility should also be available via RMS and QIO
    interfaces. Ideally, the description would be copied along
    with things like record attribute when you $COPY, $EDIT,
    etc. perhaps by having it appear in an XAB.

SIR: S86-20

Abstract: Support a PRINTABLE attribute on files

Description: Binary-type files can do awful things to
    terminals and printers when PRINTed or TYPEd. An attribute
    which would cause the commands to reject such files unless
    overridden would save many confused and frightened users.
    Modifying just compilers and the linker as a first step
    would probably eliminate 90% of the problem.

    This item appeared as a past top-10 SIR in which DEC
    proposed an alternate solution in which the printing
    software would make an empirical decision about the
    printability of a file it was given. However, this
    solution seems increasing difficult to implement given the
    proliferation of "intelligent" printing devices.

SIR: S86-21

Abstract: Proposed improvement to RMS user control block run
    time macros

Description: The RMS control block run time macros,
    $fab_store, $nam_store, $rab_store, and $xabxxx_store have
    a number of options which allow one to set various RMS
    attributes (such as fop-cbt, for contiguous best try, in
    the $fab_store macro). There is, however, no way to turn
    these attributes off using these macros. (one has to use
    symbolic offsets, such as fab$l_fop and fab$v_cbt,
    instead). It would be convenient if each store macro had a
    clear, or unset, form of its on/off attributes (such as
    fop=nocbt in the $fab_store macro).

SIR: S86-22

Abstract: Support a per-user logical name table

Description: Extend the logical name table system to provide
    user-level (not group-level) tables which remain after a
    process goes away. They should behave much as group tables
    do but should be unique to a full UIC. An appropriate
    privilege (USERNAM?) may be required since this would
    entail the use of system resources.

SIR: S86-23

Abstract: Better program control of output to sys$error is
    needed, especially for detached processes.

Description: In detached processes it is desirable to merge
    the sys$output and sys$error files. The ways/orders in
    which they are opened causes difficulties. For example,
    two files appear when one is desired. A possible solution
    is to give user programs control over where error output is
    to appear, e.g., traceback output. There is no control
    over how a process opens sys$error.

DCL and Utilities

SIR: S86-24

Abstract: Increase the 255 byte limit on foreign command
    lines.

Description: User commands that utilize the foreign command
    interface to get argument data may now exceed the 255 byte
    limit that is enforced with VMS 4.0. This is now a problem
    since the limit was not enforced in VMS v3.x software and
    longer arguments were working and are now not. Also the

expansion of file names and extensions to 39 characters adds to the need for longer argument strings.

SIR: S86-25

Abstract: Enhancements to the VMS MAIL facility.

Description: There are several enhancements to mail that would generally increase the ease of use and functionality of mail:

1. Include the subject of the mail message in the broadcast message mail sends to logged in users.

2. In captive accounts which run from controlled command procedures some way to prevent users from using the spawn capability in mail is needed. This facility might also be useful elsewhere, restricting the subprocess count is not a solution since the running application may need to create a subprocess.

3. The ability to delete more than one message at a time would be desireable. I.E. DELETE 1:10 to remove message 1 thru 10 or DELETE 1,3,5,7 to remove messages 1, 3, 5, and 7.

4. Some form of search facility on the mail folders to aid in finding specific entries is needed. This could be restricted to the folder names or extended as far as searching thru the actual messages themselves. The syntax and command capabilities in the VMS SEARCH command are an excellent example.

5. When using the @FILESPEC method of sending mail to many people it is very desireable to have the option to specify that the detailed list expansion be included in the message in some form. This will allow a newly added member of the list to see who comprises the list. Additionally the number of redundantly forwarded mail messages could be reduced.

SIR: S86-26

Abstract: Show open files for a process.

Description: It is frequently necessary to attempt to determine the files that a specified process has open. With many devices on a system this can be a time consuming exercise to search through all of them. A more desireable method would be some form of show command, possibly SHOW FILES/PROCESS=pid or SHOW PROCESS/ID=PID/FILES

SIR: S86-27

Abstract: SHOW PROCESS/CONT is inadequate for large images.

Description: When watching large processes with SHOW PROCESS/CONTINUOUS the actual memory that is in use by the process is not shown by the fixed window. Some form of window specification where the user declares the memory range to be shown or some form of automatic selection of the window address is needed.

SIR: S86-28

Abstract: Add restarting to DIRECTORY command.

Description: Some mechanism is needed for either specifying a start point for the command or a restart from last file shown if interrupted by a control_c. For example, DIRECTORY/FROM=oldprog would start at the first file entry that in the normal sort order came after oldprog. Additionally a /UNTIL=stoptext would be very useful. With both features management of large directories would be very easy. It is understood that subdirectories are a more efficient way of organizing files, however this is not always possible.

SIR: S86-29

Abstract: Enhance DCL command recall.

Description: Additional desirable features in the command line recall facility:

1. When a command is recalled from the saved commands, move it to the top of the stack and erase it from the position it occupied. This eliminates loosing the oldest command. This also changes the recall facility to remember the last 20 unique commands.

2. Allow some method of specifying the number of commands to store.

3. Allow a method of ignoring short commands that are less than a specified length.

SIR: S86-30

Abstract: SHOW SYSTEM display should have the old form of UIC numbers.

Description: The enhanced version of SHOW SYSTEM maps UIC
entries into named entries. Some method of getting the old
UIC entry back is needed. Possibly SHOW SYSTEM/OLD would
display the numeric UIC entries only, UIC entries that are
not of the old numeric form would not be displayable by
this command.

SIR: S86-31

Abstract: Enhanced DCl substitution and parsing capabilities.

Description: DCL would be more useful to advanced and expert
users if the following concepts were implemented.

1. Direct substitution of parameters in DCL commands.

2. Multiple command entries on one DCL line.

For example:
         BUILD:==MACRO/LIST ^P1 |LINK/MAP ^P1

         BUILD FOO expands to MACRO/LIST FOO
                              LINK/MAP FOO
The syntax shown is only for the purposes of writing an
example.

SIR: S86-32

Abstract: Enhance the DEASSIGN command.

Description: In many cases a simple enhancement to the
DEASSIGN command would simplify writing command procedures
and increase clarity and execution speed.

For example DEASSIGN LNAME1,LNAME2,LNAME3 would deassign
all three logical names and is easier to write than.

              DEASSIGN LNAME1
              DEASSIGN LNAME2
              DEASSIGN LNAME3
This general ability to restart the DCL command for more
than one action would be useful in as many places as
possible such as:
              CLOSE LNAME1,LNAME2,LNAME3
              REWIND LNAME1,LNAME2,LNAME3 etc.

                        VAXClusters

SIR: S86-33

Abstract: Provide a feature to keep system times on a
VAXcluster synchronized.

Description: It is extremely important for the time-of-day to
be consistent across all nodes in a VAXcluster. Often,
applications involve cooperating processes on several
nodes. Without synchronized times, it becomes extremely
difficult to consistently handle any operation which
requires timestamping. Even coordinated processing of
accounting or performance data is impossible without
coordinated times. Manual synchronization impossible to do
if high accuracy is required.

SIR: S86-34

Abstract: Provide support for cluster-wide print queues
spooled via a device.

Description: In single node systems, it is possible to declare
a printer device as spooled so that files written to the
device are printed without the necessity of using a PRINT
command. Applications are written to take advantage of
this behavior. In a cluster, this mechanism no longer
works, since the device will not be present on every node
of the cluster. A mechanism is needed which provides a
"dummy" alias for the printer, which appears on every
system. Defining a logical name to include a DECnet
nodename (e.g. DEFINE LP NODE:LPA0:) is not acceptable
since username and account information might not be the
same for the DECnet server process.

SIR: S86-35

Abstract: Provide cluster-wide system management tools.

Description: Digital stresses that a VAXcluster should be
managed as a single domain. To do this, the standard VMS
system management tools need to be expanded to work
cluster-wide. Cluster-wide ACCOUNTING and and cluster-wide
MONITOR are two examples.

SIR: S86-36

Abstract: Provide a capability similar to the NCP TELL to pass
DCL commands to other nodes on a cluster.

Description: It is frequently necessary to execute the same,
or similar, commands on more than one node of a cluster.
This is typically a command that has just been put in
SYSTARTUP.COM, for example an INSTALL command, which is
needed before the next reboot. We are currently using

DECNET and batch jobs which are slow and inconvenient. What is needed is DCL TELL for use on a cluster.

SIR: S86-37

Abstract: Provide high-speed task to task communication on a cluster using SCS, not DECNET.

Description: Some applications need to get large amounts of data (100-300 Kbytes) from one node to another as fast as possible. At present, this can only be done using a shared file or DECNET transfers - neither of which is fast enough. The bandwidth of the CI is fast enough. What is required is the ability to do 'block' mode transfers via the CI. A simple device driver interfacing to SCS would suffice.

Security

SIR: S86-38

Abstract: Mechanism needed that allows one user to grant other users access to a file only via a user-defined image.

Description: Non-privileged users sometimes need to give other non-privileged users controlled access to data files through a program. Through this facility any user would be able to control who could access his data files and what kind of access they may have. In the current system, in order to allow another user to add a record to a file, that user must be given WRITE access to that file, which means he could alter existing data or delete records from the file.

Presently this requires the system manager to install the program with privilege, which is both an administrative nuisance and a security problem, as the privileged image would also have access to other system data files as well as the intended files. This mechanism should be under user control, i.e., the user should be able to determine which images could access a file. For example, the UIC of the image and data file could be required to match before access would be permitted. This could be accomplished by an option on the compiler or linker when the image was being created. It could also be implemented by allowing the system manager to install an image with a particular identifier. Then the user could set up the access control list for that file to permit access by that Identifier. This would be less flexible but would permit a user to allow access from images other than his own, e.g., a data base manager. This capability could also be provided by

file passwords, since the passwords could be imbedded in the programs that were intended to have access to the files; however, file passwords would be difficult to administer and prone to disclosure.

SIR: S86-39

Abstract: Implement mandatory security controls in VMS.

Description: Many VAX systems are being operated by government agencies or contractors and either are processing or need to process classified data. Mandatory security controls are needed in VMS to support such classified processing. An operating system that could be evaluated by the National Computer Security Center at the B1 level or higher would encourage more VAXes to be used for classified processing and make system management much easier for those already doing classified processing on existing systems. The system manager should be able to specify a security level for each user account using the AUTHORIZE utility. When a file is created, it should be given the security level of the creating process, and any subsequent access to the file should be controlled in accordance with the mandatory security policies. If a file is edited or copied, it should retain it's classification. A utility should be provided to allow a person with a special (SECURITY) privilege to change the classification of a file.

SIR: S86-40

Abstract: End-to-end encryption of logical connections within DECnet-VAX.

Description: The assumption made by DECnet that all nodes and communications paths are trustworthy is not viable in many environments. End-to-end encryption of the data portion of network packets is required in these environments to assure that eavesdropping is fruitless, both in Local Area Networks (broadcast) and Wide Area Networks (multi-hop). This encryption should be implemented so as to be transparent to the application programmer and user, i.e., the mechanism should be located in the NSP (or OSI session) layer. New encryption keys should be generated for each logical connection between cooperating, encryption-capable processors. (Some nodes will not be capable of encryption and should be allowed to participate in the network without performing encryption.) Intermediate nodes should not be required to participate in, or be knowledgable of, the key distribution/management or the encryption process. The DES algorithm should be utilized in the near term but should be readily replaced as NBS standards change. Provisions should be made for encryption hardware to boost performance where necessary.

SIR: S86-41

Abstract: DECnet-VAX Node Authentication

Description: In both Local Area and Wide Area Network
, environments, a malicious node can readily assume the
identity of another node. Node passwords are inadequate
protection, are easily circumvented, and are applicable
only to adjacent nodes on point-to-point links. An
encryption based node authentication mechanism is required
(perhaps using the RSA algorithm). This improved
authentication mechanism should provide the (local) node
manager with a high degree of certainty that a remote node
is who/what it claims to be.

The authentication mechanism should also be immune to
eavesdropping by intermediate routing nodes. After this
capability is developed within DECnet-VAX, it will also be
needed in other DECnet implementations, so that routers and
PC workstations can take advantage of enhanced security in
their communications with VAX systems.

SIR: S86-42

Abstract: Record attempted usernames on login failures.

Description: An OPTION is needed that permits the system
manager to record login failures in more detail, including
the ATTEMPTED USERNAME, terminal name, and time, even if
the username is invalid. This would help the system
manager to assess whether login failures are the result of
penetration attempts, noisy lines, or just merely a user
who is having difficulty. The system manager would be
responsible for protecting the accounting file
appropriately to prevent the disclosure of passwords.

This capability must be an OPTION, as large numbers of
system managers have indicated a strong need for this
capability, and many others have expressed an equally
strong concern that passwords might appear in their
accounting files. Both groups of person have valid
arguments and should be given a choice in the recording of
login failure information.

SIR: S86-43

Abstract: Security alarm messages to a file.

Description: Add an option to the Access Control Entries
(ACE's) that specifies a file into which security alarms
for that file/directory are written. This would allow a
user to review security alarms for his own files, rather
than depending on the system manager to perform the

auditing. Of course, security alarms requested by the
system manager via the SET AUDIT command should be written
to the system-wide security log.

SIR: S86-44

Abstract: Prevent users from re-using passwords.

Description: When passwords expire, users can change them to
some other password such as XXXXXX, and can then
immediately change them back to the original password. A
mechanism is needed to prevent users from re-using old
passwords. This might be implemented via a MINIMUM
password lifetime or by maintaining a list of "n" previous
passwords for each user.

SIR: S86-45

Abstract: Control access to printers through ACL's.

Description: ACL's are needed to control the usage of print
devices, print queues, and batch queues. The UIC-based
protections are now available on queues, but ACL's are not,
so the system manager does not have sufficient granularity
in granting access to the system queues and print devices.
ACL's can be placed on physical devices, but they only
control the ability of users to allocate the devices and do
not control their ability to use shared devices such as
printers. This is undesirable in cases where the user is
expected to use some shared devices and not others.

SIR: S86-46

Abstract: ACL's are ignored for images installed open and
header resident.

Description: In some cases, the system manager may wish to
control the use of images (such as compilers and the
linker) through ACL's, but would still like to take
advantages of the efficiencies of installing the images
OPEN and/or HEADER_RESIDENT. In VMS 4.2, ACL's applied to
these images are ignored, even though UIC protection is
enforced. Full ACL protection should be included.

SIR: S86-47

Abstract: Make password validation a user-callable service.

Description: Username/password validation should be made a
user-callable system service or run-time library routine.
Such a routine should one-way encrypt the password, open
the UAF, read the record for the specified username, and
return a status indicating whether the username/password

pair is valid. For security reasons this service should
only be available to users holding system privileges, e.g.
SYSPRV. Such a capability would be useful for applications
requiring users to log into shared or captive accounts,
which would then need to further validate the individual
users.

SIR: S86-48

Abstract: Allow OPER to override LOGINS=0 only if it is a
    DEFAULT privilege.

Description: It is useful to give OPER privilege to certain
    users, so that they can control print and batch queues.
    However, granting OPER as an AUTHORIZED privilege also
    allows that user to login even when logins are disabled.
    VMS should check the DEFAULT privileges (not the AUTHORIZED
    privileges) for an account when determining whether or not
    to allows that user to override disabled logins.

SIR: S86-49

Abstract: Asynchronous data security erase on file deletion.

Description: The data security erase occurs on file deletion
    when the disk volume or an individual file is set to
    /ERASE_ON_DELETE. If the file is large, this erasure can
    take a long time. The erasure should be performed
    asynchronously to the user process, so that a performance
    degradation is not seen by interactive users.

VMS Languages and Tools

SIR: S86-50

Abstract: Allow EDT to set tabs in any column.

Description: Currently, only the first tab stop in EDT can be
    changed. The rest are in multiples of 8 (16, 24, etc.). A
    user should be able to tab to any column. This feature is
    useful when entering data that is not in every 8th column.

SIR: S86-51

Abstract: Provide windowing capability in EDT.

Description: A window capability in EDT would be useful, so
    that several portions of a file could be viewed and
    editted. It should also be possible to view and edit
    multiple files.

SIR: S86-52

Abstract: Provide column editting capability in EDT.

Description: It would be useful to easily insert, delete, or
    swap columns of text in EDT. Currently, these operations
    are quite awkward, compared to inserting, deleting, or
    swapping rows of text. It would also be useful if the
    column number of the current cursor position could be
    displayed.

SIR: S86-53

Abstract: Provide a SPAWN command in EDT.

Description: It would be useful to SPAWN from an editting
    session so that mail could be read, that status of batch
    jobs could be examined, etc.

SIR: S86-54

Abstract: Provide insert and overwrite modes in EDT.

Description: It would be useful if both insert and overwrite
    modes were available, and toggling between them would be
    easy. Overwrite mode would be especially useful in
    editting formatted files.

SIR: S86-55

Abstract: Provide depth-level numbers on Fortran listings.

Description: It would greatly aid debugging programs, if the
    Fortran listing would include the depth level of DO loops
    and the depth level of nested IF's. The level number could
    be printed on the left margin of the listing, to the right
    of source code line number and sequence number.

SIR: S86-56

Abstract: Provide a qualifier on the Fortran compiler that
    would make all integer and real variables and constants use
    64-bit fields.

Description: A qualifier on the Fortran compiler is needed
    that would make all real variables and constants double
    precision, and all integer variables and constants occupy
    64 bits. This qualifier would also cause functions to
    return a 64-bit quantity, unless otherwise specified. The
    IMPLICIT statement is not adequate, since constants are not
    affected.

SIR: S86-57

Abstract: Provide more support for structure variables in
    Fortran.

Description: It would be convenient to have an aggregate
    constant feature that could be used to initialize the
    fields in an aggregate assignment statement. It would also
    be convenient if aggregate fields could be specified in
    formatted I/O statements.

SIR: S86-58

Abstract: Provide a DO UNTIL construction in Fortran.

Description: A DO UNTIL construction is needed in Fortran.
    There are occasions when a DO UNTIL is simpler to use than
    a DO WHILE, and cleaner code will result.

SIR: S86-59

Abstract: Provide a qualifier on the linker that would force
    inclusion of all modules in the library.

Description: It would be useful if the linker could be
    directed to include all modules in a library, when building
    an executable program. This feature is useful when there
    are common routine names in libraries that are used to
    build an executable, and you need to be sure that all
    modules from a particular library are used.

SIR: S86-60

Abstract: Provide source line debugging in MACRO.

Description: The debugger does not always effectively
    translate offsets and other symbolic information. Source
    line debugging would remedy this problem. Source line
    debugging would be useful for programs that make extensive
    use of macros. The capability to STEP and SET BREAK points
    according to source lines would also be useful.

SIR: S86-61

Abstract: Provide LSE Support for VAX MACRO.

Description: The Language-Sensitive Editor should support VAX
    MACRO. Support would include a /DIAGNOSTICS qualifier on
    the MACRO command, assembling from within LSE, and
    templates for the commonly used system services.

SIR: S86-62

Abstract: Provide a word count in the RUNOFF output log.

Description: It would be useful to know how many words are in
    a document that RUNOFF processes. This number could be
    included on the output log file, along with the number of
    pages. This word count would not include section headings,
    chapter headings, etc.

Digital Responds to the Fall 1985 SIR Ballot

Gary Grebus
SIR Coordinator
and
Richard Merewood
VMS Engineering

At the Anaheim Symposium, Digital responded to the VAX
SIG's most recent System Improvement Request ballot. The
ballot was originally published in the September issue of
the Pageswapper. Only 172 ballots were returned, a
disappointing decrease in participation by the SIG
membership. A complete summary of the ballot appeared in
last month's Pageswapper. Below is a summary of the top 10
items along with with Digital's responses, as presented by
Richard Merewood of the VMS Engineering group.

SIR F85-11   Position: 1   Points: 847

Abstract: Provide a means to perform an in-place compression of
a disk.

Response: While the importance of on-line compression is
obvious in retrospect, this is a new request for us in that
it has never appeared so prominently in past SIR ballots.
We do not have any current plans to build such a facility.
However, we understand its importance and we understand
that it will increase as time goes on.

There are a number of difficult problems to deal with,
including coordinating ongoing file activity with the
compression process, and solving the performance problem on
large disks (since disk reorganization is inherently an
n-squared order problem).

We will investigate this for future VMS development.

(Note: comments offered from the audience at the symposium
session suggested that support of ongoing file activity was
not a requirement, particularly in an initial
implementation.)

SIR: F85-55   Position: 2   Points: 588

Abstract: Enhance the SHOW command of AUTHORIZE. Specific
suggestions included the ability to show a particular item
across a group of users, selection of records by the
contents of various items, and better support for
wildcarding UIC's.

Response: It has been recognized that the user friendliness of
the VMS AUTHORIZE utility falls short of requirements, and
therefore a major re-work is being planned.

These suggestions are good ones and they will certainly be
considered.

SIR: F85-06   Position: 3   Points: 494

Abstract: This is actually a layered software specific
improvement request and is not within the realm of the
operating system to address.

However, we agree that the ability to support running
multiple versions of layered products in a given
environment would be an attractive enhancement to the
VAX/VMS system family. As noted in the SIR, many
dependencies such as images, shareable libraries, DCL
tables, etc. must be considered as well as any enhanced or
revised features in a product that would preclude an
existing or new function in that product.

Other ramifications that make the implementation of this
SIR complex are VAXclusters and the various flavors of
clustered environments that might exist in a customer
environment (such as complex database locking scenarios on
both common and non-common disks throughout the cluster).
These pose resource allocation and file/record access
considerations. A new release of a product could introduce
advance features incompatible with databases jointly
accessed with an older version of the product.

Additionally, product validation, test complexity, and
ensuring system compatibility grows rapidly in order to
ensure continued quality in each product.

There are ways of accomplishing this to some extent with
today's products by the use of alternate system roots, if
you system environment is flexible enough to allow this.
These alternate roots, however, must reside on alternate
system disks and cannot be an alternate root on a common
system disk.

VAX Datatrieve accomplishes this by allowing a unique
suffix per additional version (at the installer's request).

In summary, we agree that this is an attractive and useful

feature to add and would consider implementing it on a product by product basis as the design and implementation of each allows.

SIR: F85-27   Position: 4   Points: 477

Abstract: Enhance DELETE command behavior so that when a version number was omitted, the command would be interpreted as ";*/CONFIRM" (as is done in RSX-11.)

Response: Historically, when the VMS DELETE command was planned, it did parallel the behavior of the RSX DELETE command; it originally gave an error message when the ";" was omitted. The RSX version was changed after the VMS command was implemented.

Any inconvenience to the user of the DELETE command caused by the accidental omission of a file version specification is easily minimized by the use of the command recall feature of DCL.

SIR: F85-42   Position: 5   Points: 471

Abstract: Enhance network support in BACKUP. Particularly, support a remote magnetic tape server, rudimentary DECnet capability for standalone BACKUP, and the ability to boot stand-alone BACKUP over Ethernet.

Response: a) Remote BACKUP Server: This is an interesting and useful idea. We will consider it for a future VMS release. While there are some dependencies involved in building such a facility, we do not see any insurmountable problems.

b) Rudimentary DECnet capability for stand-alone BACKUP: Unfortunately, there is no such thing as a rudimentary DECnet capability. DECnet/VAX consists of a considerable number of components, all of which must be present and functioning before any level of DECnet capability is available. Also, the operation of DECnet requires the presence of DCL, RMS, and the file system, non of which are currently present in stand-alone BACKUP. To provide DECnet support in stand-alone BACKUP would require expanding stand-alone BACKUP to incorporate most of the full VMS kernel.

c) Boot stand-alone BACKUP over Ethernet: Because the kernel that stand-alone BACKUP runs on is simply a VMS executive, this amounts to booting the VMS executive and a single-task system over the Ethernet. This is obviously a rather complex job.

We feel that the solutions b) and c) are sufficiently complex and costly that it is unlikely that they will be

implemented. However, we are looking at alternatives to improving system installation and maintenance in the Ethernet environment.

SIR: F85-04   Position: 6   Points: 375

Abstract: Provide enhanced CPU usage reports in MONITOR, particularly breakdown of kernel mode time by system component, and breakdown of per-user CPU time by access mode.

Response: This is a good idea because it eliminates a lot of the guess work that's involved in determining the culprit in a system performance problem.

However, implementation of this type of collection requires many hooks to be placed in the executive and some analysis work to make sure that these hooks do not create new performance problems.

At this point we have not researched any aspect of this feature and have no plans to do so in the immediate future. But, because the information is highly beneficial, it may be a candidate for a future release.

In the meantime, information of this kind can be obtained using the performance analysis products VAX SPM and PCA.

SIR: F85-49   Position: 7   Points: 371

Abstract: Provide a screen editor for AUTHORIZE

Response: Again, it has been recognized that the user friendliness of the VMS AUTHORIZE utility falls short of requirements, and therefore a major re-work is being planned.

This request, or something very similar, is also being given active consideration.

SIR: F85-18   Position: 8   Points: 364

Abstract: Add a class based scheduler to VMS

Response: We also feel that a class-based scheduling capability is very important, and believe that it is applicable for both single-node and cluster configurations. This is particularly true in large timesharing service environments. We also noted with interest that this particular request has risen from 16th to 8th place in the ballot.

Note however, that its implementation requires major and

extensive work in the executive and could therefore only be
shipped in a major release.

. Although preliminary research has taken place, this work is
not included in the modifications expected to be completed
prior to the next rebuild of the executive. Therefore, it
will be some time before this feature is available.

SIR: F85-24   Position: 9   Points: 330

Abstract: Add an automatic closing option to DCL file opens so
   that when a procedure aborts for some reason, all files it
   opened would be closed.

Result: We currently don't plan on implementing the file
   closing feature that is requested because A) it would
   require several design changes in DCL's file processing,
   and b) an easy work around exists which will provide the
   functionality required.

   The context that is saved whenever DCL opens up a file
   contains neither the procedure level at which the file was
   opened, which is irrelevant to an opened file, or the
   logical name associated with the file.

   As a work around, you can specify exit handlers that are
   activated by either by an error condition or control-Y, by
   using the ON command. By use of the "/NOLOG" qualifier on
   the CLOSE command, you can suppress the error message
   generated by trying to close a file that wasn't yet opened
   at the time of the error.

SIR: F85-36   Position: 10   Points: 326

Abstract: Provide delivery tracking for VMS MAIL, including
   support in a network configuration.

Response: Receiving the "MAIL>" prompt back is actually
   instantaneous notification that the message is in the mail
   file of the receiver. If MRGATE (part of the ALL-IN-1
   product) is being used, say to get VMS MAIL to an ALL-IN-1
   system then return of the "MAIL>" prompt means that the
   message is in control of the Message Router for delivery.

   The ability to inform a sender when a message is actually
   read is a very contentious issue since many individuals
   would consider this to be an unacceptable invasion of
   privacy. Thus, there are no plans to implement such a
   feature in VMS MAIL.

   DIGITAL is currently investigating producing an enhanced
   VMS MAIL product that has directory services and an
   interface directly into the Message Router for store and

forward capability. Also, the Message Router will probably
have a message trace capability in the future.

FALL 1985 SIR BALLOT RESULTS

Gary Grebus
SIR Coordinator

Those of you who managed to find the Fall SIR Ballot in the
September Pageswapper were hopefully pleased with its contents.
I think it contained a very interesting collection of requests
involving many substantial issues. However, it appears that the
confusion and overwhelming size of the first combined newsletter
may have prevented some SIG members from finding the ballot. In
fact, the largest distribution of the SIR ballot yielded the
POOREST return ever, with only 172 ballots returned. The
summary of this voting appears below. Digital's response to the
top 10 requests overall will be presented at the Fall 1985 DECUS
Symposium in Anaheim.

Interpreting the SIR Ballot Results

The results of the System Improvement Request ballot are show on
the following pages. All of the reports have the same one page
format. Following the report title is the number of ballots
counted for that report. The number shown on the "All Users"
report is the total number of ballots which were returned.

The SIR's are listed on the page in order of points received,
from highest to lowest. The entry for each SIR begins with the
SIR number (from the ballot), a brief description, and the total
number of points received by that SIR. Next are listed the
number of ballots which assigned positive and negative points
points to the SIR. These numbers are expressed as a percentage
of the total number of ballots represented on the report.
Finally, the mean number of points assigned and the standard
deviation of the points are show.

THE TOP 45 SIR's AS RANKED BY ALL USERS

Total ballots in this category: 172   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	847	59.9	0.0	8.22	2.7078
55	Enhance AUTHORIZE SHOW command	588	48.8	0.0	7.00	2.8999
6	Multiple versions of layered products	494	41.9	0.6	6.77	3.6573
27	Enhance DELETE command behavior	477	39.0	1.7	6.81	3.5684
42	Enhance network support in BACKUP	471	33.1	0.0	8.26	2.3109
4	Enhance CPU usage reports in MONITOR	375	35.5	0.0	6.15	3.1190
49	Provide screen editor for AUTHORIZE	371	33.1	0.0	6.51	2.7654
18	Add class-based scheduler to VMS	364	30.8	1.2	6.62	4.1522
24	Add automatic file close for DCL	330	31.4	0.6	6.00	3.3884
36	Add delivery tracking to MAIL	326	29.1	0.6	6.39	3.4472
34	Add match limit to SEARCH	325	30.2	0.0	6.25	3.3599
10	Allow INSTALL with priority and UIC	322	29.1	0.6	6.31	3.6742
28	Modify behavior of PURGE command	297	33.7	2.3	4.79	3.6581
67	Enhance EDT search command	297	24.4	0.6	6.91	3.9989
46	Add some DCL to S/A BACKUP environ	296	25.0	0.0	6.88	2.8885
8	Provide a SYSGEN DISCONNECT	296	26.2	0.0	6.58	2.9961
17	Support standard print-file format	286	27.3	0.0	6.09	2.9769
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	286	27.9	0.0	5.96	3.0032
19	Provide for extended error info	266	26.7	0.6	5.66	3.9081
20	Add keyword search to HELP	252	26.2	0.6	5.48	3.8629
45	Provide time estimates for BACKUP	249	22.7	0.0	6.38	3.2169
33	Utility for setting file attributes	239	20.9	0.0	6.64	2.8998
37	Add wildcard send to MAIL	238	26.2	1.7	4.96	3.5667
5	Enhance the ALLOCATE command	232	21.5	0.0	6.27	2.9781
2	Provide a quota on buffered I/O RATE	230	19.2	1.7	6.39	4.0798
48	Add manual recover mode for BACKUP	223	19.8	0.0	6.56	2.9969
30	Modify treatment of SET VERIFY	221	22.7	1.7	5.26	4.6068
41	Add field support to DCL READ	220	21.5	0.6	5.79	3.9192
56	Log one process stopping another	208	21.5	0.0	5.62	2.8901
38	Provide callable interface to MAIL	201	18.6	0.0	6.28	2.6789
44	Volume init parameters to S/A BACKUP	192	20.9	0.6	5.19	2.9518
62	Support descending keys in RMS	184	15.7	0.0	6.81	3.3286
21	Add /NOIMAGE debugging option to DCL	174	17.4	0.6	5.61	4.1607
15	Provide TCP/IP support	172	12.8	0.6	7.48	3.9413
59	Support printers on term servers	170	14.0	0.0	7.08	3.0205
14	Alter the priority boost mechanism	164	16.9	1.2	5.29	3.6805
57	Support print/batch job dependency	163	14.0	0.0	6.79	2.9632
63	Provide VMS definitions for all langs.	162	12.2	0.6	7.36	3.7613
40	Better SET TERM/INQUIRE with VT220	161	16.9	0.6	5.37	4.4138
22	Allow DCL procedures in libraries	161	14.0	1.2	6.19	4.4544
39	Add /NOADVANCE to DCL WRITE	160	17.4	0.6	5.16	4.0996
60	Add tape AVR and label security	158	11.6	0.0	7.90	2.9182
47	Fix S/A BACKUP on write-protected disk	154	14.0	0.6	6.16	3.6134
23	Provide a DCL optimizer	146	16.3	2.3	4.56	5.6622
74	Add DECnet End-to-end encryption	129	9.9	0.0	7.59	2.6939

THE TOP 45 SIR's AS RANKED BY BUSINESS EDP USERS

Total ballots in this category: 53   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	297	64.2	0.0	8.74	2.4533
55	Enhance AUTHORIZE SHOW command	159	43.4	0.0	6.91	3.0288
42	Enhance network support in BACKUP	157	34.0	0.0	8.72	2.0809
6	Multiple versions of layered products	155	37.7	0.0	7.75	2.8631
36	Add delivery tracking to MAIL	143	35.8	0.0	7.53	2.9130
27	Enhance DELETE command behavior	137	34.0	1.9	7.21	3.7502
34	Add match limit to SEARCH	133	39.6	0.0	6.33	3.3665
4	Enhance CPU usage reports in MONITOR	133	41.5	0.0	6.05	3.3163
18	Add class-based scheduler to VMS	132	34.0	0.0	7.33	2.7865
49	Provide screen editor for AUTHORIZE	123	32.1	0.0	7.24	2.6108
67	Enhance EDT search command	114	24.5	0.0	8.77	2.4205
24	Add automatic file close for DCL	109	32.1	0.0	6.41	3.3365
56	Log one process stopping another	104	28.3	0.0	6.93	3.2834
46	Add some DCL to S/A BACKUP environ	104	24.5	0.0	8.00	2.7689
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	102	26.4	0.0	7.29	3.3381
57	Support print/batch job dependency	101	28.3	0.0	6.73	2.8900
30	Modify treatment of SET VERIFY	92	24.5	0.0	7.08	3.4269
20	Add keyword search to HELP	84	28.3	0.0	5.60	3.0190
28	Modify behavior of PURGE command	83	30.2	1.9	4.88	3.4074
41	Add field support to DCL READ	81	24.5	0.0	6.23	2.7127
40	Better SET TERM/INQUIRE with VT220	78	24.5	1.9	5.57	5.6120
59	Support printers on term servers	77	18.9	0.0	7.70	2.0930
33	Utility for setting file attributes	77	17.0	0.0	8.56	2.9627
5	Enhance the ALLOCATE command	75	22.6	0.0	6.25	2.6671
62	Support descending keys in RMS	74	20.8	0.0	6.73	3.2586
38	Provide callable interface to MAIL	73	17.0	0.0	8.11	2.0883
48	Add manual recover mode for BACKUP	72	15.1	0.0	9.00	2.1381
1	Retain control of a MOUNT'ed device	69	18.9	0.0	6.90	3.0714
44	Volume init parameters to S/A BACKUP	69	20.8	0.0	6.27	2.9357
39	Add /NOADVANCE to DCL WRITE	68	20.8	0.0	6.18	3.3412
2	Provide a quota on buffered I/O RATE	64	17.0	0.0	7.11	2.8916
66	Editors must avoid simultaneous update	64	18.9	0.0	6.40	2.7162
19	Provide for extended error info	64	26.4	0.0	4.57	2.7376
21	Add /NOIMAGE debugging option to DCL	63	18.9	0.0	6.30	2.6687
8	Provide a SYSGEN DISCONNECT	59	18.9	0.0	5.90	3.0714
47	Fix S/A BACKUP on write-protected disk	55	15.1	0.0	6.88	3.4821
63	Provide VMS definitions for all langs.	55	13.2	0.0	7.86	2.8536
10	Allow INSTALL with priority and UIC	53	17.0	0.0	5.89	2.2575
53	Support restricted DCL environments	52	11.3	0.0	8.67	2.4221
60	Add tape AVR and label security	52	11.3	0.0	8.67	2.1602
72	Allow image-controlled file access	51	13.2	0.0	7.29	3.1472
75	Improve node authentication in DECnet	50	9.4	0.0	10.00	0.0000
37	Add wildcard send to MAIL	47	15.1	1.9	5.22	4.3525
45	Provide time estimates for BACKUP	46	13.2	0.0	6.57	3.3594
7	Support "keyboard" filters	46	13.2	0.0	6.57	3.4572

THE TOP 45 SIR's AS RANKED BY SOFTWARE DEVELOPERS

Total ballots in this category: 128   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	629	60.2	0.0	8.17	2.8070
55	Enhance AUTHORIZE SHOW command	414	46.1	0.0	7.02	2.9507
6	Multiple versions of layered products	369	42.2	0.8	6.71	3.8522
42	Enhance network support in BACKUP	354	32.0	0.0	8.63	2.0827
18	Add class-based scheduler to VMS	315	33.6	0.8	7.16	3.7038
27	Enhance DELETE command behavior	307	32.8	2.3	6.82	3.8981
4	Enhance CPU usage reports in MONITOR	298	37.5	0.0	6.21	3.1553
49	Provide screen editor for AUTHORIZE	282	33.6	0.0	6.56	2.8056
10	Allow INSTALL with priority and UIC	267	32.0	0.8	6.36	3.7923
36	Add delivery tracking to MAIL	258	30.5	0.8	6.45	3.5370
24	Add automatic file close for DCL	247	30.5	0.0	6.33	3.0464
34	Add match limit to SEARCH	243	29.7	0.0	6.39	3.3735
8	Provide a SYSGEN DISCONNECT	223	25.0	0.0	6.97	3.0531
46	Add some DCL to S/A BACKUP environ	222	24.2	0.0	7.16	3.0011
67	Enhance EDT search command	222	25.8	0.8	6.53	4.2013
17	Support standard print-file format	215	27.3	0.0	6.14	3.1168
28	Modify behavior of PURGE command	209	33.6	3.1	4.45	3.8550
45	Provide time estimates for BACKUP	204	24.2	0.0	6.58	3.3144
33	Utility for setting file attributes	199	23.4	0.0	6.63	2.9998
30	Modify treatment of SET VERIFY	194	25.8	0.8	5.71	3.6807
19	Provide for extended error info	180	24.2	0.8	5.63	4.2786
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	180	25.8	0.0	5.45	2.9590
37	Add wildcard send to MAIL	178	26.6	1.6	4.94	3.4054
20	Add keyword search to HELP	174	24.2	0.8	5.44	4.1963
2	Provide a quota on buffered I/O RATE	174	19.5	2.3	6.21	4.3662
62	Support descending keys in RMS	162	17.2	0.0	7.36	3.1705
5	Enhance the ALLOCATE command	158	19.5	0.0	6.32	3.1454
41	Add field support to DCL READ	157	21.1	0.8	5.61	4.3148
48	Add manual recover mode for BACKUP	156	18.0	0.0	6.78	2.8755
38	Provide callable interface to MAIL	155	18.0	0.0	6.74	2.5975
21	Add /NOIMAGE debugging option to DCL	141	18.0	0.8	5.88	4.5235
56	Log one process stopping another	135	18.0	0.0	5.87	2.9435
63	Provide VMS definitions for all langs.	133	13.3	0.8	7.39	3.9577
44	Volume init parameters to S/A BACKUP	131	18.8	0.8	5.24	3.1791
47	Fix S/A BACKUP on write-protected disk	131	15.6	0.0	6.55	2.7810
23	Provide a DCL optimizer	127	16.4	1.6	5.52	4.9715
14	Alter the priority boost mechanism	121	16.4	1.6	5.26	4.1584
15	Provide TCP/IP support	110	11.7	0.8	6.88	4.3340
57	Support print/batch job dependency	109	12.5	0.0	6.81	3.1669
59	Support printers on term servers	106	13.3	0.0	6.24	3.0317
35	DIFFERENCE should support C comments	104	12.5	0.0	6.50	3.3665
7	Support "keyboard" filters	102	14.1	0.8	5.37	3.1307
40	Better SET TERM/INQUIRE with VT220	102	14.1	0.8	5.37	5.1770
61	Support segmented keys of mixed type	100	10.2	0.0	7.69	3.1460
66	Editors must avoid simultaneous update	97	15.6	2.3	4.22	4.5821

VAX-47

THE TOP 45 SIR's AS RANKED BY EDUCATIONAL USERS

Total ballots in this category: 22   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	122	63.6	0.0	8.71	2.0913
55	Enhance AUTHORIZE SHOW command	80	50.0	0.0	7.27	3.0361
6	Multiple versions of layered products	78	50.0	0.0	7.09	3.5342
19	Provide for extended error info	74	54.5	0.0	6.17	3.3257
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	72	50.0	0.0	6.55	2.8058
27	Enhance DELETE command behavior	64	45.5	4.5	5.82	4.9763
15	Provide TCP/IP support	54	31.8	0.0	7.71	3.9461
10	Allow INSTALL with priority and UIC	51	36.4	0.0	6.38	3.9256
24	Add automatic file close for DCL	50	45.5	0.0	5.00	2.7487
2	Provide a quota on buffered I/O RATE	49	31.8	0.0	7.00	3.7859
28	Modify behavior of PURGE command	49	45.5	0.0	4.90	2.7264
44	Volume init parameters to S/A BACKUP	48	31.8	0.0	6.86	3.0237
23	Provide a DCL optimizer	47	36.4	4.5	5.22	5.9535
49	Provide screen editor for AUTHORIZE	46	27.3	0.0	7.67	3.0111
42	Enhance network support in BACKUP	43	22.7	0.0	8.60	2.1909
14	Alter the priority boost mechanism	41	31.8	0.0	5.86	2.5820
41	Add field support to DCL READ	40	27.3	0.0	6.67	2.5728
39	Add /NOADVANCE to DCL WRITE	39	31.8	0.0	5.57	2.5728
22	Allow DCL procedures in libraries	38	22.7	4.5	6.33	4.7610
17	Support standard print-file format	38	22.7	0.0	7.60	3.3615
20	Add keyword search to HELP	37	36.4	0.0	4.63	2.4458
36	Add delivery tracking to MAIL	37	27.3	4.5	5.29	4.8892
46	Add some DCL to S/A BACKUP environ	37	27.3	0.0	6.17	3.1885
9	Propagate file ERASE attribute	36	27.3	0.0	6.00	3.2249
8	Provide a SYSGEN DISCONNECT	34	22.7	0.0	6.80	2.9496
53	Support restricted DCL environments	31	18.2	0.0	7.75	3.3040
48	Add manual recover mode for BACKUP	31	27.3	0.0	5.17	3.3116
34	Add match limit to SEARCH	30	27.3	0.0	5.00	2.6077
4	Enhance CPU usage reports in MONITOR	30	36.4	0.0	3.75	1.5811
16	Add FILE_ID to $GETQUI	28	22.7	0.0	5.60	3.8471
56	Log one process stopping another	25	27.3	0.0	4.17	1.3292
37	Add wildcard send to MAIL	24	22.7	4.5	4.00	5.5857
59	Support printers on term servers	24	13.6	0.0	8.00	3.4641
63	Provide VMS definitions for all langs.	24	13.6	0.0	8.00	3.4641
54	"Unbundle" the CAPTIVE login flag	23	13.6	4.5	5.75	5.6789
75	Improve node authentication in DECnet	23	13.6	0.0	7.67	4.0415
30	Modify treatment of SET VERIFY	22	18.2	0.0	5.50	3.3166
7	Support "keyboard" filters	21	22.7	4.5	3.50	3.8341
43	Enhance ACCOUNTING summary report	21	18.2	0.0	5.25	3.5000
60	Add tape AVR and label security	20	9.1	0.0	10.00	0.0000
67	Enhance EDT search command	18	9.1	0.0	9.00	1.4142
35	DIFFERENCE should support C comments	16	13.6	0.0	5.33	4.0415
40	Better SET TERM/INQUIRE with VT220	15	13.6	0.0	5.00	3.0000
32	Increase LIB$GET_FOREIGN buffer	15	9.1	0.0	7.50	3.5355
68	Keep EDT select region active	14	13.6	0.0	4.67	2.8868

VAX-48

THE TOP 45 SIR's AS RANKED BY COMPUTER SCI. RESEARCHERS

Total ballots in this category: 12   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	60	58.3	0.0	8.57	2.4398
6	Multiple versions of layered products	48	50.0	0.0	8.00	2.4495
19	Provide for extended error info	47	50.0	0.0	7.83	3.3714
8	Provide a SYSGEN DISCONNECT	45	41.7	0.0	9.00	2.2361
2	Provide a quota on buffered I/O RATE	40	33.3	0.0	10.00	0.0000
10	Allow INSTALL with priority and UIC	35	33.3	0.0	8.75	2.5000
63	Provide VMS definitions for all langs.	34	33.3	0.0	8.50	3.0000
27	Enhance DELETE command behavior	33	33.3	0.0	8.25	3.5000
4	Enhance CPU usage reports in MONITOR	32	41.7	0.0	6.40	2.5100
36	Add delivery tracking to MAIL	31	33.3	0.0	7.75	2.6300
42	Enhance network support in BACKUP	30	33.3	0.0	7.50	2.8868
55	Enhance AUTHORIZE SHOW command	24	33.3	0.0	6.00	2.7080
54	"Unbundle" the CAPTIVE login flag	24	25.0	0.0	8.00	3.4641
21	Add /NOIMAGE debugging option to DCL	23	33.3	0.0	5.75	2.9861
28	Modify behavior of PURGE command	20	25.0	0.0	6.67	2.8868
23	Provide a DCL optimizer	20	16.7	0.0	10.00	0.0000
14	Alter the priority boost mechanism	20	16.7	0.0	10.00	0.0000
7	Support "keyboard" filters	18	25.0	0.0	6.00	4.0000
15	Provide TCP/IP support	18	25.0	8.3	4.50	7.1414
66	Editors must avoid simultaneous update	16	16.7	0.0	8.00	2.8284
17	Support standard print-file format	15	16.7	0.0	7.50	3.5355
33	Utility for setting file attributes	15	16.7	0.0	7.50	3.5355
44	Volume init parameters to S/A BACKUP	15	16.7	0.0	7.50	3.5355
46	Add some DCL to S/A BACKUP environ	15	16.7	0.0	7.50	3.5355
18	Add class-based scheduler to VMS	14	16.7	0.0	7.00	4.2426
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	14	16.7	0.0	7.00	2.8284
75	Improve node authentication in DECnet	13	16.7	0.0	6.50	4.9497
67	Enhance EDT search command	11	16.7	0.0	5.50	0.7071
58	Enhance handling of TAB's in SORT	10	16.7	0.0	5.00	0.0000
59	Support printers on term servers	10	8.3	0.0	10.00	0.0000
60	Add tape AVR and label security	10	8.3	0.0	10.00	0.0000
61	Support segmented keys of mixed type	10	8.3	0.0	10.00	0.0000
62	Support descending keys in RMS	10	8.3	0.0	10.00	0.0000
49	Provide screen editor for AUTHORIZE	10	16.7	0.0	5.00	0.0000
45	Provide time estimates for BACKUP	10	8.3	0.0	10.00	0.0000
34	Add match limit to SEARCH	10	8.3	0.0	10.00	0.0000
70	Unnumbered FORMAT statements in FORTRAN	10	8.3	0.0	10.00	0.0000
74	Add DECnet End-to-end encryption	10	8.3	0.0	10.00	0.0000
47	Fix S/A BACKUP on write-protected disk	10	8.3	0.0	10.00	0.0000
41	Add field support to DCL READ	9	16.7	0.0	4.50	0.7071
37	Add wildcard send to MAIL	8	16.7	0.0	4.00	1.4142
38	Provide callable interface to MAIL	8	16.7	0.0	4.00	1.4142
39	Add /NOADVANCE to DCL WRITE	8	8.3	0.0	8.00	0.0000
35	DIFFERENCE should support C comments	8	16.7	0.0	4.00	1.4142
5	Enhance the ALLOCATE command	7	16.7	0.0	3.50	0.7071

THE TOP 45 SIR's AS RANKED BY DATA ACQ./CTRL USERS

Total ballots in this category: 44   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	243	63.6	0.0	8.68	2.4045
55	Enhance AUTHORIZE SHOW command	150	52.3	0.0	6.52	3.1023
42	Enhance network support in BACKUP	146	40.9	0.0	8.11	1.9063
49	Provide screen editor for AUTHORIZE	109	36.4	0.0	6.81	2.3443
27	Enhance DELETE command behavior	101	34.1	2.3	6.31	3.5538
8	Provide a SYSGEN DISCONNECT	96	27.3	0.0	8.00	2.4863
18	Add class-based scheduler to VMS	96	31.8	0.0	6.86	3.7181
10	Allow INSTALL with priority and UIC	96	36.4	0.0	6.00	2.7809
28	Modify behavior of PURGE command	93	43.2	4.5	4.43	3.4579
46	Add some DCL to S/A BACKUP environ	90	27.3	0.0	7.50	2.6112
2	Provide a quota on buffered I/O RATE	86	29.5	2.3	6.14	3.9973
34	Add match limit to SEARCH	84	31.8	0.0	6.00	3.5734
19	Provide for extended error info	82	31.8	0.0	5.86	3.3936
6	Multiple versions of layered products	82	36.4	2.3	4.82	4.7201
59	Support printers on term servers	71	22.7	0.0	7.10	2.9981
45	Provide time estimates for BACKUP	70	22.7	0.0	7.00	3.6515
24	Add automatic file close for DCL	68	22.7	0.0	6.80	3.5214
4	Enhance CPU usage reports in MONITOR	67	27.3	0.0	5.58	3.0588
36	Add delivery tracking to MAIL	66	22.7	0.0	6.60	3.8355
21	Add /NOIMAGE debugging option to DCL	65	25.0	0.0	5.91	2.8794
33	Utility for setting file attributes	65	25.0	0.0	5.91	2.5867
38	Provide callable interface to MAIL	64	20.5	0.0	7.11	2.5712
61	Support segmented keys of mixed type	63	15.9	0.0	9.00	2.6458
44	Volume init parameters to S/A BACKUP	62	31.8	2.3	4.13	3.1593
62	Support descending keys in RMS	61	18.2	0.0	7.63	3.6621
63	Provide VMS definitions for all langs.	58	13.6	0.0	9.67	0.8165
48	Add manual recover mode for BACKUP	57	20.5	0.0	6.33	2.7839
17	Support standard print-file format	56	20.5	0.0	6.22	2.9907
67	Enhance EDT search command	55	20.5	0.0	6.11	2.9768
66	Editors must avoid simultaneous update	49	18.2	4.5	4.90	4.3321
14	Alter the priority boost mechanism	48	20.5	4.5	4.36	4.7597
15	Provide TCP/IP support	47	15.9	2.3	5.88	5.5404
47	Fix S/A BACKUP on write-protected disk	46	13.6	0.0	7.67	2.3381
26	Provide a "catch symbol" for DCL	46	15.9	0.0	6.57	3.4572
20	Add keyword search to HELP	45	20.5	0.0	5.00	3.9370
74	Add DECnet End-to-end encryption	45	13.6	0.0	7.50	2.7386
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	41	20.5	0.0	4.56	3.3953
57	Support print/batch job dependency	40	15.9	0.0	5.71	3.4983
41	Add field support to DCL READ	36	13.6	0.0	6.00	3.2249
23	Provide a DCL optimizer	36	15.9	2.3	4.50	4.3095
30	Modify treatment of SET VERIFY	36	15.9	0.0	5.14	2.8536
65	Enhance MACRO support in DEBUG	35	15.9	0.0	5.00	2.9439
53	Support restricted DCL environments	35	11.4	0.0	7.00	2.9155
75	Improve node authentication in DECnet	35	9.1	0.0	8.75	2.5000
32	Increase LIB$GET_FOREIGN buffer	34	11.4	0.0	6.80	2.7749

THE TOP 45 SIR's AS RANKED BY CAD/CAM USERS

Total ballots in this category: 28   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
6	Multiple versions of layered products	111	53.6	0.0	7.40	3.6410
11	Support in-place disk compression	111	50.0	0.0	7.93	3.3619
42	Enhance network support in BACKUP	103	42.9	0.0	8.58	1.9752
55	Enhance AUTHORIZE SHOW command	99	50.0	0.0	7.07	2.9733
28	Modify behavior of PURGE command	73	42.9	0.0	6.08	3.6794
67	Enhance EDT search command	65	35.7	3.6	5.91	6.0738
4	Enhance CPU usage reports in MONITOR	64	35.7	0.0	6.40	3.6271
36	Add delivery tracking to MAIL	51	42.9	0.0	5.08	3.6794
27	Enhance DELETE command behavior	59	32.1	0.0	6.56	3.2830
10	Allow INSTALL with priority and UIC	58	35.7	0.0	5.80	2.6162
15	Provide TCP/IP support	58	32.1	3.6	5.80	5.0288
8	Provide a SYSGEN DISCONNECT	56	32.1	0.0	6.22	2.9907
49	Provide screen editor for AUTHORIZE	56	32.1	0.0	6.22	2.4889
18	Add class-based scheduler to VMS	53	35.7	0.0	5.30	3.3015
46	Add some DCL to S/A BACKUP environ	46	25.0	0.0	6.57	2.7603
44	Volume init parameters to S/A BACKUP	42	28.6	0.0	5.25	2.0529
38	Provide callable interface to MAIL	40	28.6	0.0	5.00	2.2039
59	Support printers on term servers	39	17.9	0.0	7.80	2.5884
48	Add manual recover mode for BACKUP	39	21.4	0.0	6.50	2.5884
40	Better SET TERM/INQUIRE with VT220	37	28.6	0.0	4.63	3.6621
63	Provide VMS definitions for all langs.	35	14.3	0.0	8.75	2.5000
34	Add match limit to SEARCH	34	28.6	0.0	4.25	3.1510
58	Enhance handling of TAB's in SORT	34	25.0	0.0	4.86	2.7343
5	Enhance the ALLOCATE command	34	21.4	0.0	5.67	2.8752
30	Modify treatment of SET VERIFY	34	25.0	3.6	4.25	4.7734
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	34	21.4	0.0	5.67	3.2042
17	Support standard print-file format	33	28.6	0.0	4.13	2.1671
20	Add keyword search to HELP	32	25.0	3.6	4.00	6.2335
24	Add automatic file close for DCL	32	28.6	0.0	4.00	2.8284
74	Add DECnet End-to-end encryption	30	14.3	0.0	7.50	2.8868
60	Add tape AVR and label security	28	10.7	0.0	9.33	1.1547
2	Provide a quota on buffered I/O RATE	27	17.9	0.0	5.40	2.5100
72	Allow image-controlled file access	27	14.3	0.0	6.75	3.3040
47	Fix S/A BACKUP on write-protected disk	27	14.3	0.0	6.75	2.7538
41	Add field support to DCL READ	25	17.9	3.6	4.17	7.8592
37	Add wildcard send to MAIL	25	21.4	0.0	4.17	3.5449
19	Provide for extended error info	24	17.9	3.6	4.00	7.8994
1	Retain control of a MOUNT'ed device	21	10.7	0.0	7.00	2.6458
14	Alter the priority boost mechanism	21	21.4	0.0	3.50	1.3784
75	Improve node authentication in DECnet	20	7.1	0.0	10.00	0.0000
33	Utility for setting file attributes	19	14.3	0.0	4.75	3.8622
56	Log one process stopping another	19	14.3	0.0	4.75	3.7749
43	Enhance ACCOUNTING summary report	18	14.3	0.0	4.50	4.0415
7	Support "keyboard" filters	18	14.3	0.0	4.50	2.5166
45	Provide time estimates for BACKUP	18	17.9	0.0	3.60	1.5166

THE TOP 45 SIR's AS RANKED BY SERVICE BUREAU OPERATORS

Total ballots in this category: 12   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	48	50.0	0.0	8.00	3.1623
67	Enhance EDT search command	46	41.7	0.0	9.20	1.7889
27	Enhance DELETE command behavior	37	50.0	0.0	6.17	3.0605
57	Support print/batch job dependency	34	41.7	0.0	6.80	3.1145
55	Enhance AUTHORIZE SHOW command	33	41.7	0.0	6.60	3.2094
37	Add wildcard send to MAIL	30	33.3	0.0	7.50	3.3166
6	Multiple versions of layered products	28	33.3	0.0	7.00	2.1602
56	Log one process stopping another	27	33.3	0.0	6.75	3.9476
42	Enhance network support in BACKUP	26	25.0	0.0	8.67	2.3094
18	Add class-based scheduler to VMS	25	33.3	0.0	6.25	2.6300
4	Enhance CPU usage reports in MONITOR	25	33.3	0.0	6.25	4.5000
34	Add match limit to SEARCH	25	33.3	0.0	6.25	2.9861
8	Provide a SYSGEN DISCONNECT	22	33.3	0.0	5.50	3.3166
46	Add some DCL to S/A BACKUP environ	22	25.0	0.0	7.33	2.5166
36	Add delivery tracking to MAIL	21	25.0	0.0	7.00	2.6458
5	Enhance the ALLOCATE command	20	25.0	0.0	6.67	3.0551
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	20	16.7	0.0	10.00	0.0000
72	Allow image-controlled file access	20	16.7	0.0	10.00	0.0000
60	Add tape AVR and label security	19	25.0	0.0	6.33	4.0415
66	Editors must avoid simultaneous update	19	33.3	0.0	4.75	1.8930
28	Modify behavior of PURGE command	18	25.0	0.0	6.00	3.6056
19	Provide for extended error info	18	33.3	0.0	4.50	0.5774
23	Provide a DCL optimizer	17	16.7	0.0	8.50	2.1213
38	Provide callable interface to MAIL	17	16.7	0.0	8.50	2.1213
22	Allow DCL procedures in libraries	16	25.0	0.0	5.33	0.5773
53	Support restricted DCL environments	16	25.0	0.0	5.33	4.1633
20	Add keyword search to HELP	16	25.0	0.0	5.33	0.5773
39	Add /NOADVANCE to DCL WRITE	15	16.7	0.0	7.50	3.5355
45	Provide time estimates for BACKUP	15	16.7	0.0	7.50	3.5355
40	Better SET TERM/INQUIRE with VT220	15	16.7	0.0	7.50	3.5355
63	Provide VMS definitions for all langs.	14	16.7	0.0	7.00	4.2426
1	Retain control of a MOUNT'ed device	14	16.7	0.0	7.00	4.2426
25	Enhance CTRL-T capability	13	16.7	0.0	6.50	4.9497
3	Provide for parsing a privilege string	12	25.0	0.0	4.00	1.0000
62	Support descending keys in RMS	12	33.3	0.0	3.00	1.4142
24	Add automatic file close for DCL	11	25.0	0.0	3.67	1.5275
9	Propagate file ERASE attribute	11	25.0	0.0	3.67	1.5275
59	Support printers on term servers	10	8.3	0.0	10.00	0.0000
12	Enhance use of swapping to page files	10	8.3	0.0	10.00	0.0000
52	Enhance SHUTDOWN procedure	10	16.7	0.0	5.00	0.0000
2	Provide a quota on buffered I/O RATE	10	8.3	0.0	10.00	0.0000
29	Improve /CONFIRM qualifier	10	8.3	0.0	10.00	0.0000
47	Fix S/A BACKUP on write-protected disk	10	8.3	0.0	10.00	0.0000
48	Add manual recover mode for BACKUP	10	8.3	0.0	10.00	0.0000
74	Add DECnet End-to-end encryption	10	8.3	0.0	10.00	0.0000

THE TOP 45 SIR's AS RANKED BY HARDWARE DEVELOPERS

Total ballots in this category: 21   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
10	Allow INSTALL with priority and UIC	81	47.6	0.0	8.10	2.4698
6	Multiple versions of layered products	75	52.4	0.0	6.82	3.5726
42	Enhance network support in BACKUP	75	42.9	0.0	8.33	2.2913
11	Support in-place disk compression	65	47.6	0.0	6.50	3.8944
46	Add some DCL to S/A BACKUP environ	64	42.9	0.0	7.11	2.8480
55	Enhance AUTHORIZE SHOW command	64	42.9	0.0	7.11	2.8480
49	Provide screen editor for AUTHORIZE	51	42.9	0.0	5.67	2.5981
18	Add class-based scheduler to VMS	48	28.6	4.8	6.86	5.4903
21	Add /NOIMAGE debugging option to DCL	46	28.6	0.0	7.67	2.6583
28	Modify behavior of PURGE command	46	42.9	0.0	5.11	2.6667
63	Provide VMS definitions for all langs.	46	23.8	0.0	9.20	1.7889
4	Enhance CPU usage reports in MONITOR	43	33.3	0.0	6.14	3.2367
24	Add automatic file close for DCL	43	33.3	0.0	6.14	2.7946
17	Support standard print-file format	41	38.1	0.0	5.13	2.4749
36	Add delivery tracking to MAIL	40	33.3	0.0	5.71	2.8115
27	Enhance DELETE command behavior	40	28.6	0.0	6.67	3.7771
14	Alter the priority boost mechanism	38	28.6	0.0	6.33	4.0332
20	Add keyword search to HELP	35	23.8	0.0	7.00	4.2426
67	Enhance EDT search command	35	28.6	0.0	5.83	3.3116
74	Add DECnet End-to-end encryption	35	19.0	0.0	8.75	2.5000
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	34	33.3	0.0	4.86	2.1931
35	DIFFERENCE should support C comments	32	28.6	0.0	5.33	3.8816
66	Editors must avoid simultaneous update	32	28.6	0.0	5.33	2.5820
38	Provide callable interface to MAIL	30	23.8	0.0	6.00	2.5495
19	Provide for extended error info	29	23.8	0.0	5.80	3.1937
37	Add wildcard send to MAIL	29	33.3	0.0	4.14	2.1931
41	Add field support to DCL READ	28	19.0	0.0	7.00	3.5590
5	Enhance the ALLOCATE command	27	19.0	0.0	6.75	4.2720
1	Retain control of a MOUNT'ed device	26	14.3	0.0	8.67	2.3094
2	Provide a quota on buffered I/O RATE	25	14.3	0.0	8.33	2.8868
59	Support printers on term servers	25	14.3	0.0	8.33	2.8868
72	Allow image-controlled file access	23	14.3	0.0	7.67	4.0415
8	Provide a SYSGEN DISCONNECT	21	23.8	0.0	4.20	3.3466
56	Log one process stopping another	20	9.5	0.0	10.00	0.0000
75	Improve node authentication in DECnet	20	9.5	0.0	10.00	0.0000
58	Enhance handling of TAB's in SORT	19	23.8	0.0	3.80	1.7889
33	Utility for setting file attributes	18	23.8	0.0	3.60	1.6733
62	Support descending keys in RMS	17	9.5	0.0	8.50	2.1213
64	Add /D_LINES support to VAX C	15	9.5	0.0	7.50	3.5355
30	Modify treatment of SET VERIFY	15	14.3	4.8	3.75	6.7515
29	Improve /CONFIRM qualifier	14	19.0	4.8	2.80	4.6043
23	Provide a DCL optimizer	14	19.0	0.0	3.50	2.3805
7	Support "keyboard" filters	13	14.3	0.0	4.33	2.5166
45	Provide time estimates for BACKUP	13	14.3	0.0	4.33	4.9329
68	Keep EDT select region active	13	9.5	0.0	6.50	4.9497

THE TOP 45 SIR's AS RANKED BY SCIENTIFIC/ENGINEERING USERS

Total ballots in this category: 99   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	472	58.6	0.0	8.14	2.7938
55	Enhance AUTHORIZE SHOW command	328	49.5	0.0	6.69	2.7478
27	Enhance DELETE command behavior	281	40.4	1.0	6.85	3.2059
42	Enhance network support in BACKUP	277	35.4	0.0	7.91	2.4177
6	Multiple versions of layered products	230	38.4	1.0	5.90	4.0183
49	Provide screen editor for AUTHORIZE	226	35.4	0.0	6.46	2.6717
17	Support standard print-file format	209	33.3	0.0	6.33	2.9333
24	Add automatic file close for DCL	207	31.3	0.0	6.68	3.0701
8	Provide a SYSGEN DISCONNECT	203	31.3	0.0	6.55	2.9647
28	Modify behavior of PURGE command	198	38.4	2.0	4.95	3.1619
18	Add class-based scheduler to VMS	198	30.3	2.0	6.19	4.9019
46	Add some DCL to S/A BACKUP environ	196	27.3	0.0	7.26	2.6398
10	Allow INSTALL with priority and UIC	193	31.3	1.0	6.03	3.8895
45	Provide time estimates for BACKUP	189	27.3	0.0	7.00	3.0000
4	Enhance CPU usage reports in MONITOR	186	30.3	0.0	6.20	3.0332
34	Add match limit to SEARCH	161	26.3	0.0	6.19	3.4412
19	Provide for extended error info	159	26.3	1.0	5.89	4.4836
48	Add manual recover mode for BACKUP	149	23.2	0.0	6.48	2.9675
33	Utility for setting file attributes	147	25.3	0.0	5.88	2.6032
37	Add wildcard send to MAIL	145	28.3	0.0	5.18	2.6674
20	Add keyword search to HELP	138	23.2	1.0	5.75	4.5612
5	Enhance the ALLOCATE command	137	22.2	0.0	6.23	3.0850
2	Provide a quota on buffered I/O RATE	136	21.2	3.0	5.67	4.4689
30	Modify treatment of SET VERIFY	133	24.2	2.0	5.12	4.8442
15	Provide TCP/IP support	132	18.2	1.0	6.95	4.1563
67	Enhance EDT search command	131	21.2	1.0	5.95	4.7256
36	Add delivery tracking to MAIL	125	24.2	0.0	5.21	3.2568
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	115	22.2	0.0	5.23	2.7243
44	Volume init parameters to S/A BACKUP	114	23.2	1.0	4.75	2.6744
47	Fix S/A BACKUP on write-protected disk	108	15.2	0.0	7.20	2.6780
38	Provide callable interface to MAIL	100	18.2	0.0	5.56	2.3066
41	Add field support to DCL READ	87	17.2	1.0	4.83	4.6935
63	Provide VMS definitions for all langs.	84	11.1	1.0	7.00	4.3901
35	DIFFERENCE should support C comments	83	15.2	0.0	5.53	3.5429
14	Alter the priority boost mechanism	81	18.2	2.0	4.05	3.5314
21	Add /NOIMAGE debugging option to DCL	80	16.2	1.0	4.71	4.7666
39	Add /NOADVANCE to DCL WRITE	79	16.2	1.0	4.65	4.8211
40	Better SET TERM/INQUIRE with VT220	78	16.2	0.0	4.88	2.9411
7	Support "keyboard" filters	77	14.1	0.0	5.50	3.0064
22	Allow DCL procedures in libraries	76	11.1	1.0	6.33	5.8205
60	Add tape AVR and label security	75	10.1	0.0	7.50	3.3417
62	Support descending keys in RMS	75	12.1	0.0	6.25	3.6213
9	Propagate file ERASE attribute	73	16.2	0.0	4.56	2.5025
74	Add DECnet End-to-end encryption	71	10.1	0.0	7.10	2.5144
59	Support printers on term servers	69	11.1	0.0	6.27	2.9357

THE TOP 45 SIR's AS RANKED BY OFFICE AUTOMATION USERS

Total ballots in this category: 50   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	230	58.0	0.0	7.93	2.6313
55	Enhance AUTHORIZE SHOW command	178	56.0	0.0	6.36	3.0818
42	Enhance network support in BACKUP	141	36.0	0.0	7.83	2.3825
27	Enhance DELETE command behavior	136	40.0	2.0	6.48	3.4150
6	Multiple versions of layered products	128	36.0	0.0	7.11	3.2879
49	Provide screen editor for AUTHORIZE	120	36.0	0.0	6.67	3.0098
24	Add automatic file close for DCL	111	42.0	0.0	5.29	3.0190
28	Modify behavior of PURGE command	110	42.0	2.0	5.00	3.0551
18	Add class-based scheduler to VMS	109	34.0	0.0	6.41	3.4652
67	Enhance EDT search command	101	30.0	0.0	6.73	3.2616
36	Add delivery tracking to MAIL	96	34.0	0.0	5.65	3.7239
34	Add match limit to SEARCH	93	30.0	0.0	6.20	3.7834
8	Provide a SYSGEN DISCONNECT	85	28.0	0.0	6.07	3.2691
46	Add some DCL to S/A BACKUP environ	83	26.0	0.0	6.38	2.9308
63	Provide VMS definitions for all langs.	81	22.0	0.0	7.36	2.6560
17	Support standard print-file format	81	28.0	0.0	5.79	3.0929
38	Provide callable interface to MAIL	80	24.0	0.0	6.67	2.2293
47	Fix S/A BACKUP on write-protected disk	80	24.0	0.0	6.67	3.0251
20	Add keyword search to HELP	78	26.0	0.0	6.00	3.0551
5	Enhance the ALLOCATE command	76	28.0	0.0	5.43	2.6808
48	Add manual recover mode for BACKUP	76	24.0	0.0	6.33	2.9025
40	Better SET TERM/INQUIRE with VT220	76	28.0	0.0	5.43	3.5673
33	Utility for setting file attributes	75	22.0	0.0	6.82	2.6007
37	Add wildcard send to MAIL	72	32.0	2.0	4.24	2.9692
19	Provide for extended error info	70	26.0	0.0	5.38	3.3798
41	Add field support to DCL READ	68	20.0	0.0	6.80	2.8206
44	Volume init parameters to S/A BACKUP	67	24.0	0.0	5.58	2.6443
21	Add /NOIMAGE debugging option to DCL	67	20.0	0.0	6.70	3.6833
4	Enhance CPU usage reports in MONITOR	65	24.0	0.0	5.42	3.1176
72	Allow image-controlled file access	65	16.0	0.0	8.13	2.9001
74	Add DECnet End-to-end encryption	65	16.0	0.0	8.13	2.5877
53	Support restricted DCL environments	63	18.0	0.0	7.00	3.3166
45	Provide time estimates for BACKUP	63	22.0	0.0	5.73	2.9695
15	Provide TCP/IP support	62	18.0	2.0	6.20	5.2451
10	Allow INSTALL with priority and UIC	61	20.0	0.0	6.10	3.2472
56	Log one process stopping another	60	22.0	0.0	5.45	3.4165
59	Support printers on term servers	60	14.0	0.0	8.57	2.4398
35	DIFFERENCE should support C comments	55	14.0	0.0	7.86	3.6710
62	Support descending keys in RMS	55	18.0	0.0	6.11	3.2575
57	Support print/batch job dependency	55	18.0	0.0	6.11	3.0596
7	Support "keyboard" filters	54	18.0	0.0	6.00	3.5355
1	Retain control of a MOUNT'ed device	53	16.0	0.0	6.63	3.2923
75	Improve node authentication in DECnet	53	12.0	0.0	8.83	2.8577
22	Allow DCL procedures in libraries	52	16.0	0.0	6.50	2.7255
30	Modify treatment of SET VERIFY	49	22.0	0.0	4.45	2.5442

THE TOP 45 SIR's AS RANKED BY TELECOMMUNICATIONS USERS

Total ballots in this category: 23   Fall 1985 Ballot

SIR Nr.	SIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
11	Support in-place disk compression	84	47.8	0.0	7.64	3.0421
6	Multiple versions of layered products	77	43.5	0.0	7.70	2.9458
42	Enhance network support in BACKUP	76	39.1	0.0	8.44	2.3511
55	Enhance AUTHORIZE SHOW command	69	47.8	0.0	6.27	3.2277
4	Enhance CPU usage reports in MONITOR	63	39.1	0.0	7.00	3.3541
8	Provide a SYSGEN DISCONNECT	59	39.1	0.0	6.56	3.4681
19	Provide for extended error info	50	34.8	0.0	6.25	3.6936
20	Add keyword search to HELP	46	30.4	0.0	6.57	3.3094
28	Modify behavior of PURGE command	45	34.8	4.3	5.00	4.0927
67	Enhance EDT search command	45	26.1	0.0	7.50	3.7283
33	Utility for setting file attributes	43	26.1	0.0	7.17	3.3714
44	Volume init parameters to S/A BACKUP	43	30.4	0.0	6.14	2.7343
10	Allow INSTALL with priority and UIC	41	30.4	0.0	5.86	2.1931
48	Add manual recover mode for BACKUP	41	26.1	0.0	6.83	3.8687
49	Provide screen editor for AUTHORIZE	40	21.7	0.0	8.00	2.7386
36	Add delivery tracking to MAIL	39	21.7	0.0	7.80	3.8987
47	Fix S/A BACKUP on write-protected disk	39	21.7	0.0	7.80	3.1937
24	Add automatic file close for DCL	39	30.4	0.0	5.57	3.3094
2	Provide a quota on buffered I/O RATE	35	21.7	4.3	5.83	5.7067
27	Enhance DELETE command behavior	35	30.4	0.0	5.00	3.1091
18	Add class-based scheduler to VMS	33	30.4	0.0	4.71	2.4976
40	Better SET TERM/INQUIRE with VT220	33	30.4	0.0	4.71	3.8607
35	DIFFERENCE should support C comments	33	17.4	0.0	8.25	3.5000
5	Enhance the ALLOCATE command	32	30.4	0.0	4.57	1.9024
37	Add wildcard send to MAIL	31	26.1	0.0	5.17	3.1252
63	Provide VMS definitions for all langs.	30	13.0	0.0	10.00	0.0000
73	Implement government classifications	30	13.0	0.0	10.00	0.0000
62	Support descending keys in RMS	28	17.4	0.0	7.00	3.5590
14	Alter the priority boost mechanism	28	21.7	0.0	5.60	2.5100
38	Provide callable interface to MAIL	27	21.7	0.0	5.40	3.6469
21	Add /NOIMAGE debugging option to DCL	27	17.4	0.0	6.75	3.7749
61	Support segmented keys of mixed type	25	13.0	0.0	8.33	2.8868
30	Modify treatment of SET VERIFY	25	17.4	0.0	6.25	4.3493
41	Add field support to DCL READ	25	13.0	0.0	8.33	2.8868
34	Add match limit to SEARCH	25	13.0	0.0	8.33	2.8868
31	SET PROTECTION for logical name tables	25	13.0	0.0	8.33	2.8868
50	Add /INTERACTIVE and /IMAGE to SHOW SYS	23	13.0	0.0	7.67	3.2146
7	Support "keyboard" filters	22	13.0	0.0	7.33	4.6188
17	Support standard print-file format	21	21.7	0.0	4.20	3.7014
39	Add /NOADVANCE to DCL WRITE	21	13.0	0.0	7.00	3.6056
60	Add tape AVR and label security	21	13.0	0.0	7.00	3.6056
22	Allow DCL procedures in libraries	21	13.0	0.0	7.00	3.6056
9	Propagate file ERASE attribute	20	17.4	0.0	5.00	1.0000
59	Support printers on term servers	20	8.7	0.0	10.00	0.0000
64	Add /D_LINES support to VAX C	19	8.7	4.3	6.33	6.3509

VAX System SIG Committee List

As of October 28, 1985

Osman K. Ahmad - TOPS-VAX
      Association of American Railroads
      Technical Center, Research and Test Department
      3140 South Federal Street
      Chicago, IL 60616

Joe Angelico - Assistant Symposium Coordinator
      US Coast Guard CCGD8(DT)
      Hale Boggs Federal Building
      500 Camp Street, New Orleans, LA 70130

Elizabeth Bailey - Volunteer Coordinator
      222 CEB
      Tennessee Valley Authority
      Muscle Shoals, AL 35660

June Baker - Planning
      Computer Sciences Corporation
      6565 Arlington Boulevard
      Falls Church, VA 22046

Joe L. Bingham - Librarian
      Mantech International
      2320 Mill Road
      Alexandria, VA 22314

Bob Boyd - Commercial
      GE Microelectronics Center
      MS 2P-04
      Post Office Box 13409
      Research Triangle Park, NC 27709

C. Douglas Brown - Security
      Sandia Labs
      Division 2644
      P.O. Box 5800
      Albuquerque, NM 87185

Jack Cundiff - Assistant Symposium Coordinator
      Horry-Georgetown
      Post Office Box 1966
      Conway, SC 29526

Tom Danforth - Handout Editor
      Woods Hole Oceanographic Institute
      Woods Hole, MA 02543

Jim Downward - Migration and Host Development, VAXintosh
      KMS Fusion Incorporated
      3941 Research Park Drive
      Ann Arbor MI 48106

Jane Furze - Campground
      3830 West Cochise
      Phoenix, AZ 85064

Dennis Frayne - Real Time/Process Control
      McDonnell Douglas
      5301 Bolsa Avenue
      Huntington Beach, CA 92646

Carl E. Friedberg - Internals
      In House Systems
      165 William Street
      New York, NY 10038

Don Golden - Publications Coordinator
      c/o Shell Development Company, D-2132
      Westhollow Research Center
      Post Office Box 13480
      Houston, TX 77001

Gary Grebus - System Improvement Request
      Battelle Columbis Labs
      Room 11-6011
      505 King Avenue
      Columbus, OH 43201-2693

B. Hancock - Network
      Dimension Data Systems, Incorporated
      2510 Limestone Lane
      Garland, TX 75040

Jeffrey S. Jalbert - Historian
      J C C
      Post Office Box 381
      Granville, OH 43023
      614-587-0157

Ken Johnson - VAXcluster Working Group
      Meridian Technology Corporation
      Post Office Box 2006
      St. Louis, MO 63011

Ray Kaplan - VAXeln
      Pivotal Incorporated
      6892 East Dorado Court
      Ticson, AZ 85715

Lawrence J. Kilgallen - Newsletter Editor
        Box 81, MIT Station
        Cambridge, MA 02139-0901

Margaret Knox - Chair
        Computation Center
        University of Texas
        Austin, Texas 78712

Ross W. Miller - Vice Chair and Working Group Coordinator
        Online Data Processing, Inc.
        N 637 Hamilton
        Spokane, WA 99202

Eugene Pal - Multiprocessor
        US Army
        CAORA (ATOR-CAT-C)
        Fort Leavenworth, KA

Thomas Provost - Hardware
        MIT/LNS Bates Linac Facility
        Post Office Box 846
        Middleton, MA 01949

Susan Rehse - System Management
        Lockheed Missiles
        3251 Hanover Street
        Palo Alto, CA 94301-1187

Bob Robbins - Advisor
        Array Computer Consultants
        5364 Woodvale Drive
        Sarasota, FL 33582

Larry Robertson - Real Time/Process Control
        Bear Computer Systems Inc.
        5651 Case Avenue
        North Hollywood, CA

David Schmidt - LUG Coordinator
        Management Sciences Associates
        5100 Centre Avenue
        Pittsburgh, PA 15232

Al Siegel - Advisor
        Battelle Memorial Institute
        505 King Avenue
        Columbus, OH 43201-2693

D. Slater - Artificial Intelligence
        Institute for Defense Analysis
        1801 North Beavregard Street
        Alexandria, VA 22314

INPUT/OUTPUT

A SIG Information Interchange

A form for INPUT/OUTPUT submissions is available at the back of
the issue.

INPUT/OUTPUT 480

Caption: VI for VMS? Wordstar for VMS?

Message: Does anyone know where I can get a VI-like editor for
          VMS without buying a layered UNIX? I would also like
          to find a Wordstar lookalike for VMS without buying a
          hardware/software solution.

Contact: Greg Collver
          Josephine County Schools
          706 NW A Street
          Grants Pass, OR 97526
          Telephone (503) 476-7721

Date: December 12, 1985

INPUT/OUTPUT 481

Caption: YACC, LEX and LINT for MicroVAX I under VMS

Message: We are looking for versions of the UNIX utilities
          YACC, LEX and LINT that will run on a DEC MicroVAX I
          under VMS 4.1 with DEC C.

Contact: W. E. Wilson
          Nuclear Radiation Center
          Washington State University
          Pullman, WA 99164-1300
          Telephone (509) 335-8317

Date: December 16, 1985

INPUT/OUTPUT 482

Caption: Program to read IBM-PC floppy disks on a MicroVAX I

Message: We desire to be able to read 5-1/4" floppy disks
            generated on an IBM-PC on the RX50 drive of a MicroVAX
            I. The July 1985 issue of the Pageswapper (I/O 431)
            indicated that Allison Hamilton in Canada had written
            a FORTRAN program that would perform this function. I
            wrote to Mr. Hamilton but did not receive a reply.

Contact: W. E. Wilson
            Nuclear Radiation Center
            Washington State University
            Pullman, WA 99164-1300
            Telephone (509) 335-8317

Date: December 16, 1985

INPUT/OUTPUT 483

Caption: Dialing out on the VAX using a DF112 Modem

Message: I wrote a Macro program which under VMS 3.7 would
            successfully dial out and connect our VAX to remote
            locations. Now under VMS 4.1 the program no longer
            works. Has the QIO function or related functions
            changed or is the problem in the terminal line set up?

Contact: Tim Barrett
            Chalet Susse International, Incorporated
            Chalet Drive
            Wilton, NH 03086
            Telephone (603) 654-2000 ext 276

Date: December 24, 1985
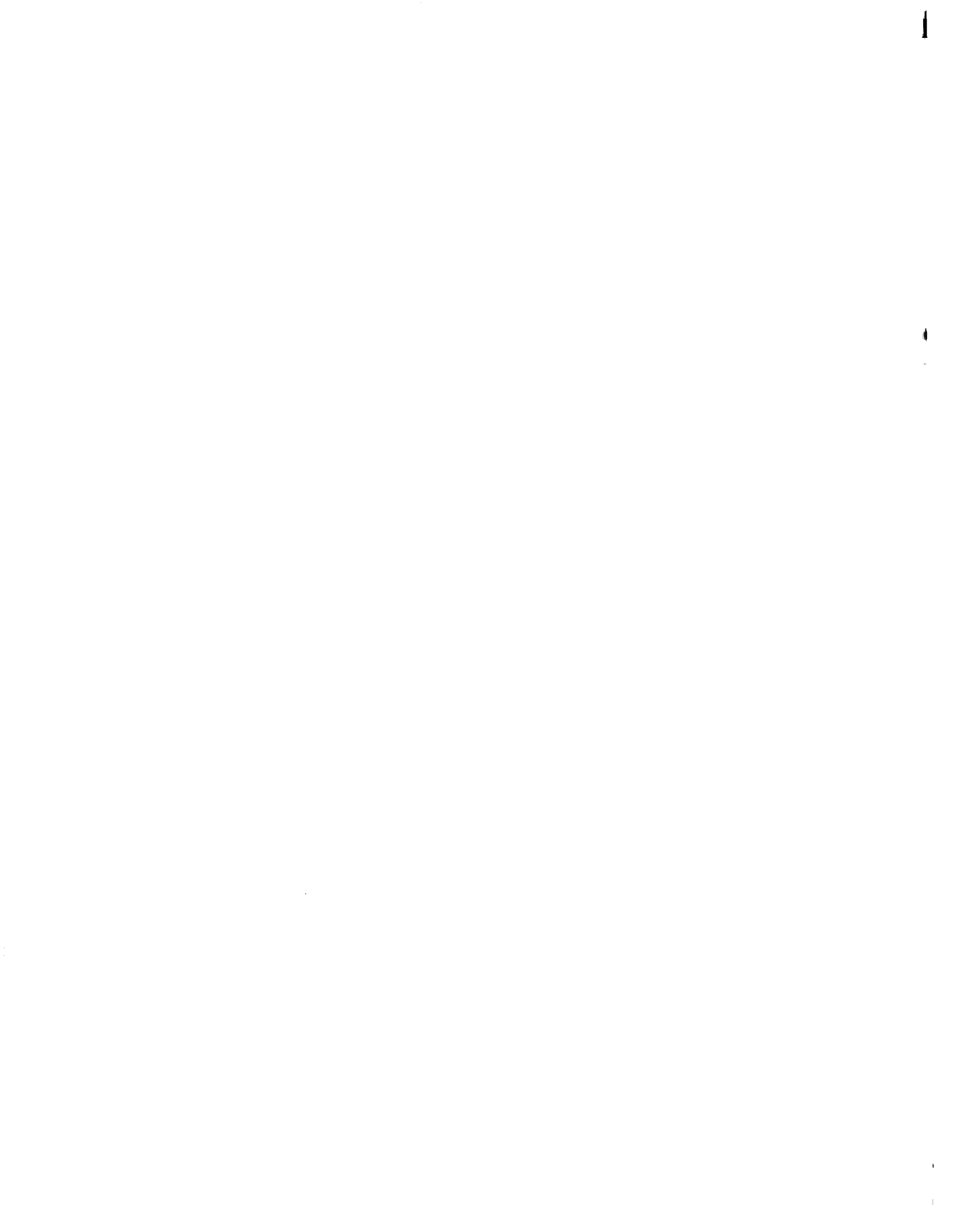
LIBRARY
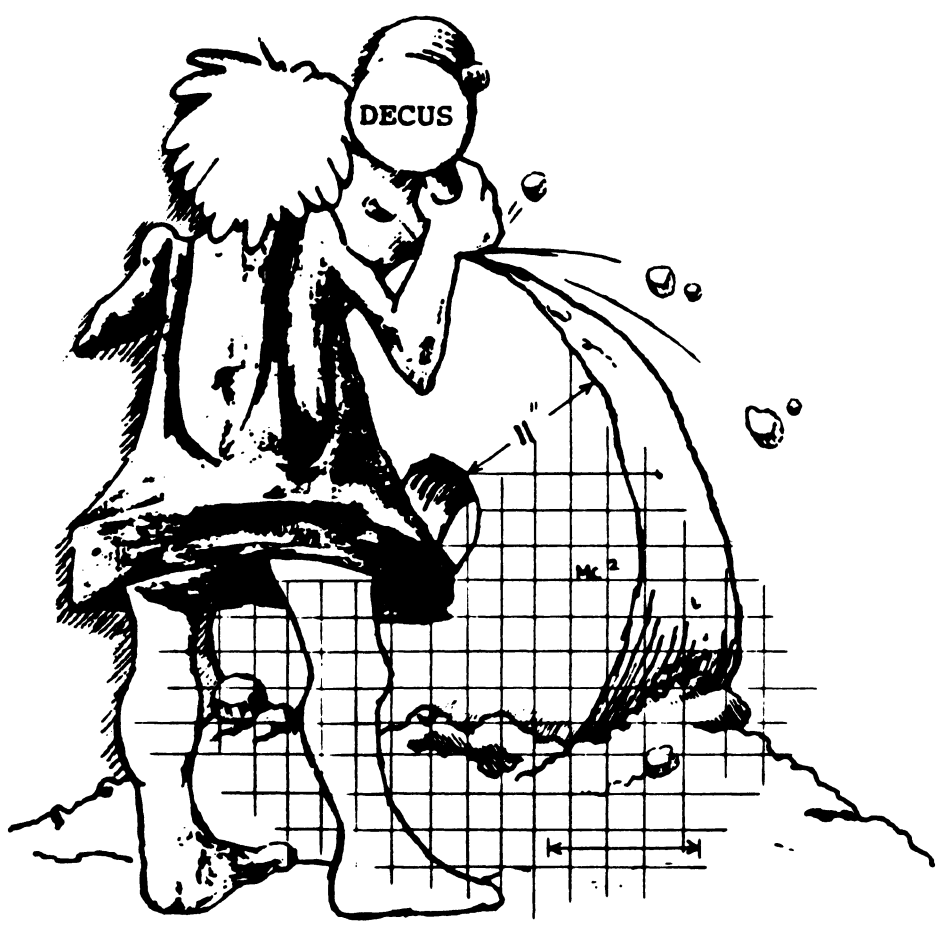
DECUS PROGRAM LIBRARY

NEW LIBRARY PROGRAMS AVAILABLE FOR THE PDP-11 COMPUTER FAMILY

DECUS ORDER NO: 11-813
Title: PEP: A Peptide Sequencing Program, Version:-June 1985

Author: Charles Hamm, National Institute of Environmental Health, Research Triangle Park, NC, Operating System: RSX-11M V4.1C, Source Language: FORTRAN 77, Memory Required: 19KW, Other Software Required: FORTRAN 77 compiler or resident FCS library

Abstract: This program is intended to help researchers find possible constructs of peptides given the mass spectrum as generated by a fast-atom bombardment (FAB) tandem mass spectrometer and the suspected composition of the peptide. The program compares all permutations of a given combination of amino acids forming a peptide to the spectrum of the actual peptide. The comparsion is made by mathematically breaking each permutation at each of its possible cleavage points and counting the number of ion fragments that have a corresponding mass in the spectrum list. Only the permutations that have the highest number of matched fragments are considered candidates for the actual peptide and are listed in an output file.

Documentation on magnetic media. Media (Service Charge Code): Floppy Diskette (KA), 600' Magtape (MA), Format: FILES-11

NEW LIBRARY PROGRAMS AVAILABLE FOR THE PDP-11 COMPUTER FAMILY

DECUS ORDER NO: 11-814
Title: LOANS, Version:V1.0, October 1985

Author: James H. Norman, US Army, White Sands Missile Range, NM, Operating System: RSX-11M V3.2, RT-11 V4.0, VAX/VMS V4.0, Source Language: FORTRAN IV Memory Required: 1450KW

Abstract: LOANS is a program to compute a repayment schedule of a loan. The user inputs the loan amount, the interest rate, and the length of the loan.
The program then computes the monthly payment, the amount applied to the principal, the amount of interest paid and the loan balance. The output is a disk file which contains a table of the Orabove values. The total interest paid on the loan is written at the end of the table. The file may be listed on a terminal or printed on a line printer.

A rounding routine is included to round each value to the nearest cent.

Documentation on magnetic media. Media (Service Charge Code): Floppy Diskette (KA), 600' Magtape (MA), Format: RT-11

DECUS ORDER NO: 11-815
Title: DPRINT, Version:V1.6, December 1984

Author: Ed Mills, Harris Semiconductor Corp., Melbourne, FL, Operating System: TSX, RT, Source Language: PASCAL, Other Software Required: OMSI-PASCAL (if re-compilation is required); RT-11 MACRO language, Special Hardware Required: LA-100 printer or LA-100 compatible; VT-100 or compatible terminal with terminal attributes.

Abstract: DPRINT is a PASCAL program written to control DEC LA-100 printers. It allows the user to enter one or more file names and set the print parameters (i.e. letter/draft quality; margins; font; etc.), as he or she desires. It is very user-friendly and performs error-checking/recovery. It was written for a DEC PDP 11/23 under the TSX operating system, although it should run under RT-11 as well. The user fills out a print menu and exits. The print control characters are seen, followed by the queued files. Lastly, the printer default parameters are reset. Control-characters within the text change the printer parameters as they normallly would when sent to a printer.

Release Notes are included with each order.

Restrictions: The software is designed to run under TSX or RT-11 only. Since the package makes MACRO-11 calls, it will not run under RSX without modifications.

Documentation on magnetic media. Media (Service Charge Code): Floppy Diskette (KA), 600' Magtape (MA), Format: RT-11

NEW LIBRARY PROGRAMS AVAILABLE FOR THE DECSYSTEM-20 FAMILY OF COMPUTERS

DECUS ORDER NO: 20-SP-10
Title: Symposium Tape from DECSYSTEM-20 SIG, Spring 1985, New Orleans, Version:Spring 1985

Author: Various, Submitted By: Steve Attaya, Wiener Enterprises, Harahan, LA, Operating System: TOPS-20 V5.1 Source Language: MACRO-10

Abstract: The TOPS-20 Symposium Tape from Spring 1985 (New Orleans) contains JKILLR, SETERM and NNFT mods from Eastman Kodak; terminal control, wide directory display, and file searching utilities from Computer Sciences Corporation; a set of MACRO macros

with sample programs and DUMCPY, a DUMPER tape copying facility, a user mode COMND% JSYS simulator for TOPS-10/20 from SOHIO Petroleum; TAPSAV, a user-mode replacement for DUMPER and WPSIM, a low-overhead, sophisticated word-processing editor from Wesleyan University; MSGDAE, a general-purpose IPCF message handler and LPTSPL patches for TTY lines from American Mathematical Society.

No guarantees are made as to the completeness, usability, or quality of the programs on this tape; and the material has not been checked or verified.

Complete sources not included. Documentation on magnetic media. Media (Service Charge Code): 2400' Magtape (PS)

REVISIONS TO LIBRARY PROGRAMS

DECUS ORDER NO: PRO-143
Title: RT on P/OS, Version: V2A, October 1985

Author: Chester Wilson, Charleville, Australia, Operating System: RT-11 V5, Source Language: MACRO-11, Other Software Required: RT-11 distribution Special Hardware Required: Professional-350

Abstract: RT on P/OS allows a PRO-350 to run RT-11 from a contiguous file on a portion of the hard disk set up for P/OS. The "DC" handler is actually a modified "DW" handler, with an ability to allow the user to specify a "device" size and offset position within the hard disk.
The distribution is provided on a DZ(RX50) disk with instructions for mating with a foreground/background or virtual memory monitor from the RT-11 distribution kit.

Documentation on magnetic media. Media (Service Charge Code): 5 1/4" Floppy Diskette (JA), Format: RT-11

DECUS ORDER NO: VAX-129
Title: FORTRAN Programming Tools, Version:VII.0, September 1985

Author: Arthur E. Ragosta, US Army, Moffett Field, CA, Operating System: VAX/VMS V4.0, Source Language: DCL, FORTRAN 77, MACRO-32, Memory Required: Varies

Abstract: The FORTRAN Programming Tools are a series of tools used to support the development and maintenance of FORTRAN source codes. Included are a debugging aid, source code maintenance aids, print utilities, a CPU time monitoring program, a NAMELIST-like package, and a library of useful, well-documented routines. These tools assist in reducing development

time and encouraging high quality programs. Although intended for FORTRAN users, some of the tools can be used on data files or other programming languages.

Release Notes are distributed with each order.

Note: Uses VMS Version 4.0 BRKTHRU System Service.

Changes and Improvements: Major bug repairs were in the BUGOUT system and in several routines in MERLIB. Major enhancements are utility to assist transfer of files across card-oriented communications systems and inclusion of uninitialized variable checking and global variable cross-reference in the BUGOUT system. All of the PASCAL programs from previous versions were replaced with FORTRAN programs to enhance coherence and efficiency of package.

Complete sources are not included. Documentation on magnetic media. Media (Service Charge Code): 600' Magtape (MA), Format: VAX/ANSI (Blocked at 2048)

DECUS PROGRAM LIBRARY CHANGES

DECUS Program Library CHANGES:
- For DECUS Order Number V-SP-3, Symposium Tape from the VAX SIG, Spring, 1980 Chicago, the Catalog lists the format as VAX/ANSI (Blocked at 2048). This is incorrect, it should read RMSBCK Format.
- For the revision to DECUS Order Number 11-SP-46, PORTACALC, please add the media, Manual (EA).

HOW TO SUBMIT TO A SPECIFIC SECTION OF THE NEWSLETTER

The following is a listing of the Newsletter Editors with their addresses and phone numbers. All submissions to the newsletter should be submitted directly to the appropriate Editor.

ARTIFICIAL INTELLIGENCE
Terry Shannon
160 State Street
Boston, MA 02109
(617) 367-7190

BUSINESS APPLICATIONS
Thomas Byrne
L Karp & Sons
1301 Estes
Elk Grove, IL 60007
(312) 593-5705

DATA MANAGEMENT SYSTEMS
Russ Poisson
Seed Software Corp.
2121 Eisenhower Avenue
Alexandria, VA 22314
(703) 783-4944

DAARC
Ellen Reilly
William H. Rorer
500 Virginia Drive
Ft. Washington, PA 19034
(215) 628-6547

GRAPHICS APPLICATION
Michael Anton
P.O. Box 591293
Houston, TX 77259-1293
(713) 928-4838

IAS
John Ross Roman
McDonnell Douglas
600 McDonnell Blvd.
Hazelwood, MO 63042
(314) 234-0984

LARGE SYSTEMS
Michael Joy
1st Church of Christ
Scientist
Boston, MA 02115
(617) 262-2300 x 3903

NETWORKS
Vicki Hancock
2510 Limestone Lane
Garland, TX 75040
(214) 495-7353

PERSONAL COMPUTER
Caroline Mack
6415 Adelphi Road
University Park, MD 20782
(301) 927-0108

RSX
Dominic DiNollo
Loral Electronics
Engineering Computer Center
Ridge Hill
Yonkers, NY 10710
(914) 968-2500 x2210

SITE MANAGEMENT & TRAINING
Gregory Brooks
Washington University
Behavior Research Lab.
1420 Gratton St.
St. Louis, MO 63104
(314) 241-7600 x257

VAX SYSTEMS
Larry Kilgallen
c/o DECUS Office
219 Boston Post Road. (BP02)
Marlboro, MA 01752

APL
Doug Bohrer
Bohrer & Company
903 Ridge Road, Suite 3
Wilmette, IL 60091
(312) 251-9449

COMMERCIAL LANGUAGES
Ted Bear
RAMTEK
2211 Lawson Lane
Santa Clara, CA 95950
(408) 988-2211

DATATRIEVE
Joe H. Gallagher
Cleveland Clinic Foundation
9500 Euclid Avenue
Cleveland, OH 44106
(216) 444-2551

EDUSIG
Fred Bell
Taft College
29 Emmons Park Drive
P.O. Box 1437
Taft, CA 93268
(805) 763-4282

HMS
William Walker
Monsanto Research Corp.
P.O. Box 32 A-152
Miamisburg, OH 45342
(513) 865-3557

LANGUAGES & TOOLS
Alan Folsom Jr.
Fischer & Porter Company
E. County Line Road
Warminster, PA 18974
(215) 674-7154

MUMPS
Janet Berryman
2405 N. Bush
Santa Ana, CA 92706
(714) 953-1025

OFFICE AUTOMATION
Margaret Drake
Univ. of TX Health Science Ctr.
7703 Floyd Curl Drive
San Antonio, TX 78284
(512) 691-6105

RSTS
Bill Hobbs
ComManD. Inc.
6535 E. 82nd St., Suite 102
Indianapolis, IN 46250
(317) 842-5320

RT
Bill Leroy
The Software House, Inc.
470 E. Paces Ferry Road Park
NE 1020
P.O. Box 52661
Atlanta, GA 30355
(404) 231-1484

UNISIG
William Toth
Harvard-Smithsonian Ctr. for
Astrophysics
60 Garden Street P353
Cambridge, MA 02138
(617) 495-7181

Bruce Bergman
UserWare International
2235 Meyer Avenue
Escondido, CA 92025-1070
(619) 741-8825

```
--------------------------------------------------------------------
            SUBMITTING ARTICLES TO THE HMS SIG NEWSLETTER
--------------------------------------------------------------------
```

The purpose of the HMS SIG Newsletter is to serve as a forum
to share information related to DEC hardware with the
members of the SIG. As such, the existence of the
newsletter is entirely dependent on your contributions. If
you have an HHK item, a better or safer way to do something,
product news, a tutorial article of general interest, etc.,
we are interested in publishing it in the newsletter. It is
intended that the HMS SIG Newsletter be published at least
four times a year.

There are several ways to submit material for the
newsletter:

   o   The Hardware Submission Form in the back of the
       newsletter can be used for brief items (there is
       not enough room if you have a lot to say).

   o   You can send me camera-ready hard-copy (this saves
       me a lot of typing).

   o   I will accept submissions on floppys. I can handle
       RX50's or 8" diskettes (either density, single or
       double sided). I prefer RT-11 format, if possible,
       but I can probably handle RSX or VMS stuff somehow.
       I will return your diskette(s), of course.

   o   Those of you that have access to DCS can send
       things to username WALKER. I check DCS daily.

   o   I am also on CompuServe as "Bill Walker 71066,24".

In any event, if you have anything to submit, send it! If
it is a mess, but I can read it, I will get it in the
newsletter somehow. Finally, if you have any question about
submitting material, call me. My telephone number is listed
below.

Contributions can be sent to:

        HMS Editor   William K. Walker
        DECUS   OR   Monsanto Research Corp.
        BP02   ==   P.O. Box 32   A-152
        249 Northboro Road   Miamisburg, OH   45342
        Marlboro, MA 01752   (513) 865-3557

If you need to get something to me quickly, send it to my
work address.

DECUS SUBSCRIPTION SERVICE
SIGs NEWSLETTERS
U.S. CHAPTER MEMBERS ONLY

As a member of DECUS U.S. Chapter, you are entitled to contribute and subscribe to the DECUS monthly publication, **SIGs Newsletters.** You also have the opportunity to subscribe to the Symposia Proceedings which are a compilation of the reports from various speakers at the U.S. National DECUS Symposia.

- **No Purchase Orders will be accepted.**
- The order form below must be used as an invoice.
- All checks must be made payable to DECUS.
- All orders MUST be paid in full.
- No refunds will be made.
- The address provided below will be used for all DECUS mailings; i.e. Membership, Subscription Service and Symposia.
- SIGs Newsletters Price is for a one-year subscription beginning the month following receipt of payment.

Name_____ DECUS Member No. _____

Company_____

Address_____

_____

City_____ State _____ Zip_____

Phone_____

Subscription Service Offering	Qty.	Unit Price	Total
SIGs Newsletters	_____	$35.00	_____
Spring '85 Proceedings (SP5) Unavailable	_____	12.00	_____
Fall '85 Proceedings (FA5)	_____	12.00	_____
Spring '86 Proceedings (SP6)	_____	12.00	_____
Fall '86 Proceedings (FA6)	_____	12.00	_____
TOTAL COST OF SUBSCRIPTION			$_____

☐ MASTERCARD ☐ VISA ☐ DINERS CLUB/CARTE BLANCHE®

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ Exp. Date_____

I understand that there will be no refunds even if I decide to cancel my subscription.

Signature: _____

FOR DIGITAL EMPLOYEES ONLY	FOR DECUS OFFICE ONLY
Badge No._____ CC: _____	Check No._____
CC Mgr. Name_____	Bank No._____
CC Mgr.Signature_____	Amount $ _____

Subscription Service, DECUS (BP02), 219 Boston Post Rd, Marlboro, MA 01752 (617) 480-3418.

![DECUS logo]
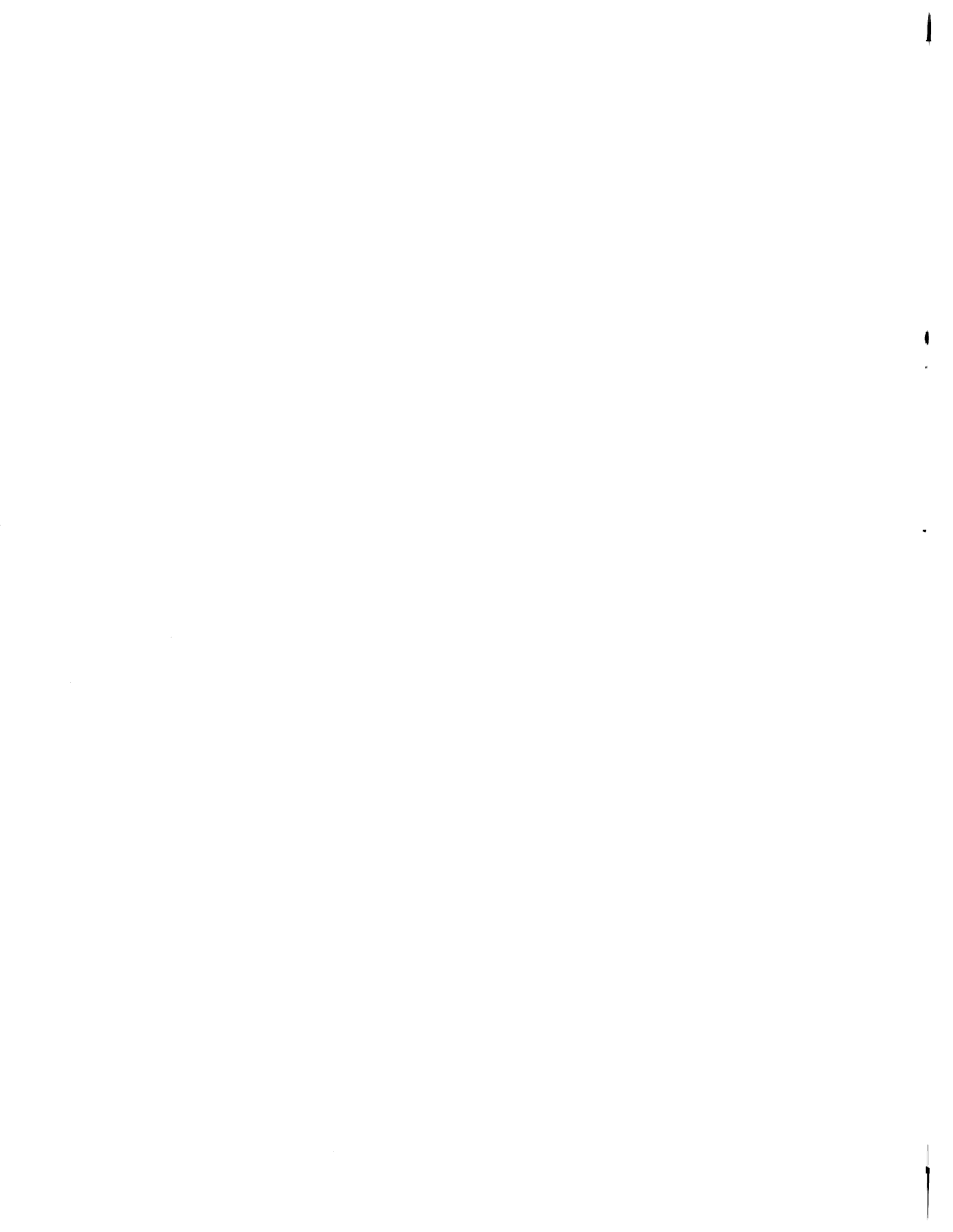
DECUS U.S.CHAPTER
APPLICATION FOR MEMBERSHIP

☐ New Membership ☐ Update to current membership profile Current DECUS Member. # ___ ___ ___ ___ ___ ___

NOTE: PLEASE PRINT CLEARLY OR TYPE!
PLEASE PROVIDE A COMPLETE MAILING ADDRESS, INCLUDE ZIP CODE IN ACCORDANCE WITH POSTAL REGULATIONS FOR YOUR LOCALITY.

ARE YOU AN EMPLOYEE OF DIGITAL EQUIPMENT CORPORATION? ☐ YES   ☐ NO

Name: _____
(first)   (Middle Intial)   (Last/Family Name)

Company: _____

Address: _____

_____

_____

City/Town: _____ State: _____ Zip: _____

_____

Telephone: Home (___) _____ Work (___) _____

HOW DID YOU LEARN ABOUT DECUS? Please check applicable item.

1 ☐ ANOTHER DECUS MEMBER	4 ☐ DIGITAL SALES	13 ☐ LOCAL USER GROUP
2 ☐ SYMPOSIA	5 ☐ HARDWARE PACKAGE	14 ☐ SPECIAL INTEREST GROUP
8 ☐ DECUS CHAPTER OFFICE	6 ☐ SOFTWARE PACKAGE	7 ☐ SOFTWARE DESPATCH
10 ☐ DIGITAL STORE	12 ☐ ADVERTISING	(DIGITAL Newsletter)

DO YOU WISH TO BE INCLUDED IN MAILINGS CONDUCTED BY DIGITAL (for Marketing purposes etc.?)   ☐ Permission
☐ Refusal

TYPE OF DIGITAL HARDWARE USED: Please check those applicable to you.

20 ☐ DECMATE	52 ☐ LSI-11	21 ☐ PROFESSIONAL	5 ☐ WPS-8
82 ☐ DECsystem-10	3 ☐ PDP-8 FAMILY	22 ☐ RAINBOW	51 ☐ WPS-11
83 ☐ DECSYSTEM-20	50 ☐ PDP-11 FAMILY	54 ☐ VAX FAMILY	

MAJOR OPERATING SYSTEMS? LANGUAGES USED: Please check those applicable to you

1 ☐ ADA	26 ☐ CORAL-66	47 ☐ FOCAL	67 ☐ OS/8	109 ☐ RT-11
2 ☐ ALGOL	28 ☐ COS	48 ☐ FORTRAN	68 ☐ PASCAL	97 ☐ TECO
5 ☐ APL	34 ☐ DATATRIEVE	51 ☐ GAMMA	72 ☐ PL-11	70 ☐ TOPS-10
7 ☐ BASIC	35 ☐ DBMS	110 ☐ IAS	92 ☐ RPG	71 ☐ TOPS-20
17 ☐ BLISS	38 ☐ DECnet	53 ☐ IQL	81 ☐ RSTS/E	104 ☐ VMS
19 ☐ C	43 ☐ DIBOL	58 ☐ MACRO	83 ☐ RSX	107 ☐ WPS-8
22 ☐ COBOL	45 ☐ DOS-11	65 ☐ MUMPS	91 ☐ RMS	

TYPE OF BUSINESS (ENVIRONMENT)/COMPUTER APPLICATIONS
Please check that which best describes your business/application

21 ☐ ACCOUNTANCY	1 ☐ EDUCATION/PRIMARY	73 ☐ NUMERICAL CONTROL
7 ☐ BANK	2 ☐ EDUCATION/SECONDARY	68 ☐ OEM-COMMERCIAL
64 ☐ BUSINESS/COMMERCIAL	61 ☐ EDUCATION-TECHNOLOGY	78 ☐ OEM-TECHNICAL
74 ☐ BUSINESS/INFORMATION SYSTEMS	3 ☐ EDUCATION/UNIVERSITY	56 ☐ PHYSICAL SCIENCES
57 ☐ CHEMISTRY	67 ☐ ENGINEERING	20 ☐ RESEARCH/DEVELOPMENT
54 ☐ CLINICAL LABORATORY	65 ☐ FINANCE/ACCOUNTING	10 ☐ RETAIL
63 ☐ COMPUTATION	77 ☐ GOVERNMENT	76 ☐ SOFTWARE DEVELOPMENT
11 ☐ CONSUMER ELECTRONICS	75 ☐ GRAPHICS	53 ☐ TELECOMMUNICATIONS
18 ☐ CONSULTANT	4 ☐ HOSPITAL	19 ☐ TELEPHONE/UTILITIES
72 ☐ DATA ACQUISITION	62 ☐ INDUSTRIAL	51 ☐ TIMESHARING
52 ☐ DATA COMMUNICATIONS	55 ☐ LABORATORY/SCIENTIFIC	80 ☐ TRAINING/INSTRUCTION
13 ☐ DATA PROCESSING SERVICES	14 ☐ LIBRARY	66 ☐ TYPESETTING/PUBLICATION
71 ☐ DATA REDUCTION	58 ☐ LIFE SCIENCES	
17 ☐ DIGITAL EMPLOYEE-ENGINEERING	70 ☐ MANUFACTURING	
15 ☐ DIGITAL EMPLOYEE-MARKETING	79 ☐ MARKETING	
16 ☐ DIGITAL EMPLOYEE-SERVICE GROUP	59 ☐ MEDICAL RESEARCH	
60 ☐ EDUCATIONAL ADMINISTRATION	6 ☐ MILITARY INSTALLATION	

SPECIAL INTEREST GROUP (SIGs) ENROLLMENT
I wish to participate in the following DECUS U.S. Chapter Special Interest Groups.

33 ☐ APL SIG	11 ☐ HARDWARE AND MICRO	36 ☐ PERSONAL COMPUTER
2 ☐ COMMERCIAL	35 ☐ IAS	18 ☐ RSTS/E
LANGUAGES	31 ☐ DAARC(LABS)	17 ☐ RSX
6 ☐ DATA MGMT.SYS.	27 ☐ LARGE SYSTEMS	19 ☐ RT-11
5 ☐ DATATRIEVE	16 ☐ LANG. AND TOOLS	32 ☐ SITE MGMT.& TRNG
7 ☐ BUSINESS APPL.	14 ☐ MUMPS	21 ☐ UNISIG
8 ☐ EDUSIG	15 ☐ NETWORKS	26 ☐ VAX SYSTEMS
10 ☐ GRAPHICS APPL	34 ☐ OFFICE AUTOMATION	

JOB TITLE/POSITION - Please check:

1 ☐ CORPORATE STAFF	101 ☐ CORPORATE DIRECTOR OF DP/MIS
2 ☐ DIVISION OR DEPARTMENT STAFF	102 ☐ ADMINISTRATIVE ASSISTANT
3 ☐ SYSTEMS ANALYSIS	103 ☐ TECHNICAL ASSISTANT
4 ☐ APPLICATIONS PROGRAMMING	104 ☐ SERVICES COORDINATOR
5 ☐ SYSTEMS ANALYSIS/PROGRAMMING	105 ☐ MANAGER
6 ☐ OPERATING SYSTEM PROGRAMMING	106 ☐ ANALYST
7 ☐ DATABASE ADMINISTRATION	107 ☐ PROGRAMMER
8 ☐ DATA COMMUNICATIONS/TELECOMMUNICATIONS	108 ☐ DATABASE MANAGER
9 ☐ COMPUTER OPERATIONS	109 ☐ DATABASE ADMINISTRATOR
10 ☐ PRODUCTION CONTROL	110 ☐ MANAGER OF DP OPERATIONS

CITIZEN OF UNITED STATES OF AMERICA?   ☐ Yes   ☐ No   Country:_____

Signature: _____   Date: _____

Forward To:

      DECUS U.S. CHAPTER, MEMBERSHIP PROCESSING GROUP
      219 BOSTON POST ROAD
      MARLBORO, MA 01752, USA
      PHONE: (617) 480-3418

DATAGRAM

DATAGRAMs are short messages, comments, requests, or answers
that are published in NETwords. Please fill in the sections below
and send the DATAGRAM to:

Vickie Hancock
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Title: _____

Message: _____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

_____

Your Name: _____

Address: _____

_____

Telephone: _____

If this is a reply to a previous DATAGRAM, what *? __

Signature: _____ **Date:** ____

Vickie Hancock
NETWords Editor
2510 Limestone Ln.
Garland, Tx. 75040

Fold Here

INPUT/OUTPUT Submission Form

A SIG Information Interchange

Please reprint in the next issue of the Pageswapper

If this is a reply to a previous I/O, which number? _____

Caption: _____

Message: _____

_____

_____

_____

_____

_____

_____

Contact:

Name _____

Address _____

_____

_____

_____

Telephone _____

Signature _____ Date _____

Mail this form to: Larry Kilgallen, PAGESWAPPER Editor
Box 81, MIT Station, Cambridge, MA 02139-0901, USA

Tear out or photocopy reverse to vote on SIRs

Gary L. Grebus
Battelle Columbus Division
Room 11-6011
505 King Avenue
Columbus, Ohio 43201-2693
USA

System Improvement Request Submission Form

Page 1 of _____

Submittor: Firm:

Address: Phone:

How to write an SIR:
Describe the capability you would like to see available on VAX
systems. Be as specific as possible. Please don't assume we
know how it's done on the XYZ system. Justify why the capability
would be useful and give an example of its use. If you wish,
suggest a possible implementation of your request.

Abstract (Please limit to four lines):

Description and examples (use additional pages if required)

Tear out or photocopy reverse to submit an SIR

Gary L. Grebus
Battelle Columbus Division
Room 11-6011
505 King Avenue
Columbus, Ohio 43201-2693
USA

VAX System SIG Spring 1986 SIR Ballot

DECUS membership number _____ (six digits)

Our site uses the following VAX models (check all that apply)

8600 ____   11/782 ____   11/780,11/785 ____   11/750 ____
11/730,11/725 ____   MicroVAX ____

We use VAX's in the following applications (Check all that apply)

Business EDP ____   Software Development ____
Education ____   Computer Science Research ____
Data Acquisition/Control____   CAD/CAM ____
Service Bureau ____   Hardware Development ____
Scientific/Engineering ____   Office Automation ____
Telecommunications ____
Other _____

I support the following as the most important System Improvement
Requests. (List from zero to fifteen SIR's):

SIR Number:

_____   _____   _____   _____   _____

_____   _____   _____   _____   _____

_____   _____   _____   _____   _____

I oppose the following SIR's as detrimental. (List from zero to
five SIR's):

SIR Number:

_____   _____   _____   _____   _____

Mail to:

Gary L. Grebus
Battelle Columbus Division
Room 11-6011
505 King Avenue
Columbus, OH 43201

To be counted, you ballot must be received by April 1.

Tear out or photocopy reverse to vote on SIRs

Gary L. Grebus
Battelle Columbus Division
Room 11-6011
505 King Avenue
Columbus, Ohio 43201-2693
USA

STATUS CHANGE

Please notify us immediately to guarantee continuing receipt of DECUS literature. Allow up to six weeks for change to take effect.

(   ) Change of Address
(   ) Please Delete My Membership Record
        (I Do Not Wish To Remain A Member)

DECUS Membership No: _____

Name: _____

Company: _____

Address: _____

_____

State/Country: _____

Zip/Postal Code: _____

Mail to: DECUS - Attn: Subscription Service
          219 Boston Post Road, BP02
          Marlboro, Massachusetts 01752 USA

Affix mailing label here. If label is not available, print old address here. Include name of installation, company, university, etc.