DECUS U.S. CHAPTER

SIGs

NEWSLETTER

JANUARY
FEBRUARY
MARCH
APRIL
MAY
JUNE
JULY
AUGUST
SEPTEMBER
OCTOBER
NOVEMBER
DECEMBER

VOLUME 4

NUMBER 1

Your best connection for:

peer-to-peer interaction

technical exchange

DECUS

Printed in the U.S.A.

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

It is assumed that all articles submitted to the editor of this newsletter are with the authors' permission to publish in any DECUS publication. The articles are the responsibility of the authors and, therefore, DECUS Digital Equipment Corporation, and the editor assume no responsibility of liability for articles or information appearing in the document. The views herein expressed are those of the authors and do not necessarily express the views of DECUS or Digital Equipment Corporation.

Production Staff:

Beverly Welborne: Communications Committee Chair
R.E. Center

Frank Borger: SIG Publications Chair
Michael Reese Hospital

Judy Mulvey: Publications Manager
DECUS

Judy Tessier: Phototypographer/Graphics Designer
DECUS

Circulation: 6344

GENERAL TABLE OF CONTENTS

SECTIONS PAGE NO.

GENERAL INTEREST

DAARC SIG

DATATRIEVE/4GL SIG

EDUSIG

GRAPHICS SIG

L&T SIG

MUMPS SIG

OFFICE AUTOMATION SIG

PERSONAL COMPUTER SIG

RSX/IAS

RT SIG

UNISIG

VAX SIG

LIBRARY

SIG INFORMATION SECTION

SUBMISSION FORMS

QUESTIONNAIRE SECTION

SUBSCRIPTION & MEMBERSHIP FORMS

Digital Equipment Computer Users Society

DECUS, U. S. Chapter
219 Boston Post Road, (BP02)
Marlboro, MA 01752

U. S. Activities (617) 480-3259 or 3302
Finance & Admin. (617) 480-3634
Library (617) 480-3521

Dear DECUS Member,

This editorial was supposed to be included in last month's issue, to accompany the new Newsletters cover, but a bout on my part with pneumonia left me in the hospital and home recuperating for the better part of a month, so it's getting to you now. Oh well, we don't promise that any section publishes every month, and that must go for my front of the book ramblings also.

I hope you readers don't think the new cover is an attempt to repackage an existing product with "New ... ·Improved" splashed across the cover like some brands of laundry soap. I believe that there has been a healthy improvement in the quality, diversity and usefulness of the Newsletter lately. The inside is our real product, we're just changing the outside for the same reason that you readers go out and by some new clothes to celebrate something good that happened, and I think a lot of good things have happened to the newsletters lately.

Take a few minutes and compare the last few issues with their counterparts from a year earlier. There are a lot of changes, changes that are the result of the hard work of a bunch of editors, (and more contributors who have decided that the editors should not have to write the whole bloody issue.) There are also changes as a result of the Fall 87 readers' survey, (reported on here a couple of issues ago.)

One of those changes involved the "pilot project," an attempt to use stock DEC micros and software to produce a unified newsletter look. As almost everybody knows by now, it didn't work. Forget the problem of the editor not being able to proofread their work, unless they happened to have a spare LN03 laying around; forget the learning problem of most of the editors having to quit using Runoff, Troff, or their favorite package, and learn the "standard" text processing package; (again without a printer to see what they were getting,) forget explaining to your boss where that pile of hardware came from, and why you're not paying for it, etc. Anyway, with 20/20 hindsight we can always spot JUST what went wrong with any project that failed. That wasn't the only problem. The other editors, the ones not blessed, (or cursed) with pilot project status, were left waiting for someone to decide what to do next.

I never liked the idea of all the SIGs conforming to the same format, but I was only one of 21 editors. When I took over this product, however, I could do something. One of the first things I did was to let the editors off their leashes. I stated my policy as; "You have nnn pages, to do with as you wish. Use any text processing hardware and software you want. Go to portrait mode or stay with two-up landscape, two-column or one column is OK too." I also returned to the old ordering, (except that we now separate portrait from two up, so readers don't have to keep turning the newsletters back and forth every time they switch SIGs.)

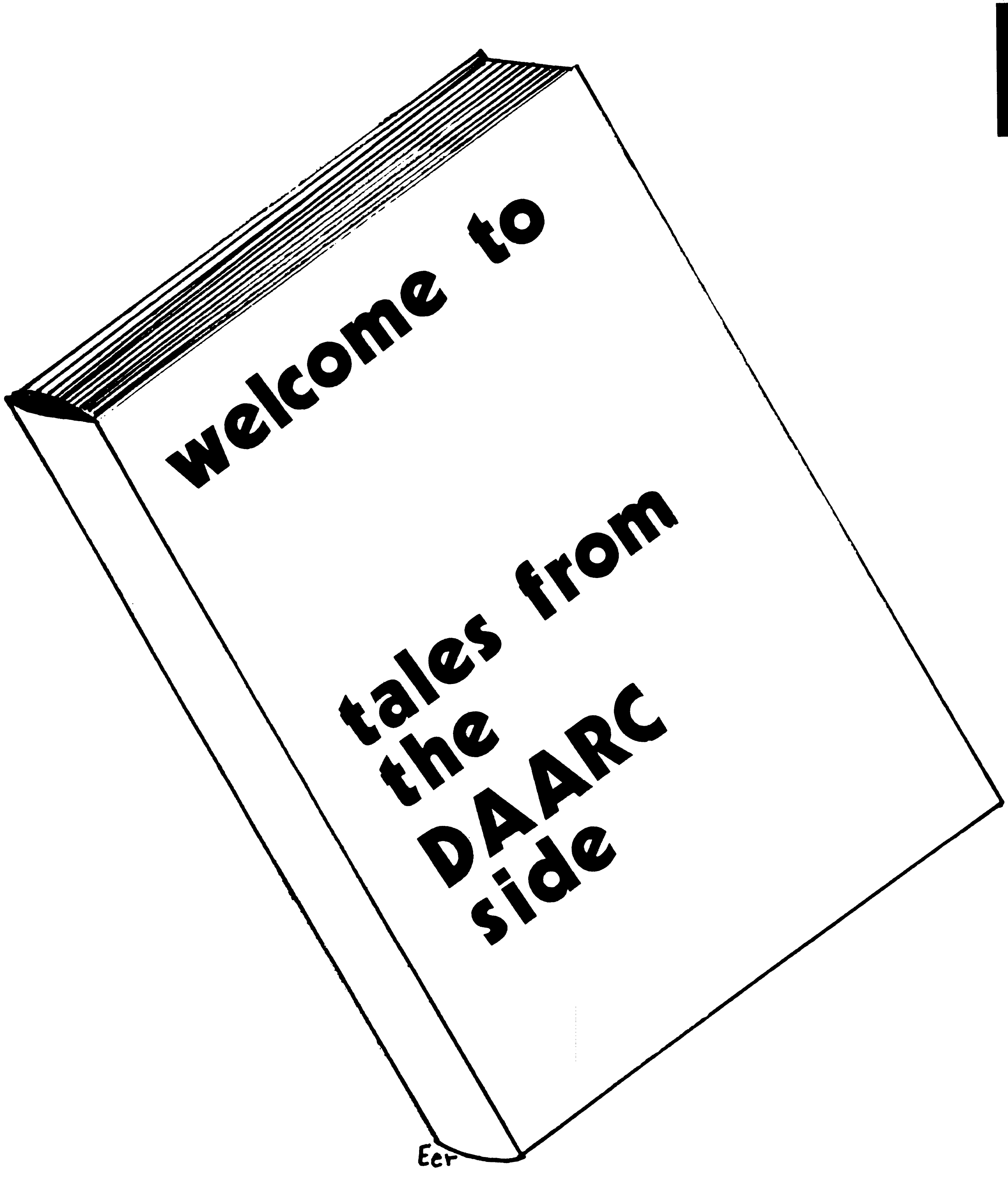
I think you are seeing the results of this emancipation. I hope I have stimulated some latent SIGs to more active publishing. (At least, we seem to be working away from the old "Snow VAX and the twenty dwarfs" style newsletters.) I hope I have been a catalyst in migration of some SIGs to the use of more text processing software and better printers.

The one thing I don't want to do is to make those SIGs still using the older formats to feel like stepchildren. We don't want to force anyone into joining the glitzy generation. If we have an editor turning out a quality section that just happens to be so chock-full of stuff that it needs to be printed two up, it WILL stay that way, as long as the editor wants it to. That's my other policy concerning portrait versus two-up.

Finally, I have to say goodbye to Larry Kilgallen as VAX newsletter editor. Only an editor can truly appreciate the monumental amount of work that he put in on the Pageswapper. Burnout is an often used word when editors get together, and Larry published the Pageswapper for what seems like forever. I really don't know how he did it. Its rather hard to give a standing ovation from a keyboard, but if they could, I think all the editors would be standing up with me after I made this statement. A simple "Thanks" just does not suffice, but it's the best I have under these circumstances. Thanks Larry.

We're not resting on our laurels. When I finally got to the Communications Committee Meeting last weekend, (after missed connections, lost baggage, and a nights' sleep at Boston's beautiful Logan airport lounge,) we discussed more changes and improvements for the Newsletters. We're continually working on improving this product. Give us a hand, if you have any ideas, yeas, boos, or whatever you want to express, write me or give me a call. We're always open to new ideas.

Frank R. Borger


welcome to
tales from the DAARC side
Eer

DATA ACQUISITION ANALYSIS, RESEARCH AND CONTROL SIG FALL SESSIONS

Symposium Coordinator: Mack Overton
SIG Chair: James Deck
Newsletter Editor: Dale Hutchison

The DAARC SIG invites you to the 1988 Fall DECUS Symposium. As you look through the following session descriptions you will notice an exciting array of sessions relating to:

- Real-time Data Acquisition and Control Applications
- Real-time Performance Issues
- VAXeln
- Signal Processing and Data Analysis
- Computer Integrated Manufacturing
- LIMS/SM Laboratory Data Systems
- Distributed Data Acquisition

We particularly direct your attention to a series of real-time performance sessions on Thursday afternoon and the Q and A session on Friday morning. These sessions are expansions of two very popular real-time performance sessions presented at the spring symposium and should be marked must attend. So plan a late Friday or Saturday departure.

We look forward to seeing you in Anaheim.

DA001
Tuesday, October 18, 1988
10:00 - 10:30 A.M.
Orange County 19 Room, Anaheim Convention Center
RADAR DATA ACQUISITION USING MICROVAX AND VAXELN CUSTOM DRIVER

The Hardware and software used to acquire tracking radar data for input to a MicroVAX are described in this session. The source data is transmitted in a serial stream. Interface hardware recovers the data, converts it to 16-bit parallel, and prepares it for transfer to the MicroVAX. A DRV11W DMA board, controlled by a VAXELN handler, is used in the MicroVAX.

DA002
Thursday, October 20, 1988
9:00 - 10:00 A.M.
California C Room, Anaheim Convention Center
MANUFACTURING MONITORING USING VAXELN AND RDB/ELN
ELN MANUFACTURING MONITORING

GE Lighting had developed Manufacturing Information Diagnostic Analysis (MIDAS) for monitoring its high speed lamp manufacturing equipment. MIDAS operates on MicroVAX II hardware using VAXELN (Digital's real-time VAX operating system) and Rdb/ELN software. This session addresses the technical details of the project:

- MIDAS requirements
- Why VAXELN and Rdb/ELN were chosen
- How MIDAS was implemented using VAXELN and Rdb/ELN.

DA003
Monday, October 17, 1988
11:00 - 12:00 noon
Orange County 19 Room, Anaheim Convention Center

INFORMATION MODELING: THE KEY TO CIM INTEGRATION

Computer Integrated Manufacturing (CIM) is too big to be done all at once, so it must be implemented in small manageable pieces. Without a good information modeling methodology, such as NIAM, the pieces will not fit together once they are developed. A NIAM information model captures all of the information needed to understand the information requirements and generate a database schema.

Information modeling, as opposed to more conventional data modeling, provides the critical user-oriented level that is missing from traditional information architectures. Combustion Engineering is using NIAM to understand, prototype, and integrate several major engineering and manufacturing systems.

This paper explains the basic NIAM concepts, including facts, objects, roles, ideas, bridges, and constraints. It describes how to automatically generate a database schema from the information model. Using CIM information models, it also shows how to easily integrate multiple NIAM models that were developed independently to support different CIM functions.

DA004
Wednesday, October 19, 1988
9:00 - 9:30 A.M.
California C, Anaheim Convention Center
MICROPOWER/PASCAL PRODUCT PANEL

This session discusses the development efforts and future plans for MicroPower/Pascal. A question-and-answer session with representatives of the MicroPower/Pascal development group follows the presentation.

DA005
Wednesday, October 19, 1988
10:15 - 11:00 A.M.
California C, Anaheim Convention Center
DEVELOPING AN APPLICATION WITH VAXELN

This session discusses a typical application developed using the VAXELN Tool Kit. It gives a brief outline of the application, with examples of the different VAXELN techniques used to solve some of the more interesting application problems.

DA006
Wednesday, October 19, 1988
9:30 - 10:15 A.M.
California C, Anaheim Convention Center
MICROPOWER/PASCAL-TO -VAXELN MIGRATION CONSIDERATIONS

This session contrasts the MicroPower/Pascal and VAXELN environments, and makes practical suggestions for easing migration of applications from one to the other.

DA007
Thursday, October 20, 1988
10:00 - 11:00 A.M.
California C, Anaheim Convention Center
INTRODUCTION TO RDB/ELN AND ITS USE IN REALTIME APPLICATIONS

This session provides an overview of the RDB/ELN product, and discusses it's use in providing a variety of application solutions in a Realtime application.

DA008
Wednesday, October 19, 1988
3:00 - 4:00 p.m.
California C, Anaheim Convention Center
REALTIME I/O WITH VAXELN

This session covers techniques for doing real-time I/O, including device driver design and implementation. The presentation discusses performance implications for device drivers, along with the multiple approaches available from VAXELN for doing Realtime I/O.

DA009
Wednesday, October 19, 1988
4:00 - 5:00 p.m.
California C, Anaheim Convention Center
VAXELN PERFORMANCE TECHNIQUES AND ANALYSIS

This session discusses the performance analysis and resource usage tools valuable within a VAXELN system. These tools can be used to tune the application for maximum performance.

DA010
Wednesday, October 19, 1988
2:00 - 3:00 p.m.
California C Room, Anaheim Convention Center
DEVELOPING VAXELN APPLICATIONS WITH FORTRAN-77

This session discusses techniques for developing VAXELN applications using VAX FORTRAN-77. The presentation includes a discussion of techniques for interfacing MACRO-32 routines and device drivers.

DA011
Wednesday, October 19, 1988
5:00 - 6:00 p.m.
California C Room, Anaheim Convention Center
MULTI-PROCESSING WITH VAXELN

Many high speed, real-time applications demand some type of multiprocessing configurations. This session focuses on VAXELN multiprocessing concepts, kernel services, and VAXELN synchronization/ communication facilities. This presentation also highlights product features, such as symmetric multiprocessing and VAX Realtime Analyzer (RTA) support.

DA012
Tuesday, October 18, 1988
9:00 - 10:00 A.M.
Orange County 19 Room, Anaheim Convention Center
VAXELN TECHNICAL OVERVIEW

This session focuses on the detail concepts used in VAXELN applications. These include kernel services, synchronization features, coexistence with VMS, and process scheduling. The presentation also highlights new features and capabilities of the product.

DA013
Friday, October 21, 1988
12:30 - 1:30 P.M.
California D Room, Anaheim Convention Center
LABORATORY AUTOMATION WORKING GROUP

This is an open forum for discussion of all aspects of laboratory automation. It addresses hardware, software, and other issues. Users of third-party systems, and home-grown systems, as well as users of Digital hardware (e.g. VAXlab) and software such as Laboratory Information Management System/Sample Management (LIMS/SM) are encouraged to come. Participants include implementors and users of systems, and representatives from Digital. This is your chance to share your opinions and problems.

DA014
Monday, October 17, 1988
2:30 - 3:00 P.M.
California C Room, Anaheim Convention Center
RTVAX PRODUCT OVERVIEW

RTVAX products are designed to provide VAX-based Realtime target platforms for a variety of applications. This session outlines the products currently offered, including an overview of the CPU technology on which some of these products are based. The products will be positioned in terms of the roles they play in the implementation of Digital's corporate Realtime program.

A question and answer session will follow the formal presentation.

DA015
Friday, October 21, 1988
9:00 - 9:30 A.M.
California D Room, Anaheim Convention Center
THE DESIGN OF AUTOMATION TOOLS FOR THE ANALYTICAL LABORATORY

Great strides forward are being made in the development of tools for the automation of analytical laboratories. This session focuses on the design philosophy behind the user configurable modules within a laboratory information management system and on the flexibility this approach provides the user. Design examples of LOGIN (a laboratory application language) and a template driven sample entry module are presented. Specific applications which utilize these tools are shown. The design requirements and implementation of a tailorable results entry module are also discussed. This module provides for user definable prompts, data fields and calculations. It also enables the user to setup automatic specification checking required in many laboratories today. The design of an instrument interfacing/file parsing module is also presented with examples to illustrate the program's capabilities.

DA016
Thursday, October 20, 1988
5:00 - 6:00 P.M.
California C Room, Anaheim Convention Center
BLAST FURNACE PROCESS DATA ANALYSIS AND ACQUISITION

This session is a case study in the acquisition, historical archival, and analysis of blast furnace process data at the LTV Indiana Harbor Works H3, and H4 Blast Furnaces.

This session discusses how the data is acquired, how it is stored, how it is played back, and the software that was integrated into the system for the analysis (Digital and third party software). This session provides information on how the data analysis is used to provide: feedback performance of the Blast Furnaces operations, predictive maintenance, and the feed forward of data into burdening models for the furnace.

DA017
Monday, October 17, 1988
4:30 - 5:30 P.M.
California C Room, Anaheim Convention Center
DESIGNING REALTIME APPLICATIONS WITH THE C LANGUAGE AND RSX

This session presents a methodology for the design of real-time programs for data acquisition and process control applications. Specific techniques are illustrated with the C language and RSX operating system, but most of the principles apply to other languages and environments. Topics that will be covered include:

- Partitioning a problem into multiple tasks;
- Establishing task priorities;
- Using shared common regions;
- Accessing devices through I/O common;
- Services; and managing the user interface.

A database driven design framework for real-time applications is described.

DA018
Thursday, October 20, 1988
1:00 - 2:00 P.M.
California C Room, Anaheim Convention Center
DATA ACQUISITION TECHNIQUES WITH COMPUTER AUTOMATION MEASUREMENT AND CONTROL (CAMAC)

This session provides a brief overview of CAMAC for data acquisition and control applications. Various techniques are discussed for solving data acquisition applications. Such issues as how to optimize throughput, operating system considerations, interrupt latency issues, "gapless" transfers, etc. are discussed.

DA019
Friday, October 21, 1988
2:30 - 3:30 P.M.
California D Room, Anaheim Convention Center
DAARC BUSINESS MEETING

This session provides an opportunity for those interested in becoming actively involved in the operation of the DAARC SIG to meet with the officers of the SIG and discuss what we do right, what we do wrong, and how they think they can contribute to the combined effort. If you have an interest in promoting activities within DECUS relating to real-time data acquisition and control, computer integrated manufacturing (CIM), or data analysis, we invite your participation.

DA020
Tuesday, October 18, 1988
9:30 - 10:30 A.M.
Fullerton 8 Room, Anaheim Convention Center
RS/1 WISHLIST AND WORKING GROUP

This session is an opportunity for RS/1 users to meet in an informal setting and discuss needs and applications. The future of RS/1 in DECUS is also an agenda item.

DA021
Friday, October 21, 1988
1:30 - 2:30 P.M.
California D Room, Anaheim Convention Center
CIM TOPICS

This session provides an interactive forum for discussing CIM problems and solutions, to aid in formulating feedback to Digital on CIM needs, improvements, and directions. The session also addresses the needs of the user community for future CIM sessions at DECUS.

DA022
Monday, October 17, 1988
9:00 - 9:30 A.M.
Orange County 19 Room, Anaheim Convention Center
DAARC ROADMAP

This session is intended to point DAARC attendees toward our sessions as well as highlight those sessions of other SIGS which may be of interest to DAARC attendees. Last minute changes and additions are also discussed as well as items of general interest which make your attendance at this symposium more enjoyable. Plan to attend and avoid disappointments.

DA026
Thursday, October 20, 1988
12:00 noon - 12:30 P.M.
California C Room, Anaheim Convention Center
A REALTIME SIGNAL PROCESSING DATA ACQUISITION SUBSYSTEM

A digital signal processing subsystem has been developed for a coherent carbon dioxide laser radar system at Lincoln Laboratory's Firepond Infrared Research Facility. The subsystem integrates the tasks of real-time data acquisition and control, signal processing, and imaging.

The subsystem's distributed processing hardware design is hosted by a MicroVAX II. Attached to the MicroVAX II is an Aptec IOC-24 system providing a fast bus to quickly move data between system components. These system components consist of the following specialized peripherals; digitizers, an array processor, an image processor, high speed disk drives, and tape drives. The peripheral devices are all off-the-shelf systems with connection capability to a VAX system.

The software for the complete Data Acquisition and Signal Processing subsystem is developed on a MicroVAX II computer. Software developed for the completed system is coded primarily in high level languages to promote flexibility, modularity, and to reduce development time. Microcode had to be used where speed is essential. All software design, development, and testing was done under VMS.

The complete signal processing subsystem: criteria, configuration, and software will be discussed.

DA027
Tuesday, October 18, 1988
3:00 - 4:00 P.M.
Orange County 19 Room, Anaheim Convention Center
MULTI-FREQUENCY DATA ACQUISITION USING VAXLAB

This session will cover the trials and tribulations of developing an application program to do synchronized multi-frequency data acquisition using the VAXlab hardware and VAXlab Subroutine Library (formerly LABSTAR) software.

DA028
Tuesday, October 18, 1988
2:00 - 3:00 P.M.
Orange County 19 Room, Anaheim Convention Center
HIGH RESOLUTION TIME STAMPING ON VAXBI BASED VAXES

Many instrumentation applications require the host processor to know when events happen in real time and track them accordingly. VAX/VMS is often derided as a non-realtime architecture and is often discounted as a potential solution in these environments. This session will concentrate on special hardware and software tools that allow VAXBI based machines to perform real time clocking, event counting and time stamping at high resolution.

DA030
Thursday, October 20, 1988
3:00 - 4:00 P.M.
California C Room, Anaheim Convention Center
REALTIME SYSTEM PERFORMANCE CAPABILITIES

Developing real-time applications requires a good understanding of a system's responsiveness to external events. We have examined latency to interrupt service routines, fork process (where appropriate), and user process for VMS, VAXELN, RSX and RT. This talk describes our methodology and instrumentation, with particular emphasis on ensuring reproducibility of the tests, avoiding test data correlations with system activity, and explaining the blocking and preemption activities which affect latencies. We specifically avoid attempting a simplistic comparisons among systems; our goal is rather to discuss strategies for configuring particular systems based on application needs. To this end, we will not emphasize specific numbers but rather discuss the overall distribution of responses to the interrupts.

DA031
Thursday, October 20, 1988
7:00 - 8:00 P.M.
California C Room, Anaheim Convention Center
REALTIME SYSTEM TUNING

We lay the groundwork for this talk with a brief recapitulation of our methodology of our interrupt latency tests which were discussed in our "REALTIME SYSTEM PERFORMANCE" session. Our tests were initially run on minimally configured systems. This allowed us to add loads incrementally and observe the effects of each addition. We looked at the impact of hardware and software, including video overhead, disk and network activity, and competing tasks and processes. We experimented with adjustments to system tuning parameters in order to assess the effects of such parameters on interrupt latency performance.

DA032
Friday, October 21, 1988
10:00 - 11:00 A.M.
California D Room, Anaheim Convention Center
REALTIME PERFORMANCE Q & A

Members of various groups within DEC will be available to discuss various aspects of real-time data acquisition performance monitoring and tuning. The discussion includes an overview of such topics as:

- high speed data acquisition
- various hardware interface alternatives, including RS-232, IEEE-488, and direct processor bus interfaces
- operating system considerations.

Audience participation is encouraged.

DA033
Thursday, October 20, 1988
4:00 - 5:00 P.M.
California C Room, Anaheim Convention Center
COMMUNICATIONS PERFORMANCE IN A DISTRIBUTED REALTIME SYSTEM

Two system performance metrics of a distributed real-time system are latency and throughput of messages when between nodes. Our model of such a system is hierarchical, with a "supervisor" and several "low level" nodes. Given this model, the types of messages of interest to us are event notification (low-level machine to supervisor without a response), command/response (supervisor to low-level with a response), and bulk data transfer (low-level machine to supervisor in large bursts).

We have examined the performance of these messages on Q-bus systems interconnected by Ethernet. We compared datagram and virtual circuit delivery with the "low-level" machines running ELN or RSX and the "supervisor" running VMS. This talk will describe our test methodology, our results to date, and analysis of these results with emphasis on tuning an application for optimal performance.

DA034
Monday, October 17, 1988
1:00 - 2:00 P.M.
California C Room, Anaheim Convention Center
DIGITAL'S RESPONSE TO REALTIME

This session focuses on Digital's directions and strategies for Realtime computing. Digital's architecture and goals for distributed and self sufficient real-time computing are covered along with the positioning of Digital's real-time products against that architecture. New products, introduced since the Spring '88 US DECUS Symposium, are discussed.

DA035
Monday, October 17, 1988
2:00 - 2:30 P.M.
California C Room, Anaheim Convention Center
CONFIGURING REALTIME COMPUTING PLATFORMS

This session describes a solution to the problem of configuring low end VAXELN development and execution systems. The presentation pays specific attention to the connectivity issues that arise when using MicroVAX 2000s networked to RTVAX 1000 targets.

DA036
Thursday, October 20, 1988
6:00 - 7:00 P.M.
California C Room, Anaheim Convention Center
STEEL ROLLING MILL AUTOMATION

At LUKENS STEEL Conshohocken we took on an aggressive task to automate our existing reversing mill operations in 1984. We contracted with CLECIM, a French corporation, to develop the system and retrofit the controls to our existing mills. Installed last year, it is producing steel of higher quality and consistency than before. This process was not a cake walk however, we had problems retrofitting the old mill.

This session discusses the mill setup and the hardware and software that is used to run the automation. Also some of the experiences during commissioning are discussed. We use a VAX 11/750 running VAX/VMS communicating with two level 1 INTEL 80186 over Ethernet.

DA037
Tuesday, October 18, 1988
1:00 - 2:00 P.M.
Orange County 19 Room, Anaheim Convention Center
INDUSTRIAL NETWORKING VIA GE GENIUS(™) NETWORK INTERFACES

This session describes a method of networking industrial I/O equipment and programmable logic controllers using the GE GENIUS series of network interfaces. Specific attention will be given to the networking protocols, and the design of the RSX, VMS, and VAXELN device drivers.

DA039
Tuesday, October 18, 1988
10:30 - 11:00 A.M.
Orange County 19 Room, Anaheim Convention Center
INSTRUMENT INTERFACING — BECKMAN LAB MANAGER METHOD

Laboratory Automation starts at the instrument connectivity level. Each analytical instrument vendor uses a different method to accomplish the communication. In this sessions, Beckman Instruments' Laboratory Automation Group discusses their interfacing language, programmable interface, and issues relating to instrument integration.

DA040
Monday, October 17, 1988
10:30 - 11:00 A.M.
Orange County 19 Room, Anaheim Convention Center
LIMS UTILITIES SUPPLIED TO THE DECUS SOFTWARE LIBRARY

This session discusses the Laboratory Information Management System (LIMS) utilities that are being supplied to the DECUS Software Library. Procedures for obtaining this software as well as a brief description of each utility will be mentioned. Users will be encouraged to use the Software Library for an exchange of public domain software related to the LIMS/SM product.

DA041
Thursday, October 20, 1988
2:00 - 3:00 P.M.
California C Room, Anaheim Convention Center
SHARED ACCESS TO THE IEEE-488 BUS

The IEQ11, Digital's IEEE-488 interface supports up to 14 instruments per port. However, since it is a non-shareable VMS device, only one process may use it at one time. To take full advantage of IEEE-488 users may need to have separate processes communicating with the various instruments on the bus. For example, user A may wish to collect data from a voltmeter while user B collects data from a digital scope on the same bus. Without special programming this scenario is not possible.

VMS supports several mechanisms for sharing resources. These include shareable device drivers, global sections, inter-process communication, lock management, ACP's, and server processes. These mechanisms will be discussed and applied to the problem of sharing an IEEE-488 bus between users.

Finally an example service process will be presented to show how processes can share the IEQ11. If time permits, this example program may be made available to session attendees.

DA043
Tuesday, October 18, 1988
11:00 - 11:30 A.M.
Orange County 19 Room, Anaheim Convention Center
LIMS INSTRUMENT INTERFACING

Instrument interfacing is more than just plugging into a device into a terminal port. Consideration must be given to the ease of integration into the workplace, the human engineering, and the maintainability of any instrument interfacing system. Laboratory Data Products has designed and built a system that gives Digital's Software Services the ability to provide customers with a coordinated and integrated system of interfaces for their lab instruments. This presentation describes that system.

DA044
Monday, October 17, 1988
8:00 - 9:30 P.M.
California C Room, Anaheim Convention Center
DATA ACQUISITION AND ANALYSIS UNDER VAX/VMS

Scientists require computer systems that provide a wide range of capabilities ranging from real-time data acquisition, to data analysis to report generation, and the ability to transfer data easily from one function to another. This session discusses tools and applications that address these needs in a VAX/VMS environment. Digital and third party products are covered, with emphasis on the VAXlab product family.

DA045
Tuesday, October 18, 1988
11:30 - 12:00 noon
Orange County 19 Room, Anaheim Convention Center
VECTOR AND ARRAY PROCESSORS - A COMPARISON

To address the time-to-solution needs of compute intensive floating point applications, there are a growing number of different products being introduced. Two types of technologies (array processor and vector processor) have been around for 20 years and continue to be used. Both approaches can be used to reduce time-to-solution (execution time) of compute intensive floating point applications. When to use one approach versus the other can be difficult to determine. This presentation discusses the similarities and differences between the two approaches and provide some guidelines for selection.

DA046
Thursday, October 20, 1988
12:30 - 1:00 P.M.
California C Room, Anaheim Convention Center
REALTIME CALLING STANDARD PROGRAM

The Realtime Calling Standard (RTCS) provides a common software interface to a wide range of real-time I/O devices, allowing real-time applications to be written in a way that is easily portable among processors, real-time devices, and I/O bus architectures. This session gives an overview of the RTCS program.

DA047
Monday, October 17, 1988
3:30 - 4:30 P.M.
California C Room, Anaheim Convention Center

THE GESTALT OF REALTIME

Computer-based real-time response systems must be designed, configured, and implemented with special care in order to succeed. This session investigates the nature of electronic systems which respond to events as they occur, emphasizing inherent hardware and software requirements.

The session explores:

- defining functional requirements,
- special requirements for time critical applications,
- benchmarking,
- configuration alternatives,
- configuring hardware and software,
- selecting implementation alternatives,
- designing applications, and
- management and potential legal aspects

for computers which monitor or control processes, sense and respond to human input, or otherwise respond to events quickly.

DA048
Thursday, October 20, 1988
8:00 - 9:00 P.M.
California C Room, Anaheim Convention Center

REALTIME PERFORMANCE CHARACTERIZATION OF VAX/VMS SYSTEM SERVICES AND OTHER SYSTEM COMPONENTS

This session describes the procedures for testing different components of VAX real-time systems. This includes the timing of VAX/VMS system services that are applicable to real-time computing and control, guidelines as to which system services are best suited for real-time applications, and the demands which each of the services places on system resources. Also discussed are effects of context switching and the methodologies of testing the latency of a context switch at process level. Statistical analysis of actual test results are included in the presentation.

DA049
Monday, October 17, 1988
7:00 - 8:00 P.M.
California C Room, Anaheim Convention Center

MICROVAX REALTIME PROGRAMMING TECHNIQUES

This session describes techniques for timing programmed I/O operations on VAX systems under VMS. The presentation specifically addresses the following topics:

- directly accessing the I/O page through a mapped global section
- executing code at elevated processor priority
- use of the VMS connect-to-interrupt driver.

These techniques enable intermediate or advanced programmers to program devices for real-time I/O under VMS without writing standard device drivers.

DA050
Tuesday, October 18, 1988
4:00 - 5:00 P.M.
Orange County 19 Room, Anaheim Convention Center
USING A VAX TO AUTOMATE A REALTIME TEST FACILITY

The Space Shuttle Main Engine (SSME) Simulation Lab provides a test bed for verifying the flight software for the SSME controller. The lab and its supporting software provide an apparently homogeneous test environment by defining a test language which uses the syntax of the system under test and a user interface consistent with that of the VAX.

Some of the hardware integrated into the lab and controlled by this language incudes a VAX (which provides primary control of the system), a PDP, an Intel based ground control system, a Motorola 68000 based fault insertion system, and a Tektronix DAS 9200. The user interface was made to appear like DCL so that it would seem to be a native part of VMS. This session provides an overview of the lab and the philosophy behind its development as an integrated test bed.

The second part of the session illustrates the methods used to make the user interface appear as a native part of DCL and the advantages gained by doing this.

DA051
Tuesday, October 18, 1988
5:00 - 6:00 P.M.
Orange County 19 Room, Anaheim Convention Center
COMPILERS ARE REALTIME CONTROL IN THE SPACE SHUTTLE MAIN ENGINE LAB

The compilers developed for the Space Shuttle Main Engine (SSME) Simulation Lab allow the user to develop test procedures using a specialized test language. The compilers execute on the VAX and generate code for "pseudo" machines implemented in software and running on heterogeneous hardware. The database provides the mechanism for maintaining the lab configuration and contains the hardware and software information necessary for the compiler to resolve identifiers.

The real time control and data reduction is provided by the "pseudo" machines that serve as the targets for the code generated by the compiler. The target hardware includes a VAX, a PDP, and Intel based ground control system, a Motorola based fault insertion system, and a Tektronics DAS 9200. The real-time control part of the system operates in a 20 millisecond control loop. The data reduction processor runs on the VAX.

The speakers will focus on the concept of "pseudo" machines; using a data base to maintain information required by the lab; problems encountered with synchronization, communication, and timing in the real-time processors (particularly the PDP); and the implementation of the data reduction processor as a "pseudo" machine on the VAX.

DA052
Friday, October 21, 1988
9:30 - 10:00 A.M.
California D Room, Anaheim Convention Center
BITBUS PERFORMANCE MEASUREMENTS AND ANALYSIS WITH A MICROVAX AS THE MASTER NODE

BITBUS is Intel's distributed control network for automation, PLC control, and cell control. Data acquisition functions are also implemented over BITBUS. As a master-slave network, the performance of BITBUS systems are largely determined by the performance of the master node. This session presents performance data for a MicroVAX as a BITBUS master node. Further it presents an analysis of application interface specifics for this implementation.

DA053
Tuesday, October 18, 1988
12:00 noon - 1:00 P.M.
Orange County 19 Room, Anaheim Convention Center
MICROPROCESSOR FARMS FOR HIGH SPEED DATA ACQUISITION AND ANALYSIS

There exist many applications which require extremely high I/O rates for data acquisition or reduction. A particular subset of these problems are easily separable or 'event orient', i.e. they are readily divided into uncorrelated segments which can be handled by independent processing nodes.

Such problems lend themselves to loosely coupled microprocessor farm architectures. A particular implementation consists of microprocessor nodes coupled both by a very high speed data transport bus, and a slow speed Ethernet bus. The embedded processing elements consist of Digital MicroVAX boards. The data transport bus is implemented using ZRL Q22DPM and Q22MPM multi-port memory boards capable of transferring data at rates up to 80 Mbytes/sec. These data busses can be used for moving data from front end electronics, or for message passing between nodes. Aggregate burst throughputs per node of 1 Gbyte/sec are possible. Additional memory ports allow the attachment of special purpose co-processors, as well as a separate communication bus to establish shared memory regions between nodes. The farm itself operates entirely under the VAXELN memory resident operating system, with connections to a host VAX running VMS.

This architecture lends itself to a variety of applications ranging from flight simulation to data acquisition for high energy physics. The use of a MicroVAX/Q22DPM based farm to implement the Level-2 trigger for the D0 collider experiment at Fermi National Laboratory will be described in detail. This trigger system will consist of an array of 50 MicroVAX processors, with performance requirements on the order of 40 MIPs at an average transfer rate of 100 Mbytes/sec. Enhancements to the present system architecture will be described, including the use of application specific co-processing elements. performance packaging, and cost specifications will be reviewed, along with a brief description of other applications.

DA054
Monday, October 17, 1988
9:30 - 10:30 A.M.
Orange County 19 Room, Anaheim Convention Center
LABORATORY AUTOMATION USING LIMS/SM

Laboratory Information Management System for Sample Management (LIMS/SM), from Digital's Laboratory Data Products group (LDP) is a VAX based system for tracking samples through the laboratory. While the original model for LIMS/SM was an industrial analytical chemistry lab, the model is applicable to a variety of similar testing environments. This presentation walks the audience through the LIMS/SM system from initial system setup, to daily operations, to data archival and retrieval. It is the goal of this session to show how the model was developed, the product designed, and the flexibility built in to allow it to be fitted into a variety of environments. While an intimate knowledge of LIMS/SM is not necessary, this session does describe many of the details of LIMS/SM that require an understanding of DBMS, DECnet, and FMS.

DA055
Wednesday, October 19, 1988
12:00 noon - 1:00 P.M.
California C Room, Anaheim Convention Center
REALTIME DATA ACQUISITION USING THE ADQ32 WITH VAXLAB SOFTWARE LIBRARY

The ADQ32, Digital's latest and greatest Analog to Digital (A/D) converter, is fully supported under the VAXlab Software Library (VSL). The A/D converter can be easily programmed using VSL, which adheres to the Realtime Calling Standard (RTCS). This session is a tutorial overview of the features of the ADQ32, with examples on how to program it.

DA056
Wednesday, October 19, 1988
11:00 - 12:00 noon
California C Room, Anaheim Convention Center
VMS V5.X, SMP AND REALTIME.

This discussion focuses on both the new features of VMS V5.x and implementation changes that are of interest to real-time system developers. Specifically we discuss the improved interrupt and DMA support, the Parallel Processing Library, and Symmetric Multi-processing (SMP). The SMP discussion includes an examination of scheduling, I/O flow, and synchronization.

DA057
Monday, October 17, 1988
5:30 - 6:00 P.M.
California C Room, Anaheim Convention Center
DISTRIBUTED SYSTEM ARCHITECTURES IN REALTIME

For many real-time applications, the most difficult part of the computer solution is that of resource management — how to ensure that the CPU and/or bus cycles required to perform time-critical tasks are available when called for. Distributed computer architectures address these problems by making it possible to compartmentalize tasks or operations within dedicated components of a configuration. This compartmentalization reduces potential unwanted interactions within the system so that high availability of system resources is guaranteed. This session examines various loosely and tightly coupled distributed architectures, and discusses how the utilization of realtime-critical resources can be optimized in each.

DA058
Monday, October 17, 1988
6:00 - 7:00 P.M.
California C Room, Anaheim Convention Center
VAX REALTIME ACCELERATOR TECHNICAL SUMMARY

This session provides a technical overview of the VAX Realtime Accelerator (VAX RTA). VAX RTA consists of hardware and software for combining VAX/VMS and VAXELN within a single, tightly coupled system. VAX RTA processors can be used to control real-time I/O devices and perform other asynchronous computing tasks in real-time applications. This greatly improves the ability to control time-critical computing resources in VAX/VMS systems. The presentation discusses design goals for the product, particularly with regard to potential improvements in resource management in real-time systems.

DA059
Friday, October 21, 1988
11:30 - 12:30 P.M.
California D Room, Anaheim Convention Center
VAXELN USERS' FORUM

This session provides an open discussion for users from VAXELN sites. If you are from a VAXeln site, please come and share your experiences and concerns.

DA060
Wednesday, October 19, 1988
1:30 - 2:00 P.M.
California C Room, Anaheim Convention Center
EXTENDING EPASCAL WITH A USER-WRITTEN PRE-PROCESSOR

This session presents a brief overview of how one development group provided the equivalent of a user-written system service package by extending the EPASCAL language. These extensions were written as an EPASCAL pre-processor that executes at compile-time on the host VMS system.

DA061
Wednesday, October 19, 1988
1:00 - 1:30 P.M.
California C Room, Anaheim Convention Center
A MENU-DRIVEN USER INTERFACE FOR VAXELN APPLICATIONS

This session describes the design and features of a screen and keyboard user interface for VAXELN that executes on a VT220 (or VAXstation) terminal. The design of this interface is highly dynamic and context sensitive. The result is an easy-to-use and very fast user interface written in EPASCAL.

DA062
Thursday, October 20, 1988
11:00 - 12:00 noon
California C Room, Anaheim Convention Center
INTERFACING A VAXELN TARGET SYSTEM TO THE WORLD: ONE USER'S EXPERIENCE

All aspects of interfacing a peripheral device to a VAXELN system are discussed in this session. For purposes of this paper, a "simple" IEEE-488 device is used as an example. The discussion stresses the software implications of the interface from the drivers at the low end on up through the running application and, eventually, the users.

DA063
Friday, October 21, 1988
11:00 - 11:30 A.M.
California D Room, Anaheim Convention Center
VAXELN PRODUCT PANEL

This session provides an opportunity for VAXELN users to meet with representatives of the VAXELN development group. The session includes a discussion of VAXELN's future vision.

DA064
Monday, October 17, 1988
3:00 - 3:30 P.M.
California C Room, Anaheim Convention Center
INTRODUCTION TO VAXELN

This session presents a high-level overview of VAXELN. The VAXELN Toolkit is a VMS layered product for developing dedicated, real-time applications for execution on VAX hardware. Topics include a comprehensive overview of VAXELN features.

DA065
Tuesday, October 18, 1988
10:30 - 11:00 A.M.
Fullerton 8 Room, Anaheim Convention Center
VAX SYSTEMS SIG VAXELN WORKING GROUP MEETING

The VAX Systems SIG VAXELN Working Group provides a forum for discussion of the needs and uses of the VAXELN Real Time Tool kit. This group also represents user desires and needs for future enhancements and acts as a clearinghouse within the VAX Systems SIG for VAXELN issues. It is the direct liaison to the VAXELN Development and Support groups within Digital, and works closely with these groups to answer the needs of its user base. The working group meeting sponsors the VAXELN Software tape distribution through the VAX Systems SIG at its meeting, and presents a forum of Digital Developers, Support Personnel, and Marketing People to speak to the future of VAXELN.

LDP Activities at Fall DECUS Symposium

Heide O'Connell, Laboratory Data Products Group

The Laboratory Data Products/Science Group is having some great activities at this Fall's DECUS in Anaheim! Please stop by our 2000 square foot booth to see demos ranging from low-cost data acquisition on the VAXstation 2000 to our VAX LIMS/SM product on the MicroVAX 3600. Our newest product, VAX Realtime Accelerator, will be showing a flight test demonstration. In another demonstration, we'll be showing the transportability of data from a VAXlab running VMS and VSL to a VAXstation II/GPX running ULTRIX and SPSS. Product information sheets and brochures can be obtained in our booth, as well.

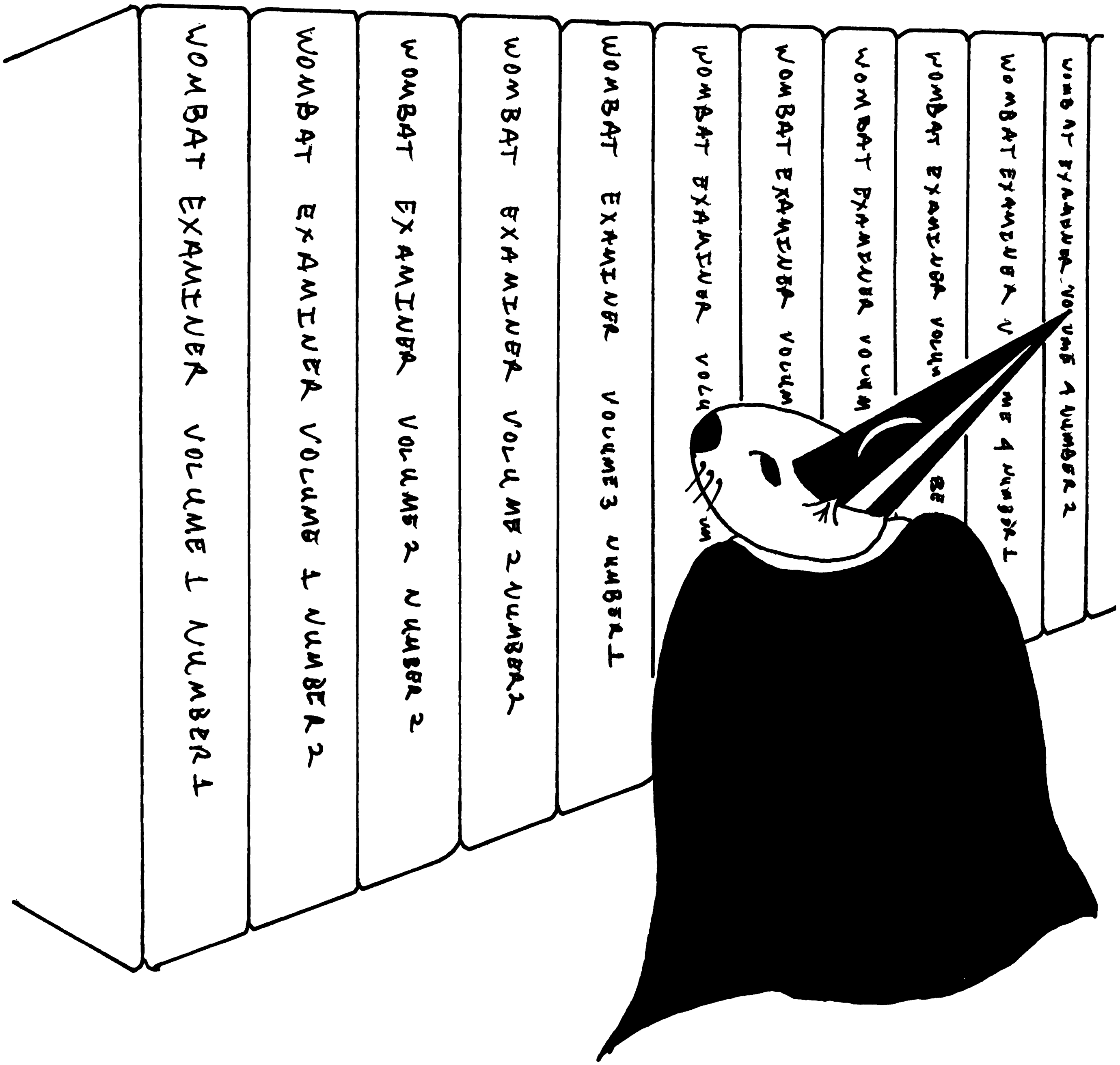
LDP Engineers will be available in DAARC SIG sessions, the joint LDP/DAARC SIG Hospitality Suite, and at the booth to answer your questions about VAXlab, VAX LIMS/SM, data acquisition and analysis, array processors, and scientific visualization products. A booth staffing schedule is available so you'll be sure the find the LDP individual you may want to meet.

Joint DAARC/LDP Hospitality Suite: LDP has joined the DAARC SIG to bring you a joint hospitality suite on Wednesday evening at the Marriott, 6:30 P.M. Stop by. We'd like the opportunity to chat with you.

The Wombat
EXAMINER and 4GL Dispatch

"Increases the Circulation of Anyone in America" *Volume 10 Number 1*

DTR


WOMBAT EXAMINER VOLUME 1 NUMBER 1
WOMBAT EXAMINER VOLUME 1 NUMBER 2
WOMBAT EXAMINER VOLUME 2 NUMBER 2
WOMBAT EXAMINER VOLUME 2 NUMBER 2
WOMBAT EXAMINER VOLUME 3 NUMBER 1
WOMBAT EXAMINER VOLU
WOMBAT EXAMINER VOLUM
WOMBAT EXAMINER VOLUM
WOMBAT EXAMINER VOLU
WOMBAT EXAMINER V ME 4 NUMBER 1
WOMBAT EXAMINER VOLUME 4 NUMBER 2

Contributions

This newsletter is a volunteer activity. There are no compensations given to any author or editor. Articles and letters for publication are encouraged from anyone. They may include helpful hints, inquiries to other users, reports on DECUS and SIG business, summaries of SPRs submitted to Digital, or any information of interest to users of either DATATRIEVE or 4th Generation Languages. However, this newsletter is not a forum for job and/or head hunting, nor is commercialism appropriate.

Machine readable input is highly desirable and machine-to-machine transfer of material is preferred, but most anything legible will be considered for publication.

Please send contributions, or for further information please contact either:

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

Joe H. Gallagher, Ph.D.
4GL Solutions
10308 Metcalf, Suite 109
Overland Park, KS 66212

Editorials and letters to the editor within the Wombat Examiner and 4GL Dispatch are solely the opinion of the author and do not necessarily reflect the views of the Digital Equipment Computer Users Society, Digital Equipment Corporation, or the author's employer. All editorials are marked as "An Editorial"; letters to the editor always begin "Dear Editor".

Table of Contents

DECUS U. S. Chapter

SIGs Newsletter, Volume 4, Number 1, September 1988
Wombat Examiner, Volume 10, Number 1

From the Editor's Pen - An Editorial

Joe H. Gallagher, Ph. D., 4GL Solutions, Overland Park, KS

At the 1987 Fall Symposia in Anaheim, we began the celebration of the 10th Anniversary of DATATRIEVE. This issue, Volume 10, Number 1, marks a milestone in the publication of the Wombat Examiner. Bart Lederman recently noted,

> *"... it has become apparent to me that the next newsletter (September) will be Volume 10, Number 1 of the 'Wombat Examiner'.*
> *"While this might not seem like much compared to some newsletters like the 'Multitasker' (the RSX newsletter), when you consider that some SIGs don't last this long, reaching Volume 10 is something which deserves at least a few words ..."*

Well, Bart, I couldn't agree more. To understand what has been accomplished, we need to review the history of the SIG and the Wombat Examiner.

During the nine and a half years of the SIG, there have been four SIG Chairs. Chuck Watson was chair from the beginning of the SIG in the Spring of 1979 to the Spring 1982 Symposium in Atlanta. Larry Jasmann then took over and was Chair until the Spring 1986 Symposium in Dallas. I was then chair serving from the Spring of '86 to the 1988 Spring Symposium in Cincinnati. Don Stern took over as Chair at the 1988 Spring Symposium; his term of office runs to the spring of 1990.

During that same period, there have been five Newsletter editors. Kathy Tamer edited from the formation of the SIG in the Spring of 1979 to end of 1981. Cynthia Curry-Mealy and Virginia Sventek co-edited in 1982. In '83 and early '84, Virginia edited. In the spring of 1984, I took over as editor and served until the end of 1985. Don Stern then edited from early '86 through the spring of 1988 when I became editor again.

An historical summary of the amount of material published in the Wombat Examiner (and later the Wombat Examiner and 4GL Dispatch) is as follows:

WOMBAT EXAMINER VOLUME NUMBER	NEWSLETTER VOLUME NUMBER	YEAR PUBLISHED	NUMBER OF ISSUES	PAGES PER VOLUME
1		79	1	29
2		80	2	80
3		81	1	53
4		82-83	3	171
5		83-84	4	284
6		85	3	96
7	1	85-86	12	314
8	2	86-87	12	436
9	3	87-88	12	203

The pages per volume fluctuated significantly in Volumes 7, 8, and 9. Volume 8, because of production and mailing costs, was published mostly in landscape (2-up mode) while Volume 9 was published entirely in portrait (1-up) mode. Thus, the change in pages per volume is a reflection in the size of pages as well as the number of pages.

Perhaps the most outstanding and unrecognized contribution to the Wombat Examiner has been the creative efforts of Bart Z. Lederman. He has written 23 original articles for the Wombat Examiner - more than any other author. From November of 1982 to the present he has created 44 unique pieces of wombat art for the cover of the newsletter! The 23 original articles do not include several Wish List articles, Wombat Magic submissions, and many submissions to other DECUS newsletters.

The work of the Wombat Examiner staff was recognized in the Fall of 1986. The period from early 1986 through the end of 1987 when Don E. Stern, Jr. was editor was particularly productive. With Don Stern as editor and Phil

Naecker as the Wombat Wizard feature editor, the quality and quantity of material that was churned out was incredible!

When the newsletter combined into the U. S. Chapter SIGs Newsletters, the opportunity to publish twelve time per year has been very strong encouragement to the editors of the Wombat Examiner and 4GL Dispatch. Only the Wombat Examiner and 4GL Dispatch and the VMS SIG's Pageswapper have not missed a monthly issue since September of 1985 and the Wombat Examiner and 4GL Dispatch contains more pages of original articles than the Pageswapper.

Since its first publication in July of 1979, the most important change in the production of the newsletter has been the creation of a newsletter staff. The publication of twelve issues per year would not be possible without the active contribution of the Production Editor, Steven Cordiviola; the creative efforts of the Associate Editors; and the full support of the SIG Steering Committee. The material published in the newsletter is really the work, ideas, and efforts of many people.

With your interest and support, we will continue to try to make the Wombat Examiner and 4GL Dispatch the best possible newsletter for DATATRIEVE and 4GL users.

Dear Wombat Wizard

Dear Wombat Wizard:

It sure looks like the DATATRIEVE developers are doing something because every time I turn around, our system manager is installing a new version of DATATRIEVE. The problem is that when our system manager installs a new version of DATATRIEVE, DECGraph stops working until I tell him about it and he reinstalls it. What can be done to make our system manager's job easier?

Signed,
"Graphing in Philly"

Dear Graphing:

You're right about the DATATRIEVE developers being busy. One look with my Digital X-Ray binoculars through the Digital windows into their secret Digital New Hampshire facility, will show them toiling endlessly on new code and new features -- burning the the midnight Digital-supplied-through-DECDirect oil.

What I expect your system manager is doing, is giving a new name (actually a different suffix) to each version of DATATRIEVE that gets installed such as DTR3240.EXE (for V4.0) and DTR3241.EXE (for V4.1). DECGraph is linked against the DATATRIEVE shareable image.

But why does he need to reinstall DECgraph but does not need to reinstall other products that use DATATRIEVE such as TEAMDATA?

Well, despite what it says in the Digital Bill of Product Rights, not all products are created equal; some, such as TEAMDATA, are more equal than others. TEAMDATA when it is installed and linked, is linked against the DATATRIEVE shared image that is defined by the logical name DTRSHR. So when a new version, with a different name, of DATATRIEVE gets installed, all the system manager needs to do is make sure that DTRSHR is defined to point to the new DATATRIEVE shareable image. So for upgrading from V4.0 to V4.1 of DATATRIEVE, the system manager issues the command (after DATATRIEVE is installed)

```
    $ DEFINE/SYSTEM DTRSHR SYS$SHARE:DTRSHR41
```

and TEAMDATA automatically now uses the new DATATRIEVE shareable image. DECGraph is not linked against the DATATRIEVE shared image defined by a logical name but against the actual name of the DATATRIEVE shared image such as SYS$SHARE:DTRSHR41.EXE.

But what DECGraph has done wrong, the Wombat Wizard, with some help from those people at Colorado Springs, can fix. What we want to do is to relink DECGraph against DTRSHR so that the logical name redefinition will work like it does with TEAMDATA.

1. Use BACKUP and split up the saveset GRAPH015.A on the distribution tape into individual files. Maybe some command like:

   ```
   $ CREATE/DIR DRIVE1:[VMIWORK]
   $ MOUNT/FOREIGN TAPE0:
   $ BACKUP TAPE0:GRAPH015.A/SAVE DRIVE1:[VMIWORK]
   ```

2. Edit KITINSTAL.COM and put in the following somewhere:

   ```
   $ ls:= write link$scratch
   $ ls "$  on error then continue"
   $!  Next line suggested by Wombat Examiner article
   $ ls "$DEFINE/SYSTEM DTRSHR ´´graph$dtr_shareable´"
   .......
   $ ls "CLUSTER=LBRCLUSTER,,,VMI$ROOT:[SYSLIB]LBRSHR.EXE/SHARE"
   $ if graph$link_with_dtr-
   $!  Next line suggested by Wombat Examiner article
   $ then ls "CLUSTER=DTRCLUSTER,,,DTRSHR/SHARE"
   ```

3. Now you put it all back in the saveset with something like:

   ```
   $ PURGE DRIVE1:[VMIWORK]
   $ INIT TAPE0: GRAPH
   $ MOUNT/FOREIGN TAPE0:
   $ BACKUP DRIVE0:[VMIWORK]*.* TAPE0:GRAPH015.A/SAVE
   ```

4. Reinstall DECGraph with the new saveset.

Now DECGraph doesn't care when you install a new versions of DATATRIEVE with different names as long as you change the logical name DTRSHR.

Signed
The Wombat Wizard (BR)

Dear Wombat Wizard:

I have a table look up which always takes the "ELSE" condition. My table is a table of employee identification numbers and employee names of the following form:

```
DEFINE TABLE LOOKUP_TABLE
0001 : "John Doe"
. . .
0230 : "Sam Smith"
. . .
ELSE "Invalid employee number"
END_TABLE;
```

When ever I try to get the employee's name via the table with code like

```
PRINT ... , EMPLOYEE_ID VIA LOOKUP_TABLE, ...
```

It always prints "Invalid employee number" if the employee number is less then 1000. WHY?

Signed,
Lost under a table

Dear Lost:

DATATRIEVE recognizes the difference between text strings which are actually numbers and strings which contain characters which are not a valid part of a number. For strings which are actually numbers it strips the leading zeroes in its internal format; it is the internal format which is used in a table lookup. However, DATATRIEVE does not strip these leading zeroes in the table description. For the employee id "0230", DATATRIEVE treats this as "230" which is not contained in the table; thus it never matches.

To illustrate this situation, consider the following test table:

```
DTR> show test-table
TABLE TEST_TABLE
"1":"one",
"01":"zero one",
" 1":"space one",
"1 ":"one space"
ELSE "not in table"
END_TABLE
```

Then, display a variable via this table using various formats.

```
DTR> DECLARE TEST-VARIABLE USAGE IS INTEGER.
DTR> TEST-VARIABLE = 1
DTR> PRINT TEST-VARIABLE VIA TEST-TABLE
one
DTR> PRINT FORMAT TEST-VARIABLE USING Z9 VIA TEST-TABLE
space one
DTR> PRINT FORMAT TEST-VARIABLE USING 99 VIA TEST-TABLE
zero one
```

This example illustrates how one can control the format of the numeric string before it is used in a table lookup. If one does not explicitly control the numeric string's format, then the table entry must have its leading zeroes removed for the entry to be found.

Signed
The Wombat Wizard (HR&JG)

PIR Ballots Analysis and Digital's Responses

T. Chris Wool, PIR Coordinator, du Pont, Wilmington, DE
John Carpenter, Counterpart, Digital Equipment Corporation, Nashua, NH

Product improvement requests (PIR's) are the process by which users give feed-back to Digital regarding improvements in products. At the 1987 Fall Symposium in Anaheim and by mail, users submitted requests for improvement. In the March 1988 issue of the newsletter (Volume 3, Number 7), all PIR's and a ballot were published.

The statistical analysis of the ballots for all 28 product improvement requests for VAX-DATATRIEVE includes the responses of 36 users. The PIRs are ordered by decreasing total points given. Also listed are the percent of ballots with positive points given, the percent of ballots with negative points given, the average points given, and the standard deviation of the points given. See the article in the March 1988 newsletter for a description of how points could be balloted for or against a PIR. The analysis is as follows:

VAX DATATRIEVE PIRs

			Pct of	Pct of		
PIR		Total	Ballots	Ballots	Avg Pts	Std Dev
No.	VAX PIR Description	Pts	Pos Pts	Neg Pts	Given	of Pts
17	Enhance DATATRIEVE to support wild card searches in the CDD.	218	69.4	0.0	8.7	2.4
20	Enhance DATATRIEVE to include the ability to trap DATATRIEVE, CDD or other VMS errors in a procedure and then do exception processing.	185	66.7	0.0	7.7	3.1
19	Enhance DATATRIEVE to allow multiple responses to a single prompt.	173	58.3	0.0	8.2	2.9
22	Enhance DATATRIEVE to show why a store failure occurs.	155	58.3	0.0	7.4	3.6
6	Support segmented keys in RMS files.	128	41.7	2.8	8.0	4.5
8	Provide a way to read a CDD object while editing a CDD object.	73	36.1	0.0	5.6	3.5
30	Implement a more generalized scheme for establishing the textual SET variables in the Report Writer especially SET REPORT-NAME, but SET DATE and SET NUMBER should also be included.	64	33.3	0.0	5.3	3.3
18	Enhance the Report Writer with an option to print fields side-by-side, separated by only one space ("dense" mode).	58	30.6	0.0	5.3	2.6
12	Enhance DATATRIEVE to call an editor (TPU in particular to edit the contents of a field.	53	25.0	0.0	5.9	2.1
23	DATATRIEVE graphics should support Tektronix 4014 CRT´s and HP-GL compatible plotters (e.g., DEC LVP16, HP7475, etc).	53	16.7	0.0	8.8	2.9
33	Implement a screen management system internal to DATATRIEVE which uses calls to the SMG library and replaces FMS and TDMS.	49	19.4	0.0	7.0	3.9
11	Enhance the DATATRIEVE Report Writer by adding a LINE-COUNT build in key word.	48	27.8	0.0	4.8	2.3
31	Establish a new variable in the Report Writer which is the page number.	46	33.3	0.0	3.8	1.4
10	Provide ability to specify COMPUTED BY fields within a view definition.	45	16.7	0.0	7.5	3.3
32	Implement a plot in DATATRIEVE which can use the time part of a date-time variable.	39	22.2	0.0	4.9	3.4
7	Enhance DATATRIEVE to handle DELTA DATES.	38	16.7	2.8	5.4	4.8
26	Enhance the way in which DATATRIEVE evaluates the logical DTR$STARTUP to include the check (and evaluation) in system, group, and process logical name tables or search lists.	34	16.7	0.0	5.7	2.9
25	DATATRIEVE needs to be able to handle both signed and unsigned variables in order to be compatible with DBMS.	32	16.7	0.0	5.3	2.8
9	Allow normal order of statement execution execution for indirect command files in procedures.	31	16.7	0.0	5.2	2.8

VAX DATATRIEVE PIRs (cont.)

PIR No.	VAX PIR Description	Total Pts	Pct of Ballots Pos Pts	Pct of Ballots Neg Pts	Avg Pts Given	Std Dev of Pts
28	Add a function to DATATRIEVE to calculate the MEDIAN of a value expression of a record stream.	29	19.4	0.0	4.1	1.3
24	DATATRIEVE should support a more generalized construction of record selection expressions.	27	11.1	0.0	6.8	2.4
29	Implement a more generalized scheme for the prompt for input. That is, allow the prompt string to be constructed from the concatenation of multiple strings.	27	19.4	0.0	3.9	2.9
16	DATATRIEVE should have an "intelligent", more English-like front end that does not require such a restrictive (stilted) syntax to work.	26	13.9	0.0	5.2	4.4
14	Enhance the READY command to include a RETRY clause.	23	13.9	0.0	4.6	3.5
27	Add a function to allow attaching to a process or sub-process.	10	2.8	0.0	10.0	0.0
21	Enhance the DATATRIEVE/FMS interface to access supervisor mode.	2	2.8	0.0	2.0	0.0
13	Allow the storage of DBMS junction records via DATATRIEVE procedures	1	2.8	0.0	1.0	0.0
15	Guide mode should have an option to clear the screen (e.g., to work anywhere on the screen).	0	0.0	0.0	0.0	0.0

Ballots for VAX-DATATRIEVE PIR's were analyzed by All Submittors, Business EDP/MIS Application Type, Education Application Type, Engineering/Scientific Application Type, Office Automation Application Type, Software Development Application Type, VAX Users, PDP-11 Users, VAX DATATRIEVE Users, DATATRIEVE-11 Users, and DECreporter Users. However, the small numbers in each subcategory and the relative uniformity of ordering does not warrant the publication of the complete analysis by category of user.

Only a comparison of how each subgroup ranked the VAX-DATATRIEVE PIRs follows:

	ALL	VAX	DTR32	Eng	Dev	EDP	OA	DTR11	PDP	Edu
Responses	36	34	34	23	23	21	16	9	6	4
Ranking										
1	17	17	17	17	17	20	17	17	10	17
2	20	20	20	20	20	17	20	30	9	8
3	19	19	19	19	19	22	19	8	30	11
4	22	22	22	22	6	19	6	1	31	19
5	6	6	6	6	22	6	22	26	12	18
6	8	8	8	10	8	8	10	10	18	16
7	30	30	30	30	30	33	12	31	17	20
8	18	18	18	23	18	7	18	33	19	28
9	12	12	12	26	10	30	25	9	20	31
10	23	23	23	8	12	11	30	16	26	25
11	33	33	11	18	33	18	8	19	28	12
12	11	11	31	12	25	12	7	29	8	29
13	31	31	10	32	31	23	32	12	24	9
14	10	10	33	31	26	31	11	32	32	30
15	32	7	7	9	9	16	29	11	25	7
16	7	26	26	25	11	14	31	22	11	23

	ALL	VAX	DTR32	Eng	Dev	EDP	OA	DTR11	PDP	Edu
Responses	36	34	34	23	23	21	16	9	6	4
Ranking										
17	26	25	25	33	24	25	33	24	23	10
18	25	9	9	24	32	29	26	28	22	33
19	9	28	28	7	7	10	28	20	29	6
20	28	32	32	28	23	27	14	23	14	13
21	24	24	24	11	29	32	16	14	7	14
22	29	29	29	29	28	26	21	25	13	15
23	16	16	16	14	14	9	9	7	15	21
24	14	14	14	13	16	24	13	13	16	22
25	27	27	27	16	21	28	15	15	21	24
26	21	21	21	15	13	21	23	21	27	26
27	13	13	13	21	15	13	24	27	33	27
28	15	15	15	27	27	15	27	6	6	32

All - All Users
VAX - VAX Users
DTR32 - VAX DATATRIEVE Users
Eng - Engineering/Scientific Application Type
Dev - Software Development Application Type
EDP - Business EDP/MIS Application Type
OA - Office Automation Application Type
DTR11 - DATATRIEVE-11 Users
PDP - PDP-11 Users
Edu - Education Application Type

From the number of responses in each user category, it is clear than an individual ballot has been counted in more than one category.

Digital's responses to the Top-10 VAX-DATATRIEVE PIR's are:

VAX-DATATRIEVE

PIR
No. PIR Description / Engineering Response

17 Enhance DATATRIEVE to support wild card searches in the CDD.

We find this an extremely interesting suggestion. We will strongly consider implementing this enhancement in a future release.

20 Enhance DATATRIEVE to include the ability to trap DATATRIEVE, CDD or other VMS errors in a procedure and then do exception processing.

Unfortunately, the current architecture of DATATRIEVE as an interpretive language precludes this enhancement. We will take this suggestion to heart for any future architecture changes or enhancements in this product area.

19 Enhance DATATRIEVE to allow multiple responses to a single prompt.

We find this an interesting suggestion. We will consider implementing this enhancement in a future release.

22 Enhance DATATRIEVE to show why a store failure occurs.

We think that this is an excellent idea that will be very informative for users who do their store from a procedure. We will keep this enhancement high on our list of priorities.

6 Support segmented keys in RMS files.

This enhancement would be extremely difficult to implement although we recognize that it would provide a high payback to our customers. We will consider implementing this request in a future release.

8 Provide a way to read a CDD object while editing a CDD object.

> *Since DATATRIEVE spawns EDT, TPU, or LSE, this enhancement cannot be done under the current DATATRIEVE architecture.*

30 Implement a more generalized scheme for establishing the textual SET variables in the Report Writer especially SET REPORT–NAME, but SET DATE and SET NUMBER should also be included.

> *We consider this a valuable suggestion. We will consider implementing this enhancement in a future release.*

18 Enhance the Report Writer with an option to print fields side–by–side, separated only by one space ("dense mode").

> *This enhancement would be tricky to implement. However, we will consider implementing this request in a future release.*

12 Enhance DATATRIEVE to call an editor (TPU in particular) to edit the contents of a field.

> *The current DATATRIEVE architecture does not lend itself to this style of field manipulation, and therefore we will file this suggestion for future thoughts. However, it is possible for this type of capability to be achieved by writing a DATATRIEVE application program to have DATATRIEVE obtain the data, and have the callable program do the field manipulation.*

23 DATATRIEVE graphics should support Tektronix 4014 CRT's and HP–GL compatible plotters (e.g. DEC LVP–16, HP7475, etc.).

> *There are currently no plans to implement this enhancement.*

Although PIR ballots were submitted for RALLY, TEAMDATA, DATATRIEVE–11, and DECReporter, the numbers of responses do not justify publication of the statistical analysis.

Digital's responses to PIRs for DATATRIEVE–11 and DECReporter are:

DATATRIEVE–11

PIR
No. PIR Description / Engineering Response

36 Enhance all versions of DATATRIEVE–11 to use a logical which controls the default open mode for READY commands.

> *Due to base system architectural restrictions, this is not feasible for DATATRIEVE–11. RSX limits the use of logical names to device specifiers while RSTS limits them to device specifiers with an optional directory specification. VMS, however, allows any arbitrary string to be assigned to a logical. Hence, it would not be possible to assign a READY mode to a logical on 11's in the way that VAX DATATRIEVE allows.*

37 Change DATATRIEVE–11 so the callable DATATRIEVE may be used without DECnet being installed.

> *This restriction was removed in DATATRIEVE–11 Version 3.2. DECnet is no longer required in order to make use of callable DATATRIEVE.*

38 Change DATATRIEVE–11 to search for and manage the startup file, QUERY.INI, in a more generalized way.

> *This is a reasonable request which we may consider for a future release of DATATRIEVE–11.*

DECreporter

PIR
No. PIR Description / Engineering Response

40 DECreporter needs to have the same capability as DATATRIEVE to do a print FIELD–NAME via TABLE–NAME.

> *There are currently no plans to implement this enhancement.*

The publication of this article is the completion of the 1988 PIR process; it is time to starting working on the 1989 product improvement requests. In the back of the newsletter you will find the DATATRIEVE/4GL SIG's Product

Improvement Request Submission Form. You may mail in a PIR to Chris Wool at the indicated address or turn in a PIR at the Fall Symposium in Anaheim. The cutoff date for PIRs will be at the end of 1988 or very early in January of 1989. All PIRs for all products (VAX-DATATRIEVE, DATATRIEVE-11, RALLY, TEAMDATA, and DECReporter) will be published along with a ballot in the spring of 1989 (possibly the March 1989 issue). Ballots will be accepted through the end of the Spring 1989 Symposium.

Exercise your influence on Digital. Participate in the PIR process.

Contest at Wombat Magic in Anaheim

Joe H. Gallagher, Ph. D., 4GL Solutions, Overland Park, KS

At Wombat Magic in Anaheim a contest will be held and prizes given for the best solution to each of the following problems. You do not have to be present to win, but your solutions must be presented at the Magic Session. So, if you can not attend the Anaheim Symposium, send your solutions to someone who will attend and will present your solution. This contest is open to anyone (DECUS member or not) and it is open to Digital employees. All contestants are restricted to use a released version of DATATRIEVE. The judging will be done by the regular Magic judging committee whose decision is final.

Problem 1 - VAX-DATATRIEVE - Sorted Extremum of a Class Problem

At the recent Symposia in Cincinatti, a user asked for help on a VAX-DATATRIEVE problem. He works in an Information Center in a company which had several personal computers to loan out to employees. There were several different units of several different types of machines. His problem was to find which unit was available to be loaned out FIRST for each type of machine and when it could be loaned and present then in the order of first available. The following test data illustrates the kink of data he has.

```
DTR> show loaners, loaners-record
DOMAIN LOANERS USING LOANERS_RECORD ON LOANERS.DAT;
RECORD LOANERS_RECORD USING OPTIMIZE
01 LOANERS_REC.
03 MACHINE.
05 MODEL PIC X(6)
VALID IF MODEL = "PC","XT","AT","COMPAC","MAC","PRO350".
05 SERIAL_NUMBER PIC X(4).
03 OUT_DATE USAGE IS DATE.
03 RETURN_DATE USAGE IS DATE.
03 FLAG PIC X. ! field required for Gallagher's solution, but
! which CAN NOT be used for a contest solution.
;
DTR> ready loaners
DTR> print machine, out-date, return-date of loaners
```

```
                         SERIAL       OUT          RETURN
MODEL                    NUMBER       DATE          DATE

 AT                      AT01      1-Jul-1988    5-Jul-1988
 AT                      AT01      6-Jul-1988    8-Jul-1988
 AT                      AT01     10-Jul-1988   12-Jul-1988
 AT                      AT02      9-Jul-1988   11-Jul-1988
MAC                      MAC1     30-Jun-1988    5-Jul-1988
MAC                      MAC1     12-Jul-1988   14-Jul-1988
MAC                      MAC2      9-Jul-1988   13-Jul-1988
 XT                      XT01     28-Jun-1988    6-Jul-1988
 XT                      XT01      8-Jul-1988   12-Jul-1988
 XT                      XT02      5-Jul-1988    9-Jul-1988
```

My solution to his problem was to use a 3GL-like approach and is given in the following procedure NEXT-LOAN.

```
DTR> show next-loan
PROCEDURE NEXT_LOAN
READY LOANERS SHARED MODIFY
DECLARE T_SERIAL_NUMBER PIC X(4).
DECLARE T_MODEL PIC X(6).
T_SERIAL_NUMBER = " "
T_MODEL = " "
FOR LOANERS SORTED BY MODEL, SERIAL_NUMBER, DECREASING RETURN_DATE
BEGIN
      IF (T_SERIAL_NUMBER EQ SERIAL_NUMBER AND T_MODEL EQ MODEL) THEN
            BEGIN
                  MODIFY USING FLAG = "N"
            END ELSE BEGIN
                  MODIFY USING FLAG = "Y"
            END
      T_SERIAL_NUMBER = SERIAL_NUMBER
      T_MODEL = MODEL
END
T_MODEL = " "
FOR LOANERS WITH FLAG="Y" SORTED BY MODEL, INCREASING RETURN_DATE
BEGIN
      IF (T_MODEL EQ MODEL) THEN
            BEGIN
                  MODIFY USING FLAG = "N"
            END
      T_MODEL = MODEL
END
FOR LOANERS WITH FLAG="Y" SORTED BY RETURN_DATE
BEGIN
      PRINT RETURN_DATE, MODEL, SERIAL_NUMBER
END
END_PROCEDURE
DTR> :next-loan
```

```
  RETURN                 SERIAL
   DATE        MODEL     NUMBER
 9-Jul-1988      XT       XT02
11-Jul-1988      AT       AT02
13-Jul-1988     MAC       MAC2
```

The contest for this problem is for someone to come up with the best solution which is funtionally equivalent to the procedure NEXT-LOAN above, but which uses a 4GL-like way of solving this problem WHICH DOES NOT USE a flag variable but uses some combination of READY LOANERS READ, FORs, CROSS, REDUCE, SORT(ED), etc. You also may not use a function (fn$something) to solve the problem; that is, it must be done within DATATRIEVE.

Problem 2 - DATATRIEVE-11 - Date Formatting

VAX-DATATRIEVE has considerably more features than DATATRIEVE-11. The features which are most often missed are FORMAT ... USING, CHOICE OF, and more pool space. However, just because one uses DATATRIEVE-11 it doesn't mean that one doesn't need some of the functionality of these features.

The problem is to come up with the best solution in DATATRIEVE-11 which is functionally equivalent to the following VAX-DATATRIEVE code.

```
DTR> declare day usage is date.
DTR> declare day-string pic x(20).
DTR> day = "today"
DTR> day-string = format day using m(9)byyyy
```

The only restriction is that it MUST work inside of a procedure (but it doesn't have to work within a BEGIN-END block) and there can be additional code before it and after it.

Wombat Magic, Spring 1988 - Part 2

Session Chair: Dick Azzi, Motorola, Phoenix, AZ
Session Editor: Joe H. Gallagher, Ph. D., 4GL Solutions, Overland Park, KS

Editor's note: The following is Part 2 of a highly edited transcription of the audio tape of the Wombat Magic Session at the 1988 Spring DECUS Symposium in Cincinatti, Ohio, which occurred on May 19, 1988. Part 1 appeared in last month's issue. Material which was presented on transparencies has been merged into the oral presentation. An attempt has been made to convey both the technical content of the Magic Session as well as the humor, covert intellectual swaggering, and the spirited interchange of the presentations. Material which appears in the text with square brackets [] has been added by the editor in an attempt to improved the understandability of this very exciting Magic Session. Special acknowledgement is given to Mary Gallagher who assisted with the transcription of the audio tapes.

Joe H. Gallagher, 4GL Solutions, Overland Park, KS

[Continuing Joe Gallagher's Magic presentations of DATATRIEVE functions which was begun in Part 1.]

There are times when you need a random number generator and if you attempt to directly attach DATATRIEVE to the FORTRAN random number generator there is a problem because DATATRIEVE functions [like all proper functions] can only return one variable. The FORTRAN callable function returns a couple of arguments so you can't link directly with it, and you need a little FORTRAN program sitting in between [them]. So you need a little piece of FORTRAN code [which is] link[ed] through [the] DTRFND.MAR linkage. And you will notice that [this] routine checks to see if it has been loaded before, and if it has not, it checks [reads] the clock, picks up bits out of there, and seeds itself and takes off. You have an automatic seeding random number generator for DATATRIEVE.

```
! fn_random.for
!  a self-initializing, FORTRAN interface for DATATRIEVE to the
!  VAX random number generator.
!
real*4 function fn_random()
real*4 mth$random
integer*4 quad(2), seed
equivalence (quad, seed)
   if (seed .eq. 0) then
      call sys$gettim(quad)
   end if
   fn_random = mth$random(seed)
return
end
```

The macro linkage in the DTRFND.MAR file is:

```
;
; FN$RANDOM- a pseudo-random number generator
;
; output is an F-floating pseudo-random number 0<= x < 1
;
$DTR$FUN_DEF FN$RANDOM, FN_RANDOM, 0
$DTR$FUN_OUT_ARG  TYPE = FUN$K_VALUE, DTYPE = DSC$K_DTYPE_F
$DTR$FUN_NOOPTIMIZE
$DTR$FUN_END_DEF
```

And example of its use is:

```
DTR> repeat 5 print fn$random
   FN$RANDOM
  5.0153E-01
  1.5545E-01
  9.6883E-01
  1.8529E-01
  6.9579E-02
```

There are restrictions on strings in DATATRIEVE. If one does some string manipulations, ... [there are] places in DATATRIEVE that automatically concatenate the strings to 31 characters. I want to manipulate a strings that is considerably longer -- like a string of a thousand characters and be able to assert things in the middle. There is a library function available in VMS and so this [is the] little interface to that replace function. ... Those of you that are heavy readers of the Wombat Examiner and 4GL Dispatch will recall an article that I wrote on browsing under program control; that particular function is used.

```
;
; FN$STR_REPLACE- String replace
;
; output is a string
; input is an output string descriptor,
; an input string descriptor
; a starting position in the input string
; an ending position in the input string
; the replacement string
$DTR$FUN_DEF FN$STR_REPLACE, STR$REPLACE, 5
$DTR$FUN_OUT_ARG  TYPE = FUN$K_STATUS
$DTR$FUN_IN_ARG  TYPE = FUN$K_TEXT, OUT_PUT = TRUE
$DTR$FUN_IN_ARG  TYPE = FUN$K_DESC, DTYPE = DSC$K_DTYPE_T, ORDER = 1
$DTR$FUN_IN_ARG  TYPE = FUN$K_REF,  DTYPE = DSC$K_DTYPE_L, ORDER = 2
$DTR$FUN_IN_ARG  TYPE = FUN$K_REF,  DTYPE = DSC$K_DTYPE_L, ORDER = 3
$DTR$FUN_IN_ARG  TYPE = FUN$K_DESC, DTYPE = DSC$K_DTYPE_T, ORDER = 4
$DTR$FUN_END_DEF
```

An example of how this might be used is:

```
declare COUNTER usage is integer.
declare TAG pic X(1000).
ready PART2DB
      for PART2DB with INTEREST_CODE starting with ITEM_CODE_PART1||""
      begin
            COUNTER = COUNTER + 1
            TAG = FN$STR_REPLACE(TAG,10*COUNTER - 9,10*COUNTER, PART2DB_CODE)
      end
! TAG now contains the keys of the records of interest
```

[Joe Gallagher's last DATATRIEVE functions is a FORTRAN functions that may be used to edit the contents of a field in DATATRIEVE. The FORTRAN code is about 11 pages and is too long to be included here. That material will be turned into an article in a future issue of the Wombat Examiner and 4GL Dispatch.]

Jeff Chimene, Tuscon Archdiocese, Tuscon, AZ

My name is Jeff Chimene and I work for the Pope in Tuscon. This is how to do VMS style message in DATATRIEVE for all of you who like that technique. This consists of a couple of steps. The first step is to define a table and on the left side you see the various mnemonics and on the right side you see their equivalent strings.

```
DTR> SHOW MESSAGES
TABLE MESSAGES
      SELNOTVAL : 'MNU-E-SELNOTVAL, Menu selection not valid'
      SELNOTIMP : 'MNU-F-SELNOTIMP, Menu selection not implemented'
      MENUEXT   : 'MNU-S-EXIT, Menu successful exit'
      . . .
ELSE          '?Unknown message'
END_TABLE
```

Of course if you don't like that technique you can specify whatever you want, but the trick here is to specify the facility ... then you can define the severity and then the mnemonics again and then the text. At this point is a demonstration of how it works.

Here I say

```
DTR> DISPLAY "SELNOTVAL" VIA MESSAGES
      DISPLAY: MNU-E-SELNOTVAL, Menu selection not valid
DTR> ABORT "SELNOTVAL" VIA MESSAGES
      ABORT: MNU-E-SELNOTVAL, Menu selection not valid
```

... I think you get the idea. Anyone not see what I'm doing here? I want to keep it short. All got the idea? OK.

So, that's basically it and then Don [Stern] suggested a second technique down here where you actually declare a variable called EXIT_STATUS.

```
DTR> DECLARE EXIT-STATUS PIC X(9).
DTR> EXIT-STATUS = "SELNOTVAL"
DTR> ABORT EXIT-STATUS VIA MESSAGES
      ABORT: MNU-E-SELNOTVAL, Menu selection not valid
DTR> DISPLAY EXIT-STATUS VIA MESSAGES
      DISPLAY: MNU-E-SELNOTVAL, Menu selection not valid
```

Then you assign the variable name or assign the mnemonic over to ... , then you are able to generate an exit handler. OK. The idea would be to have an exit handler that would do the abort and come back. The whole thing here was to have a consistent message facility and of course you can extract the table out of the dictionary, sort it and do all that stuff. [Messages are contained in one maintainable place. And system documentation is improved.] That's it.

Karnac the Magnificent

Dick Azzi exclaims, "I think I hear somebody coming. If we're lucky... . It is ... the great ... the great one himself." [Karnac enters in a costume of a black cape with a blue satin lining, and a garish turban with jewels and feathers. Karnac is greeted and assisted by his trusty companion and straight man, Bert "Ed McMahan" Roseberry.]

Roseberry: Welcome, Karnac the Magnificent.

Karnac: Thank you, dog breath.

Roseberry: What we have here, Karnac, are the envelopes which have been sealed in a Mayonnaise jar which has been outside Ken Olsen office since early yesterday. And you are going to divine the answers never before having seen the questions.

Karnac: The first envelope, please. ... Without my glasses I can't read them. I really can't read them. I can't read them! [Laughter] [Karnac's glass are brought to him.] Karnac has gotten two years older and a lot blinder since the last time I did this. [Laughter] And I had trouble the last time. My arm is still to short. ... The answer to the first question is:

A: All our specialists are busy at this time. May we have a specialist call you back?
Q: What is the fourth biggest lie? [Laughter]

A: Ready OR_NOT, HERE_I_COME.

Q: What do you have to do before you can CROSS OR_NOT with HERE_I_COME? [Laughter followed by growns]

Karnac: May you go to work in a DATATRIEVE-20 group and have to live off of the royalties. [Laughter]

A: The Combat Zone, Times Square, and Mammoth Street in Newport, Kentucky.
Q: What are Bert's favorite places? [Laughter]

A: Wesson, Quaker State, and UNIX. [Laughter]
Q: Name a vegetable oil, a motor oil, and a snake oil. [Laughter and applause]

Karnac: Oh, we can't use this. Mary Ann would kills us. It violates non-disclosure. [Laughter]

[The real joke was to see several Digital employees dive after the contents of the envelope after Karnac tossed it to the floor. The question was "Got'ch".]

A: APL, TECO, and Krishnam
Q: Name three religious experiences. [Laughter]

A: Superman and a microVAX I.
Q: Name something that is faster than a speeding bullet and something that isn't. [Laughter]

A: AIDS, MUMPS, and Latin [Laughter]
Q: Name a deadly disease and two dead languages. [Laughter]

Roseberry: And the very last one. [Cheers and applause.]

Karnac: Make a sick yak visit your microprocessor.

[Karnac gets in such a hurry that he begins to open the envelope before he has given the answer.]

A: Two – One was Ken Olsen and the other was the wrong number.
Q: How many people have gotten through to Colorado Support services on the first call? [Laughter and applause].

Ann F. Kah, The Glidden Company, Strongsville, OH

My name is Ann Kah from the Glidden Company and this is a low level magic. It doesn't have a dollar sign or function in the whole thing; this is straight ol' DATATRIEVE.

We've got people who have panels out for exposure on a field. We make paint and they paint the things and put them out in the field and they've been doing this since dinosaurs walked the earth and they've got a thousand notebooks and stuff and they are not willing to part with any of it. And you say you don't have to store all of this. OH YES, we have to store all of this junk. So, we have to describe the panels, so I set up a domain [record] that [looks like]

```
RECORD PANEL
01 PANEL-REC.
02 CODE ...
02 LEVEL ...
02 DATA-FIELD ...
02 SUBSTRATE-DATA REDEFINES DATA-FIELD.
04 SUBST ...
04 FILLER ...
04 PRETREAT ...
04 FILLER ...
...
```

... and [when the record is stored] it asks all the proper questions ... .I can use just a plain ol' procedure, no functions, no nothing. Ask them all the proper questions. If they say it's level zero, I ask then sub-straight questions. If they say its level 1 through 4, I ask them coding questions. If they say it's levels 5 through 9, its a comment and they can put in any garbage they like – and they do. Because Joe Slow back in 1961 wrote around the little holes in the 3-ring paper and they recognize things. That's the one with the green ink on the left hand side and so forth. So we've got all that garbage and it puts in a filler and it puts in another field and it puts in a filler. I can put in units when there's numerical data and it can put in a filler, all this garbage gets stored in there. Now since this is a disaster time it is not even a VAX, this is a PDP-11, we can't use all this junk because we can't handle all this junk, so we need to redefine a second thing, some data, redefine a second domain that just says this is a string, this is 60 characters of string. I don't care what's in it anymore. If you don't think that's bad enough we put it in a view and its down at the bottom of the hierarchy. So there's some garbage deeply buried and invisible and hidden. One will never ever see it again. They can look at the panel domains and call up separate fields for separate types of reports, but when they're done with this what they want is the exposure results. This thing has been out on a fence for 27 years under a pigeon's nest and they want to know what this stuff looks like. They want this stuff up in the corner that says this is what it is. And they want this whole page. If I formatted all those fields to explain what they all were the results are over on page 4. They don't want that. It looks like English cause I put in a filler, I put in units and stuff like that and so when I want to report it, I say

```
REPORT ...
AT TOP OF EXPOSURE-STUFF PRINT ALL LEVEL, DATA-FIELD OF PANEL-ID
...
```

And the results of that is in the top 2 or 3 or 4 or 5 lines, however many they filled in. You have this garbage, which looks like garbage to you and looks like garbage to me.

```
         0   ALUMINUM  X23J     FODCO   FLAT
         1   ALKYD     C29-599  GLIDDEN
         2   ACRYLIC   509-R    GLIDDEN
         5   THIS IS JUST A COMMENT LINE
         EXPOSURE      GLASS    CHALK     COLOR      CORROSION
     TIME

         -----         ----     ----      -----        ------
         -----         ----     ----      -----        ------
         -----         ----     ----      -----        ------
     ...
```

Part of the miracle is 1) they can read it 2) they're happy with it, because all they really want is all this exposure stuff which is at the top of the view. No sweat to put that out in a report and we park all that stuff up there in pseudo-English, right at the top. It went in as margin area, it came out as margin area, in between times DATATRIEVE has verified fields, lined it all up, asked the proper questions, without me having to write any other type of procedures except the DATATRIEVE thing and out it comes and they're happy with it. It pretends to be English and everybody's delighted - and amazed.

Dana J. Schwartz, Department of Defense, Washington, DC

Just got in another Top 10 list. *[Editor's note. To get the full effect, you need to cover up the list with a piece of paper and uncover them one at a time.]* This one came for the home office in Sidney, Australia.

```
                 Top 10 DATATRIEVE User Complaints

10.   The NULL process is hogging all the CPU time. [Laughter]
 9.   Can't do "Inner Join". [ooh]
 8.   Don't know what "Inner Join" is. [Laughter]
 7.   Don't have one of those shirts with a rat on it. [Laugher]
 6.   AT TOP OF PAGE PRINT NEW-PAGE doesn't work. [Laughter]
 5.   Doesn't have IF-THEN-BUT construct. [ooh]
 4.   Can't find any other DTR-20 users. [a few snickers]
 3.   No index in DECReporter internals manual. [ooh]
 2.   It's not compiled. [ooh]
 1.   It's not compiled. [ah, ha, ha and applause.]
```

Thomas R. Fournier, Puget Sound Power & Light Co., Bellevue, WA

I had a situation arise where I need to have an output on the terminal and it would just list some data selected by various means, but unfortunately the terminal I had to present it on belongs to a vice-president and they have strange terminals, they don't have smooth scroll. They have these $5,000 things that look sort of like a phone and sort of like a terminal and when you try to scroll data past it it kind of goes whoosh! So what you need to do is you need to fill up the screen and then stop that sucker so you can look at it.

```
DECLARE COUNTER USAGE IS INTEGER.
DECLARE DUMMY PIC X.
COUNTER = 0
FOR ALL FOO WITH rse ...
BEGIN
     COUNTER = COUNTER + 1
     PRINT FLD1, FLD2, FLD3, . . .
     IF (FN$MOD(COUNTER,23) EQ 0) THEN
     BEGIN
          DUMMY = *."any character to continue"
     END
END
```

So there's a way to do it if you want to declare a couple of variables. One is just a counter and the other is a dummy variable and the key to this is just because you read something in doesn't mean you really have to use it. So what you do is you have your loop and an expression here ... and you have a counter in it and you print your line and you can use the FN$MOD routine [function] and count 23 lines and anytime that hits, there you can ask for dummy with a star and type any character in to satisfy dummy and that will bring up next screen.

Can I do another one? The other thing I've run into from time to time is that I'd really like to have a single dimension array. But there's no such beast. But you can sort of fake it, if it's really important to you. This is handy if you want to say read information from 10 records and display 'em on one form or any other reason. But you can,

```
DEFINE ARRAY PIC X(80).
DEFINE COUNTER USAGE IS INTEGER.
DEFINE FLD PIC X(8).
ARRAY = ""
...
FOR FIRST 10 FOO
BEGIN
     ARRAY = FLD | ARRAY
END
COUNTER = 10
WHILE COUNTER GT 0
BEGIN
     PRINT FN$STR_EXTRACT(ARRAY,(COUNTER*8 - 1), 8)
     COUNTER = COUNTER - 1
END
...
```

... Define a string like an array. You have another counter here and the first thing you do is set your array on [equal to a] null-string and then suppose we had a field of 8 characters long you want to read in. You just read it in in a loop and each time you concatenate field to the front of the array. What happens is the last 8 blanks at the other end just kind of fall out. So if you read 10 of 'em you have this string that's got 80 characters in it and then if you want to get them out again you can use your fn$str_extract and it calculates where you want to start. In this case you take a counter times 8 minus 7 and go for 8 characters and you can pick up individual bits. The reason this looks maybe a little screwball is because you have to remember the way you concatenated things on the way in, the first one you read is on the right, its all the way over, but you can use that for various things. This is a trivial application because I just printed it, but it works for other things if you need it for calculations. [Editor's note. Tom's magic only works for short strings. If the strings are longer than 31 characters, then the implied edit–string on the concatenation causes problems. See Joe Gallagher's magic at the first of this article for a function which will work for strings longer than 31 characters.]

Just one hint, this isn't strictly DATATRIEVE but anybody that's worked with forms, be it in a DATATRIEVE application or WHIPS or ALL–IN–1. You've got a long field [in a form] and you're about 60 characters up and you notice you spelled week with 3 e's, and you're in OVERSTRIKE mode and you don't really want to type all that over again. Did you know if you go back and hit GOLD PF3 now you're in INSERT mode. If you want to get back to overstrike mode, hit the PF3 key again. [Editor's note. Both FMS and TDMS have internal field editing capabilities. The use of the left arrow and right arrow, <LF>, <DEL>, <BS>, <TAB>, and INSERT or OVERSTRIKE mode can greatly ease the burden of data entry in one of the forms products.]

Harry J. Miller, Ontario Police Department, Ontario, CA

I'm Harry Miller from the Ontario Police Department. [From the audiaence] Oh, Oh! Nobody leaves! [Laughter] For those of you who aren't in the know, Ontario is not north of the Great Lakes, it's 60 miles east of Los Angeles.

Dick Azzi, the session Chair interrupts, "Before you start can I tell a quick story?" Harry replys, "Oh, I'd be delighted Oh Great One." Dick Azzi continues, "The last time we had a police officer at Wombat Magic, that was ... I think it was in Miami and we had a great, great Magic presentation showing how to run a bookie operation that this guy had, and the guy gave the whole magic and the police officer got up next and said he would be very interested in finding out more about that and could the man please contact him right after the session. It was rather interesting. These crazy guys think they can run bookie operations. It's crazy! Go ahead."

[Harry continues his presentation.] When I was in Anaheim I saw Dr. Gallagher do his thing on generating random numbers and I couldn't come up with anything that made sense to me, you know in my limited environment as to what I would use random numbers for and then a light went on and I came out with a completely useless application. [Dick Azzi interrupts again,] "This runs at the police department doesn't it?" [Harry continues.] You know the axioms of police don't you. If it doesn't work, buy it. If it makes sense, ignore it. If it's incompetent, promote it. [Laughter and applause.] I haven't tested this running live to find out if it really works or not, but Dr. Gallagher promises me that the random number generator will work with this formula and assuming it does, this is neat. We have a practical application for it if, please God, we go to Las Vegas instead of Anaheim sometime. The random number generator produces random numbers between 1 and 6, which if you happen to have the same mentality I do, that's the same number of sides on a dice. So what this does is play a game of craps. [Laughter] [Editor's note. Information on the function fn$random appears at the beginning of this article.]

```
DELETE DICE_GAME;
REDEFINE PROCEDURE DICE_GAME
!
! This procedure pays a game of craps using the random number
! generator, fn$random, presented by Dr. Joe Gallagher
!
! declare the variables needed for the dice rolls
!
DECLARE D1 USAGE IS BYTE.
DECLARE D2 USAGE IS BYTE.
DECLARE D3 USAGE IS BYTE.
DECLARE D4 USAGE IS BYTE.
DECLARE ROLL_1 COMPUTED BY (D1 + D2).
DECLARE ROLL_2 COMPUTED BY (D3 + D4).
DECLARE POINT USAGE IS BYTE.
! allow the player to enter the starting amount; limited to $10,000
DECLARE START_STASH USAGE IS WORD
VALID IF START_STASH LE 10000.
```

```
START_STASH = *."Your starting Cash Stash"
! the amount accumulated during play must allow for an amount greater
! than the starting amount
DECLARE STASH USAGE IS LONG.
STASH = START_STASH
! do not allow the payer to bet more than he/she has
DECLARE BET USAGE IS WORD
VALID IF BET LE STASH.
DECLARE GO PIC X.
! set up a condition to allow the player to continue to the game
! until he/she runs out of money
GO = "Y"
WHILE FN$UPCASE(GO) = "Y"
BEGIN
      IF STASH < 1 THEN ABORT "Sorry, you′re broke."
      PRINT "Come out Roll", SKIP
      BET = *."your bet"
!
! 6.0*fn$random + 0.5 is uniformly distributed on the range
!           0.5 .ge. random number .lt. 6.5
! the fn$nint  (nearest integer function) collapses to the integers 1 to 6
!
      D1 = FN$NINT(6.0 * FN$RANDOM + 0.5)
      D2 = FN$NINT(6.0 * FN$RANDOM + 0.5)
      PRINT SKIP, "Die 1 rolled"|||D1, "Die 2 rolled"|||D2, SKIP,
      "Role is"|||ROLL_1, SKIP
      IF ROLL_1 EQ 2, 3, 12 THEN
      BEGIN
! This is called "CRAPS: and an automatic loser on the come-out roll
            STASH = STASH - BET
            PRINT "You Lose ... Your Bankroll ="|||STASH
      END
      IF ROLL_1 = 7, 11 THEN
      BEGIN
! This is called a "Natural" and an automatic winner on the come-out roll
            STASH = STASH + BET
            PRINT "You Win .. Your Bankroll ="|||STASH
      END
            IF ROLL_1 = 4,5,6,8,9,10 THEN
      BEGIN
! This establishes a "POINT" and the dice are rolled until
! the POINT is repeated for a win, or a 7 is rolled for a loss.
            POINT = ROLL_1
            PRINT SKIP 2, COL 3, "Your point is ..."|||POINT, SKIP
            D3 = 0        ! make sure that the previous point role doesn′t
            D4 = 0  ! affect this point
            WHILE (ROLL_2 NE POINT, 7)
            BEGIN
                  D3 = FN$NINT(6.0 * FN$RANDOM + 0.5)
                  D4 = FN$NINT(6.0 * FN$RANDOM + 0.5)
                  PRINT SKIP, "Die 1 rolled"|||D3, "Die 2 rolled"|||D4, SKIP
                  IF (ROLL_2 = POINT) THEN
                  BEGIN
                        STASH = STASH + BET
                        PRINT "You Win ... Your Bankroll ="|||STASH
                  END
                  IF (ROLL_2 EQ 7) THEN
                  BEGIN
                        STASH = STASH - BET
                        PRINT "You Lose ... Your Bankroll ="|||STASH
                  END
            END
      END
      GO = *."Another Game?"
END
PRINT SKIP 10, COL 30, "Thanks for the Game", SKIP 10
END-PROCEDURE
```

It bets all your variables for you. It allows you to decide how much money you want to start with, up to $10,000, you can't go crazy. It then allows you to figure out how much you want to bet, it will not let you bet more than you have. [Question from audience.] Will that do pass, no pass? [Harry continues.] I haven't gotten that far yet, yes, this is pass and no pass, that's all it is. You want to choose your points, you want to do odds and all that, that's running on my Apple at home. [Laughter] Very well, I owe the machine 1114 thousand dollars [From the audience, "and I bet he pays it too"]. [More laughter] Six more payments I get my daughter back. [More laughter] You're not allow you to bet more than you have ... generates 2 random numbers, adds them up, tells you the come out rule, adds them up, if your result is 7 or 11 you automatically win. It adds what your bet was to how much you had and allows you to start over again if you want. If the number comes out to be 2, 3 or 12 it says "Craps, sorry you lose", subtracts your bet from the thing, allows you to start over again. If its 4, 5, 6, 8, 9 or 10, it says this is your point and continues to generate new random numbers until you either get a 7, which means you lose and subtracts your bet. ... If you have a 4, 5, 6, 8, 9 or 10 it continues to roll, it says wild, roll, not equal point or 7, do your routine again. As soon as you hit the point (repeat the point) then prints you win, adds the money to it, ask if you want to start over again. If you hit a 7 first, it says you lose and subtracts the money and if you hit a point where you're out of money (by the way the STASH is usage long so you can have up to $2 billion in winnings) but you can only bet something like $32 grand at a shot. ... If you run out of money when you get to the end it says do you want to play again, you say Yes, it says I'm sorry you're broke and throws you out. So, there's a very impractical use for random numbers. Thank you.

[Next, Bart Z. Lederman present some magic on COMPASS and VECTOR plots. His material has already been published as a fully article in last month's newsletter. In the middle of Bart's presentation, the Great Wombat made a short appearence.]

Bert A. Roseberry, U. S. Coast Guard, Washington, DC

This one just came in.

Top 10 Things To Talk To Developers About

10. Ask Any Schneider about his kids.
9. What's your sign?
8. What's your node name on the E-net?
7. What's your badge number and cost center? [Laughter]
6. Do you know Boo? [Laughter from the in group who knows who Boo is]
5. Can I have a business card with the phone number not blacked out? [Laughter]
4. Is Wendy Herman in a "family way"? [oooh]
3. When is the Ultrix version hitting the streets?
2. How's the food in the MK cafeteria?
1. Compiler design theory. [Polite, but restrained applause.]

Alan H. Beer, Compusult Software Services, San Jose, CA

I came in late so if either of these magic [presentations] are something that somebody already did [already], don't throw tomatoes at me. My name is Al Beer. One thing I had to deal with working with a fairly popular manufacturer management system, whose name I won't mention, just to avoid commercialism, is they use a date structure that's a little different from what most people using VMS use and some people in the audience who use this product and use DATATRIEVE might recognize. [Fumbling with his transparencies.] I'm sorry I don't do windows. OK. This particular popular manufacturer application tends to store their dates using a rather unusual format. Its stored in a long ward as an integer and they take the year, multiple by 10,000 add the month multiplying by 100 and add the day and something like today would be 19,880,519 [May=05, 19th day of 1988] and then that gets stored in the long ward and that fits nicely in 4 bytes. They don't worry about time. However, when it comes to trying anything by way of arithmetic with these dates, when you're using DATATRIEVE to report it, it would really be nice to have the VMS date format. So, for those people out there who've had to deal with such products, here's a method that will let you convert on the fly. For starters, before you get into DATATRIEVE, you have to define the logical

```
$ DEFINE DTR$DATE_INPUT YMD
$DTR
! DECLARE A_DATE USAGE IS LONG
!  DEFAULT IS 19880519.
DECLARE VMSDATE COMPUTED BY
FN$DATE(FN$UPCASE(FORMAT(FORMAT
A_DATE USING 9999/99/99) USING DD-MMM-YYYY)) .
```

... just those three letters. When you define that logical name as YMD that means DATATRIEVE is gonna expect any text it gets, that's suppose to be treated as a date, to be in the order year, then month, then day. Once you've established that and gotten into DATATRIEVE, at that point I have this comment to variable declaration, actually in this, A_DATE would be the field you're reporting from. You could declare a computed variable with this ugly mess here, that takes A_DATE, first formats it as a string of dates in order of year, month, day, which is getting to look familiar, then assuming that this is being treated as year/month/day it reformats it in the normal VMS layout. Well, once that's formatted, unfortunately one of the things DATATRIEVE does in this formatting is it gives us mixed case months and there's a function in DATATRIEVE called fn$date which will take a date string and convert it to an actual date field quad ward type of thing, except it can't handle mixed case on the month, it gives you a real ugly system error, so what you've gotta do is convert that to upper case which is what the fn$upcase function does. So this whole thing taken together takes your integer, breaks it into what looks like a date, reformats it to what looks like a date, converts the date to uppercase, then passes it to the date function and you've got a date veritable. You can then perform any kind of DATATRIEVE date arithmetic you want, like adding an integer to advance its days, play any games with the date functions, etc. That might be of use to some folks around here.

Now another item that I came up with, and again if someone else did this please don't hit me, is a comment searcher for FORTRAN. Every now and then we have to deal with FORTRAN code and in order to find sections of a program that deal with a particular concept, hopefully, the program's commented. Now we could get into an editor and just start searching for the thing by hand or we could use the DCL search command, but why don't we use DATATRIEVE, if we've got it there, right.

```
DECLARE COUNTER USAGE IS LONG.
DEFINE RECORD FORCOM-RECORD USING
01 FORCOM-REC.
03 COM_CHAR PIC X.
03 COMMENT PIC X(131).
;
DEFINE DOMAIN FORCOM USING FORCOM-RECORD ON PROG;
```

So the first thing I do is declare a count variable, then I have a record definition that will take a look at my FORTRAN source code and treat the first character as its own field, which I call COM_CHAR, then it treats all the rest of it as one string field. I set up a domain and I define it so that the file for my domain is gonna just be the logical PROG.

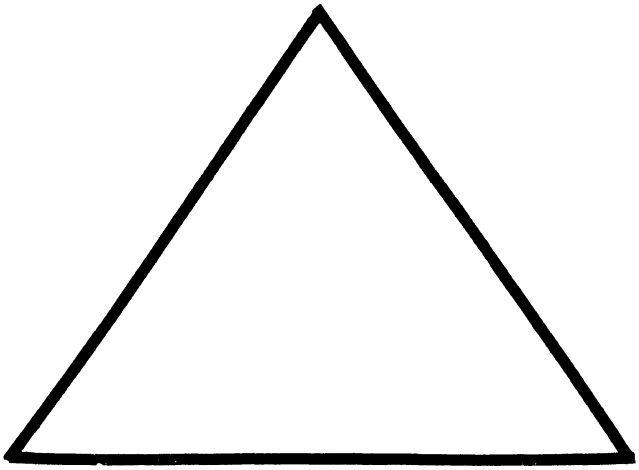
```
FN$CREATE_LOG("PROG",*."file to search")
READY FORCOM
DECLARE STR PIC X(20).
COUNTER = 0
STR = *."string to match"
FOR FORCOM
BEGIN
    COUNTER = COUNTER + 1
    IF (COM_CHAR EQ "C","*" AND COMMENT CONT STR||"") THEN
    BEGIN
        PRINT COUNTER, COMMENT
    END
END
```

I then use the fn$create_log function to establish a definition for PROG on the fly. This is nice so that every time I run my procedure I can keep pointing to a different file without having to redefine the procedure. So, I prompt for file to search and that gets plunked into user mode logical PROG which then is going to become the file if this domain ever gets ready, which it does. I declare a string variable; I think 20 characters should be adequate for what I'm going to search for. I ready my domain, which is now going to look at the file that I gave it for the logical name PROG. And first I set count to zero, then prompt for the string value, using the string to match as prompt. So at this point the person's first been asked the file to search, then he asks to enter the string to match then it gets into this loop ... increments the count and if the COM_CHAR is a C or a star which is the typical comment character in FORTRAN and comment contains what our search string is, print [the] count [and] comment so we have a line number and the comment that we were looking at. And that's it. It goes through the file, prints out every comment line that has the string we were looking for.

DECUS

edusig

EDU



EDUSIG COMMUNICATION

James R. Gerland
SUNY/Buffalo

DECUS Symposia sessions tend to promote considerable discussion on methods, policies, and general purpose information relating to the 'running of the shop' back home. Such conversations brings out bothpositive and negative results of user experiences. Invariably though, they prove beneficial as we all have to deal with similar difficulties no matter what the academic environment. They also help us in EDUSIG to identify subject material worth consideration for future presentation and on-going discussion. Probably the biggest problem though is that many of us are unable to attend the DECUS Symposia and even those who can attend don't have enough time during the week to exploit this information sharing.

OK people, we now have an alternative available which should extend this kind of discussion beyond the symposia sessions themselves. I have established an electronic discussion list to promote EDUSIG related dialogue on a year-round basis. One of the prime finctions many of us would like to see this list used for is to facilitate the discussion of future DECUS symposia session topics. Anyone with ideas for topics or speakers (or even wishes to volunteer something!) is encouraged to participate.

The remainder of this article outlines this service.

EDUSIG-L @ UBVM

Back in the dark ages people who worked in offices would have stacks of paper on their desks with tiny slips of paper stapled to them known as distribution lists. This worked fine as long as you were one of the top three people on the list, but if you were lower in the distribution stream, by the time you finally received the notice it was old news.

Welcome to the 1980's. Now there is a way for all of us to share our EDUSIG discussions in a very timely manner. At SUNY/Buffalo there exists an IBM VM/CMS disconnected service machine (a detached process in VAX/VMS terms) called LISTSERV@UBVM. This software was written by Eric Thomas of the Central School of Paris France to allow people to exchange information using BITNET.

This discussion list (EDUSIG-L@UBVM) will enable us to exchange EDUSIG related information. Some of you were members of the original list and some of you have added yourself already but what about the rest of you. You can send commands to LISTSERV@UBVM to add, modify, or remove yourname from our list. To access this list via BITNET use your favorite mail utility to send a letter to LISTSERV@UBVM with the 'command' you want LISTSERV to perform as the first line in the body of the mail, not the subject: line. The following is a description of valid commands, their syntax, and function, which LISTSERV understands and will process (the UPPER CASE letters indicate the allowed abreviations):

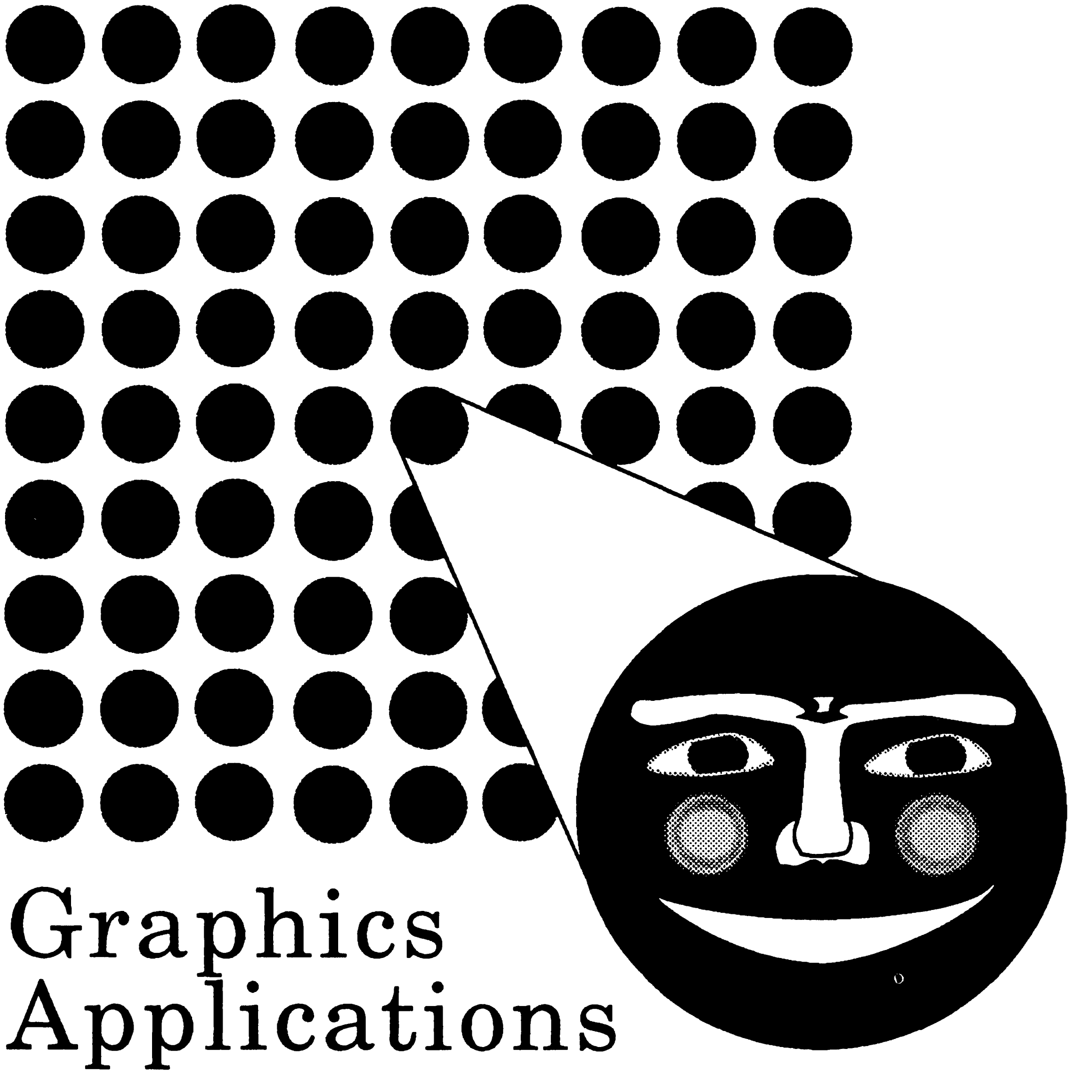
```
Help                          Obtain a list of commands


Info <topic ¦ ?>  <F=fformat> Order a detailed information
                              memo, or ? for a list of
topics:
                              GENintro REFcard NEWs KEYwords
                              Features LPunch LINKing OWNers
                              COORDination JOB  DISTribute
                              AFD FILEs FILEOwner

List <options>  <F=fformat> Obtain a description of all lists
          Short
          Long = Detailed

Query    listname             Review  your  distribution
                              options for a given list

REView   listname <(options>  Review the contents of a list
                   LOCal
                   Countries
                   Short
                   NOHeader
                   Msg

SET    listname <options>   Alter your distribution options
               Files/NOFiles    for a given list
               Mail/NOMail
              ACK/NOACK/MSGAck

SIGNOFF  listname             Remove yourself from a list

SUBscribe listname <full_name> Subscribe  to  an open list
                               Change your name if already
                               subscribed

Parameters description
----------------------
listname  = 1 to 8 characters from the following set: A-Z 0-
9 $#@+-_: full_name = first_name middle_initial surname
(*not* "userid at node")
```

The full list of commands is available from LISTSERV by sending the INFO REFCARD' command to LISTSERV@UBVM via mail.

Once you are a mamber of EDUSIG-L you can participate in the on-going discussions by sending mail to the userid EDUSIG-L@UBVM. The LISTSERV machine then takes your letter and redistributes it to every name on the EDUSIG-L list.

If you have any questions about using this facility send mail to me
<GERLAND@UBVMS.BITNET>.

GRA



Graphics Applications Special Interest Group Newsletter

table of contents

submissions

Articles, copies of viewgraphs, tips and tricks, and graphics output are all welcome submissions for the Graphics Applications Special Interest Group (GAPSIG) newsletter; here's how YOU can make submissions:

1) Send in a tape. Tapes can be 1600 or 6250 BPI density. Please provide straight ASCII files or Mass-11(TM) format documents and a letter with your name and address. Please place any charts in separate files.

2) Send in paper. Hey, your editor can type and chew gum at the same time, so don't be afraid to send in hard-copy. And, if all you have is notes, FINE! Send them in!!! We have many folks who can take the ideas and flesh them out with English language extensions. Questions, with or without answers, are desirable, too.

3) Mail the article, etc. to user name HAYS on DCS.

Your editor's address appears in **from the editor**, so mail your submission today!

from the editor

This editorial is solely the opinion of the author and does not necessarily reflect the views of DECUS, Digital Equipment Corporation, or KMS Fusion, Inc.

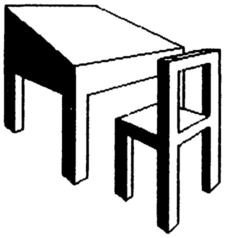
Bob Hays
KMS Fusion, Inc.
3621 South State Rd.
Ann Arbor, MI 48106

Here is part one of the UIS to X migration material that the rumor mill said might exist inside Digital. Many people have asked about this document at the symposiums and at our local user's group (MIVAXLUG), so here it is, in three monthly installments. Since the first installment is a bit short, Paul Waterstraat's report on the last ANSI standards meeting is included for your perusal.

You'll also note the addition of the **from the chair's desk** column in this issue. Bijoy Misra is the Chairperson of the Graphics Applications SIG, and he plans to provide a regular column with information about SIG business and activities.

Next month is the Fall Symposium. I hope everyone attending has a super time and learns a great deal. Also, if you have an interest in Graphics Applications (you're reading this, right?), stop by the SIG Suite or the SIG Campground (see the weekend Update.Daily for locations) and offer to help out. User's groups require member support and volunteers to succeed. It only takes a little time and effort; the results are often astonishing.

Also please note the competition for best color hardcopy at the Fall Symposium. This should be a real treat, and, if we can swing it, the winners may get their images published right here (if you can get approval for publication and if we can get approval for color pages in the newsletters).



from the chair's desk

Bijoy Misra
GAPSIG Chairperson

GAPSIG Organization

DECUS is a perfect example of the creative spirit of volunteer energy! The Graphics Applications SIG team is lucky to have a set of excellent players, but we need more. Today is an exciting time to work in graphics; the whole world is turning to it and needs expertise and experience to support the activities. Also we need to know what level and kind of support people need. Our Special Interest Group has come a long way to become a strong unifying force for graphics in Digital. But we need your input to determine what we can do to be more effective. Please send us any comments, suggestions or opinions for our SIG. Don't feel shy. Write up your thoughts in the form of a letter and send it to Bob and let it reach out to other people. This is your newsletter, and we're all volunteers. There is always scope for improvement!

Talking about the newsletter, we would also like to find volunteer(s) to act as Associate Editors. We need people to solicit articles and to report on various graphics applications like DECwindows, Vaxstation UIS, PHIGS, GKS, image processing, electronic publishing, etc. Let us know if you've worked in any of these areas and would like to share your expertise. After hearing from you, we'll form new working groups in these areas. We also need an expert to coordinate the SIG activities in engineering and laboratory applications. Thanks again to Digital, Eric Rehm was stolen from us. All is not lost again, of course, since he will be available to help us inside Digital when we need such help.

The Graphics Steering Committee Woods Meeting was held last spring. It's interesting and exciting to see how much experience and expertise is available in this group. There is publishing and production experience in Henry Schneiker and Mike York; expertise in high level software and standards in Bob Hays and Paul Waterstraat; expertise in algorithms and hardware in Jim Sims and Mike McPherson; expertise in image processing and animation in Bob Goldstein and myself. Dottie Elliott is involved in human interface designs, and Carol Schwob is expected to help us in the areas of workstations and DECwindows. Robert Krieg, who is our new Library Co-ordinator has expertise in system management, and Stephen Schultz, the new Store Representative, brings with him strong expertise in imaging technology.

The detailed list of various people with the titles of their job functions appears elsewhere in this newsletter. You may write to them or call them directly on any ideas that you would like us to pursue. If you have any questions, again, please send them to Bob Hays or contact another Steering Committee member, and we shall furnish you with an answer or a response through the newsletter. If you need help in any of your specific applications, you may call any of us directly and we'll try to help you as best as we can. The advantage of a SIG organization is that if an individual doesn't have an answer, somebody else may have an answer. Hence, it doesn't hurt to ask!

I wish to thank Bob Hays for the excellent leadership he has demonstrated in putting together this newsletter. About a year ago, the newsletter project for our SIG was in a sorry state. Then came Jim Flatten, our energetic DEC counterpart. I may remind you that he's the same Jim who was the Standards Co-ordinator for the SIG and exposed us all to the vagaries of graphics standards and encrypted abbreviations. Well, the SIG's loss was DEC's gain. But we did not lose out completely. He has been one of our DEC counterparts and continues to have an official position in the SIG. Jim was the one who brought the newsletter from its deep sleep and revived it. Now Bob is putting together lots of energy and many new ideas. My hearty congratulations to both Jim and Bob!

ANSI graphics (X3H3) meeting

Paul Waterstraat
GAPSIG Standards Coordinator

A general meeting of the American National Standards Institute (ANSI) committee on Computer Graphics (X3H3) was held in Fairfax, Virginia during the week of May 8-13, 1988. The ANSI X3H3 committee is the official body licensed to develop American National Standards in the area of computer graphics. The ANSI X3H3 committee also serves as the U.S. representative to the International Standardization Organization (ISO) in matters relating to the development of international graphics standards.

X3H3 is divided into eight task groups that focus on the following areas:

X3H3.1	Programmer's Hierarchical Interactive Graphics System (PHIGS)
X3H3.2	Graphics Architecture
X3H3.3	Virtual Device Interface
X3H3.4	Language Bindings
X3H3.5	Graphical Kernel System (GKS)
X3H3.6	Window Management
X3H3.7	Validation, Testing, and Registration
X3H3.8	Imaging.

The first three days of the meeting in Fairfax were spent in individual working group meetings resolving issues, writing documents, reviewing previous work and planning new work. The last two days were spent in plenary session with presentations by the working groups on progress and plans. What follows is a brief summary of the reports of those working groups.

<u>X3H3.1</u>
(Programer's Hierarchical Interactive Graphics System)

The Programer's Hierarchical Interactive Graphics System (PHIGS) standard is intended to provide a set of functions for modelling and presentation of 2D and 3D graphical objects. The modelling concepts include multi-level hierarchical structuring of graphical data and rapid, dynamic generation of a picture as a result of an interaction in a near "real time" environment.

PHIGS is currently on the verge of becoming an ANSI standard. Final approval is expected during the June meeting of the International Standardization Organization (ISO) in Tucson, Arizona. It is hoped that the Tucson meeting will also produce text for a matching ISO standard.

The PHIGS group has also been authorized to continue work on the extensions to PHIGS known as PHIGS PLUS. These enhancements are expected to be in the areas of curves and surfaces and in lighting and shading.

X3H3.2
(Graphics Architecture)

The Graphics Architecture task group works in coordination with the other X3H3 task groups to develop overall reference models for computer graphics. The purpose of a reference model is to explain the relationship between existing graphics standards and to define relationships between standards in computer graphics and those in other areas. In addition, the task group coordinates the identification of user requirements, and develops and promotes formal specification techniques for computer graphics. The goal here is to provide an overall framework or architecture into which current and future work on graphics standards will smoothly fit.

X3H3.3
(Virtual Device Interface)

The Virtual Device Interface task group develops standards for the interface between the device-independent portions of a graphics system (i.e., those portions that construct or describe a picture) and the device-dependent portions (i.e., those portions that render or display a picture on a graphics device). Project development is subdivided into two groups.

The Computer Graphics Interface (CGI) task group deals with the "internal" interfaces that permit dissimilar graphics devices to be controlled by a single general graphics system.

The Computer Graphics Metafile (CGM) task group deals with the "external" interface (for example, picture files and graphics metafiles) that permit graphics information to be exchanged among dissimilar and sometimes physically separate graphics installations.

At this meeting the Computer Graphics Interface (CGI) task group has presented a new Interim Draft of the proposed CGI standard. Current work includes preparing U.S. positions to ISO on remaining open issues and checking the Interim Draft for proper incorporation of resolved issues. The CGI group hopes to have the second Draft Proposal of the CGI standard available by mid-November, 1988. Work on the CGI Binary Encoding will need to wait for the final semantics decided upon for the Draft Proposal.

The Computer Graphics Metafile (CGM) group is currently working on three areas, named CGM Addendum 1, 2, and 3.

CGM Addendum 1 concerns concerns extensions to the CGM to accommodate GKS Metafiles for all output levels of GKS, by building upon CGI functions. Among the features added in Addendum 1 are: segmentation support, device viewport control, closed figure support, pixel array and drawing mode support, ability to override font list and character list on a picture by picture basis, and capabilities needed for dynamic picture regeneration.

Comments on the first Draft Proposal of CGM Addendum 1 were processed at a meeting last March at Blakeney, in Norfolk, England. Most of those comments were expressions of concern that the CGM addendum was diverging from the proposed CGI standard or that the CGM addendum was dependent upon the CGI proposal, which is unstable. The principle reply was that any divergence from the proposed CGI was unintentional and probably the result of confusion about the current state of the CGI. (It has been over a year and a half since the previous draft of the CGI was released.) The availability of the new CGI Interim Draft (see above) should help resolve those differences. The second Draft Proposal of CGM Addendum 1 is likely to begin circulation in late 1988.

CGM Addendum 2 concerns extensions to the CGM to accommodate GKS-3D metafile support. The scope of CGM Addendum 2 was greatly expanded by Germany and the U.K. at the meeting last March in Blakeney, England to encompass not only GKS-3D metafiles, but also PHIGS metafiles and PHIGS archive files. One serious conflict that arises from this expanded scope concerns coordinate systems. The GKS-3D metafile is expressed in normalized device coordinates, the PHIGS metafile is expressed in normalized projection coordinates and the PHIGS archive file is expressed in modelling coordinates. As the CGM was initially designed as a picture capture mechanism, it is not clear how well the new scope of CGM Addendum 2 fits with the original architecture. Work is currently focused on clarifying scope and goals for Addendum 2.

CGM Addendum 3 concerns extensions to the CGM to support engineering drawing, technical illustration, raster images, color models beyond RGB, and advanced text and font support. A functional specification and a proposal seeking support of this work is currently being drafted.

X3H3.4
(Language Bindings)

The Language Bindings group develops the semantics of language- specific procedure calls for graphic standards. The sequence of events through which a standard proceeds is: proposal for work, functional specification, working draft, draft proposal, public review, perhaps a second draft proposal and public review, and finally, publication of the standard. Shown below is a table that indicates the status of the various Language Binding projects.

Project	Status	completion date
GKS Language Bindings:		
FORTRAN	Standard	1988
Pascal	Standard	1988
ADA	Final Review	1988
C	Public Review	1988-89
LISP	Proposal for work	?
MUMPS	Proposal for work	?
GKS-3D Language Bindings:		
FORTRAN	Draft Proposal	?
Pascal		?
ADA		?
C		?
PHIGS Language Bindings:		
FORTRAN	Draft Standard	1989
Pascal	Working Draft	?
ADA	Functional Spec.	?
C	Working Draft	?

X3H3.5
(Graphical Kernel System (GKS))

The Graphical Kernel System (GKS) currently is an American National Standard (ANSI X3.124-1986). Current work of this task group is to provide maintenance, interpretation for, and plan enhancements to GKS. In this area, several projects are under way: a GKS Information Bulletin, development of GKS-3D, and GKS Revision.

An initial draft of the GKS Information Bulletin has been circulated for public review and comments are being processed.

At present there are no plans to make GKS-3D an ANSI standard. The only activity involves participating in the progression of GKS-3D as an ISO standard. All significant work on GKS-3D has been undertaken in the International arena. There is currently not enough interest in this country to continue work on GKS- 3D. ISO is expected to publish a GKS-3D standard during mid to late 1988.

The GKS standard has been around for almost five years now, and there is growing interest in revising or updating the standard. Discussion of the GKS Revision Cycle at the ISO meeting in Berlin last November showed a diversity of opinion as to its purpose, ranging from fixing of errors to a non-compatible total replacement. The only decision reached so far was to draft a new work proposal.

X3H3.6
(Window Management)

The Window Management task group is currently working toward developing a standard specifying the functional behavior of an X Window System 11 server and the data stream encoding for communication between client programs and X Window System version 11 servers. This boils down to transforming the X Window System from Project Athena at MIT into an ANSI standard. The group has technical support from the X Consortium. Bob Scheifler, one of the principal architects of the X Window System spent several days at this meeting working with the window management group to answer questions and help clarify issues.

The proposed name for the document to be produced by this group is "X Window System Data Stream Encoding". The document will comprise three parts: a functional specification, a data stream encoding, and a KEYSYM encoding which will provide mapping from the symbol on the keycap to a specific binary code.The group adopted an aggressive schedule which, if met, would mean standardization of X by the end of 1988.

Future work will include consideration of establishing a language binding to X at the X-lib level.

X3H3.7
(Validation, Testing, and Registration)

This group works to develop validation and testing strategies for computer graphics standards. In detail they work to coordinate activities concerning the development of test suites for computer graphics.

The CEC Project for GKS Certification has established a Control Board which will review tests and complaints about them. Licenses to operate official testing services will be distributed soon to the National Bureau of Standards (NBS), and others. At the NBS, information packets are currently being distributed. (The test software will be available for a few hundred dollars. Having a test performed will cost thousands of dollars.)

The GKS/FORTRAN tests have been converted to C, and are now awaiting the final approval of the ANSI C standard before final revision and distribution.

A draft for the standard on Conformance Testing is currently being written.

X3H3.8
(Imaging)

This is a new group within X3H3, and is in the process of reviewing current practice in imaging in an attempt to identify the kinds of functionality that are candidates for standardization. This group will also be developing a model to elucidate the relationship between imaging and graphics.

X3H3
(General Business)

Documents to be included in the next general mailing to X3H3 membership should be in John McConnell's hands by June 17. Letter ballots to be included in that mailing should be dated July 15. The deadline for the following mailing is August 19, and those letter ballots should be dates September 9.

The following is a tentative schedule for future X3H3 meetings:

October 17-21, 1988 Houston Texas

February 6-10, 1989 Milpetas, California

June 26-30, 1989Kona, Hawaii

The Graphics Arts Contest

The Graphics Applications SIG will host a "Graphics Arts Contest" during the Fall 1988 DECUS Symposium at Anaheim, Ca. All members of DECUS are encouraged to participate in this contest. The only requirement for entry into the contest is that the graphic was produced on or processed with Digital equipment. The following information must be provided for each entry:

> the hardware and software used to produce the picture,
>
> the name and address of the creator(s), and
>
> the following statement, which must be signed: This graphic may be displayed at the DECUS Symposium, Fall 1988, and may be published in the DECUS SIG Newsletters.

An optional 25 to 100 word description of the algorithms used and/or the purpose for the graphic may be included. The graphic will be returned to the submitter(s) only if a note requesting the return is provided with a return address. Entries will be accepted at the campground on Monday, October 17 morning between 9 am and 12 noon, or for those not attending the symposium, mail entries to:

Bijoy Misra
Center for Astrophysics
60 Garden Street
Cambridge, MA 02138

Posted entries must reach Bijoy before Friday, October 14. Two prizes each of $100/ and $50/ will be awarded in the categories of:

(a) color, and
(b) black & white.

All pictures will be exhibited in the Graphics SIG campground. The winners will be announced in the Friday's Update.Daily at the Symposium, and the winning entries will be published in the DECUS SIG Newsletter. Good luck!

a guide to migrating UIS applications to X

January 20, 1988
Digital Equipment Corporation
Maynard, Massachusetts

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 software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license. No responsibility is assumed for the use of reliability of software on equipment that is not supplied by Digital Equipment Corporation of its affiliated companies. The following are trademarks of Digital Equipment Corporation:

DEC
VAX
VMS
ULTRIX
VAXstation

UNIX is a trademark of AT&T.

X Window System is a trademark of Massachusetts Institute of Technology

Cray is a trademark of Cray Research, Inc.

TABLE OF CONTENTS

Preface

As the cost of computers has decreased and the power has increased, styles of computing have changed. Batch computing gave way to interactive computing which yielded great increases in productivity. Networking then brought about the concept of distributed computing. The advent of workstations such as the VAXstation family has generated another quantum leap in the productivity of a wide variety of users. Applications benefiting significantly from workstations include software development, engineering, image processing, medical, publishing and many others.

One of the key features of workstations that has helped to bring about this productivity gain is the window system. Window systems create an environment similar to a desktop. In this environment, there may be many documents on the desktop with some visible and others partially or fully covered. In the same manner, window systems allow you to have many windows (corresponding to the documents) on a physical screen (corresponding to the desktop). Text and graphics may be combined in these windows.

Each window may contain a separate task, all of which can run simultaneously. In addition, a single task may control more than one window. The user may input to any of the windows at any time. For instance, an IC layout editor could allow a designer to create one window which displays the layout of the entire IC and another window with an enlarged view of the area to be edited. Editing could be reflected in both windows.

Most window systems have required applications to run on the workstations CPU in order to take advantage of the windowing capabilities. The MicroVMS Workstation Software (VWS or UIS) is an example of this type of window system. Recently, window systems such as the X Window System (X) have eliminated this restriction by allowing applications to perform windowing functions across a network. This type of window system affords many advantages. In particular, an application may utilize any computing resources available on the network. while displaying all graphics on the local workstation.

Digital has standardized on the X Window System in order to provide this networking capability and a common window environment across all of its workstation platforms regardless of the operating system. The X Window System is the cornerstone of the DECwindows architecture.

The X Window System was developed by Project Athena and the Laboratory for Computer Science at the Massachusetts Institute of Technology with funding and participation by Digital Equipment Corporation and International Business Machines Inc. Because X has been placed in the public domain and was designed to be device independent, ports have been made to a variety of platforms. For example, some users today are executing their X applications on Cray supercomputers while displaying the output directly on VAX workstations.

While X has been supported on Digital's ULTRIX based workstations for the past year, Digital and several other companies recently issued a joint press release announcing the adoption of the X Window System as their corporate window offering. This announcement committed support for X under a variety of operating systems.

As a result of Digital's standardization on the X Window System, support has been committed for both the VMS and ULTRIX operating systems. Thus, many users will wish to port their VMS applications from UIS to X.

Standardizing on the industry leading window system, the X Window System, will provide applications with enhanced portability across operating system and hardware boundaries.

1 INTRODUCTION AND OVERVIEW

This document provides information to aid software developers and architects with the port of UIS applications to X. The intended audience of this document is the application developer who has written a VWS based application or is in the process of designing a VWS based application which will be ported to the X Window System. This document assumes that the reader has some knowledge of graphics and windowing systems. As always, certain terms have conflicting meanings when used in different software architectures. When this situation arises, this document gives the definition based on the X and VWS usage.

1.1 Structure Of This Document

Preface

Part I - This section discusses the intended audience, purpose and structure of this document.

Part II - This section is a brief overview of the VWS and X window systems. This section gives an architectural overview of the windowing systems, comparing and contrasting the capabilities of each.

Part III - This section is an application design guide for the application producer who is in the process of building a UIS application with intentions to migrate to X when it becomes available under VMS. This section will provide hints to make applications more portable between UIS and X.

1.2 Bibliography

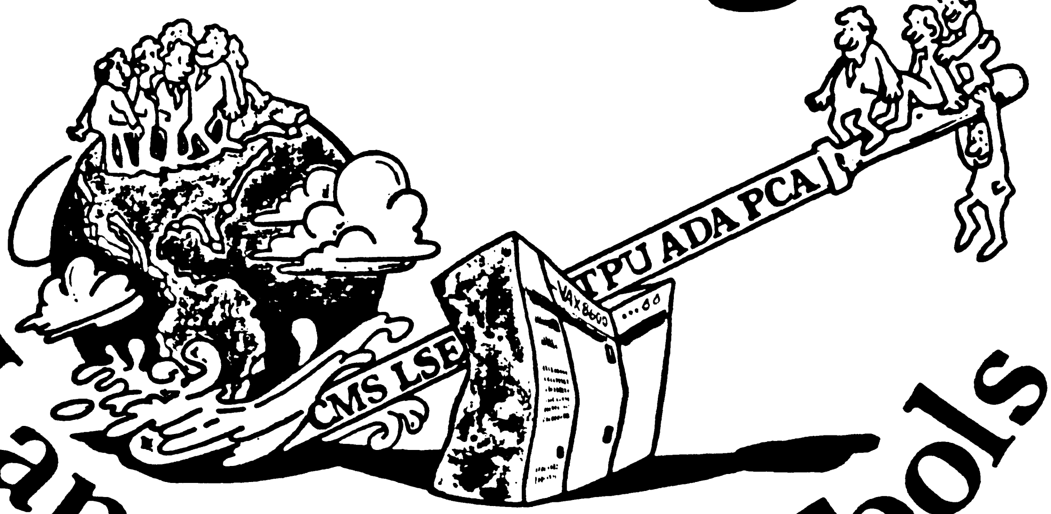
Digital Equipment Corporation
"MicroVMS Workstation Graphics Programming Guide"
May 1986, Maynard, MA

Digital Equipment Corporation
"Xlib - C language X Interface"
Jan 1986, Maynard, MA

Massachusetts Institute of Technology
"Xlib - C language X interface"
"Protocol version 11"
Jan 15, 1987

Paul Kimball, Worksystems Consultant
"An Overview of VAX Workstation Software Under VMS and ULTRIX"
Regional Mini-DECUS,
26 August 1986, San Jose, CA

Leverage



Languages & Tools

L & T

TABLE OF CONTENTS

EDITOR'S NOTES

Once again, I'm rushing to get this issue to the DECUS office on time. For once, however, it's not entirely my fault. Most of the material was formatted when I left for a SIG Steering committee meeting last weekend, with plenty of time to do the remainder on my return. My car, however, had other ideas. The timing belt gave up the ghost on I495 just outside Marlboro, and I enjoyed two extra days of hospitality from the Marlboro Holiday Inn. Of course it only took three hours or so to fix the problem, but the local Ford dealership, which seems to feel that the word "service" only means (eventual) repair work, would not even look under the hood until forty-one hours after the car was towed in. At any rate, I have not had the opportunity to proofread completely, and accept responsibility for any egregious formatting or typographical errors.

After two empty months, this is our second substantial issue in a row. The first article is a synopsis of the various L & T SIG meetings at the Cincinnati symposium. Our SIG Secretary, Mark Kidwell, has compiled discussions from the meetings throughout the course of the week.

The second article is By Dr. William W. Hastings, of Fordham University. In it, he discusses efforts he has made towards generating a compiler for DCL, and problems encountered. This is quite an interesting contribution, and could spark some useful discussion.

Next comes an announcement of the public review and comment period for the draft Pascal standard. *If you are involved in Pascal, you should read this immediately!* The review period is short and the lead time in producing this issue gives you very little time to procure and study the draft before the comment period ends.

Bob Van Keuren has submitted a report of the APL working group meeting from Cincinnati. One of the "deliverables" from our working groups is supposed to be a report of activities, and I'm very glad to see some of these starting to be submitted to *Leverage*. Bob's report is excellent, and if you are involved with APL you should be sure to read what the Working Group is pursuing.

Finally, Ken Schilling has submitted two articles regarding the DIBOL standards activities. First is a report on the standards committee meeting, followed by a background article on the history of the ANS standards activities for DIBOL. I confess to knowing nothing about DIBOL, but found this very interesting reading from a historical standpoint.

As a last reminder, I still need submissions for the newsletter. In paricular, our working groups should be using this medium to report on their activities and solicit participation. My address is included at the end of this issue in the Steering Committee listings.

Al Folsom
Leverage Editor

L&T IN CINCINNATI

by Mark Kidwell
L&T SIG Secretary

The following is a synopsis of the L&T Steering Committee meetings during the 1988 Spring Symposium in Cincinnati.

Saturday, May 14:

This symposium organization meeting dealt primarily with the logistics of L&T's Spring Symposium. Below are notes of current or long term interest:

Session Notes Editor Mark Katz reported that L&T's Cincinnati Session Notes contain 982 pages! Mark has done an outstanding job of putting these together in four volumes: Monday, Tuesday, Wednesday + Friday, and Thursday. There is some concern that DECUS attendees may have difficulty with these session notes. For example, having Wednesday and Friday session notes in one binder could be confusing. Another difficulty in using the notes could be the volume. Approximately 650 pages had been estimated for the Anaheim session notes, and more than 900 pages were finally submitted. It was suggested that a description of L&T Session notes organization might help new DECUS attendees (or perhaps even repeat DECUS attendees) make better use of them. Sam will give this in his Roadmap Session.

Mark Katz is considering having session presenters forward notes in a printable format. The resulting notes might be "less pretty" (and formatted in many different ways), but his work load would be reduced.

An attempt will be made to obtain questions/presentations for the Wizardry Session and possibly the Q&A session prior to those sessions in order to make the best use of time. The intent is to "filter" the questions and NOT to restrict the types of questions/presentations!! If this task can be accomplished, its effectiveness should be evaluated later.

Dave Ream is working on the L&T Volunteers database. He requested that ALL the Working Group Chairs send him membership lists with addresses AS SOON AS POSSIBLE. He would also like to receive SIG position job descriptions. Joe Pollizzi agreed to submit a general Working Group Chair description. Any Working Group Chairs who feel their particular interest groups require specific job descriptions should submit them to Joe.

Several Steering Committee positions are vacant and need to be filled. Some are per-symposium positions basis (eg. Campground hosts), others are permanent positions. The following are positions which need to be filled:

A. Folder Editor
    [Ed. Note: Subsequently filled by Don Amby]

B. Store Representative
    Recruits and keeps a list of SIG Store volunteers, submits names to Dave Ream for entry in the volunteer database, and ensures the L&T SIG provides workers for the store.
    [Ed. Note: Howard Holcombe elected to continue in this post]

C. Bird Watchers
    Attend BOFS's related to L&T.
    Secure lists of the attendees & their addresses.
    Write brief reports.
    Determine whether working groups might be formed.

D. BOF Coordinator:
    Recruits & organizes bird watchers.
    Collects reports on the BOFs.
    Keeps track of scheduled BOFs and assigns chairs for them.

E. Campground Hosts
    Collect surveys & distributes gifts.
    Distribute buttons, pins.
    Identify campground features (message board, literature).
    Act as reference points for directing visitors to experts.

F. Tape transcribers:
    Transcribe and edit session tapes for the newsletter.
    Work closely with Symposium Coordinator, Newsletter Editor, Working Group Chairs.
    Do not necessarily have to attend symposia. Most of the group agreed that tapes could be sent to people not in attendance.

The Masters List must be kept current. The Masters Coordinator should ensure that people on the list are still willing to be called. More importantly, addresses and phone numbers must be up-to-date. The Masters Coordinator will send postcards to all Masters to see whether they wish to continue as Masters in their topics. Master's Coordinator Dena Shelton and Assistant master's Coordinator George Scott will trade positions, with George becoming Master's Coordinator.

The group discussed the 1989 DECUS/DECWORLD conflict. There is really nothing that L&T can do about the problem. Celeste LaRock stated that the situation should not hurt the SIG. DECWORLD is primarily a "Marketing" event, whereas DECUS is more technically oriented. DECUS did consider changing the DECUS schedule (to Las Vegas in fact), but too many logistics/commitments are already set.

Symposium Rep Earl Cory and 4 or 5 others will be meeting in June to study the problem of reducing the number and increasing the quality of L&T Sessions. Hopefully, Earl can be relieved of some of his burden.

Sam asked for ideas that could be used for store items at the Anaheim symposia. The store has some really tight deadlines. They are:

May 27	Commit to the type and style of item to be placed in the Store.
June 10	Firm description and quantity of the items.
July 11	Final artwork and color schemes
July 15	Place the order

It was suggested that the Anaheim "caps" should be top hats. The possibility of changing the L&T logo will be also investigated.

Any Steering Committee member who does not have a picture in the L&T Folder should send one to Sam ASAP. There was also discussion regarding other items to be placed in the folder. Anyone who has any new ideas for representing the Working Group Roadmaps should contact Joe Pollizzi.

The SIG tape was supposed to be done by Friday of the Symposia, but one of the contributors had made enhancements to software previously submitted and wanted the enhanced software to be included. The tape will be completed by the end of May.
[Ed. Note: The SIG tape was finished and submitted during the Symposium]

Saturday's Meeting ended with a 1.25 hour non-disclosure session by Celeste LaRock.

Thursday May 19 Open Steering committee meeting:

Digital employees with set-up badges experienced problems gaining access to the campground on Saturday at this symposium for the purpose of installing equipment. We had understood that this would be allowed, although we were aware that SIG members would not have access until Sunday. Rick Eliopoulos, the Symposia Committee's Campground Coordinator, explained to Sam Whidden and Earl Cory that the problem arose this time because the campground area doesn't belong to DEC (as the Exhibit area does); it is shared by the L&T/AI/UNISIG SIGs, so it is a DECUS (Symposium Committee) responsibility. The L&T Steering Committee will meet on Saturday from 1 p.m. to 5 p.m. in Anaheim. Access to the campground will be needed on Sunday at 9 a.m. Sam and Earl plan to work on this problem. We have stressed the need to address our requirements early for the Anaheim symposium.

In Anaheim the symposium will use the Convention Center and the Marriott Hotel. This will cause things to be "distancy". Lunches will be served at the East parking lot of the Marriott. The Tri-SIG campground will be 18x27. Campground Coordinator Joe Pollizzi has the layout for Celeste, Sam, and those with immediate need for it.
[Ed. Note: After the Symposium, word was received that the Anaheim Campgrounds will be larger than the space originally allocated, which was great news indeed!]

Barry Breen, L&T's Seminars Rep, raised a question about the possibility of having 2 days (Saturday and Sunday) for Presymposium Seminars. The general response was that there has to be a limit to the number of PSS's and Sessions. An additional day would pose too much of a problem for people involved with the managing the SIG's Symposium. Some present felt that PSS's may have an impact on the quality of symposium sessions, in that people who used to give sessions at the symposia and are now giving PSS's have

stopped giving sessions - or at least have stopped giving sessions with topics related to those of the presymposium seminars.
[Ed. Note: It might be helpful to see whether these beliefs are backed by statistics.]

If at all possible, Disneyland will be the Anaheim reception attraction despite rumors to the contrary and what may have been stated at the previous woods meeting.

There was some discussion regarding the reasons why the symposium schedule is Monday long, Tuesday reception, Wednesday off, and Thursday long. One question raised was why not have Monday night off, Tuesday long, Wednesday reception, Thursday long. The idea was to allow travellers to recover from any time difference, and to give the people who attended the presymposia seminars a break. Counter-arguments raised were several:

1. Tradition - DECUS members typically do not like changes in the symposia schedules. Monday night has been getting longer to accommodate the extra sessions.
2. By keeping the long days separated, DECUS attendees have a "greater chance" of recuperating between long days.
3. Many companies use Wednesday night as the "company night" - especially if there is a branch of the company nearby that can play host to the other company employees.

The 45 minute sessions were evaluated. They are generally not liked and, the group agreed that whenever possible 1 hour sessions should be scheduled. Dissatisfaction was raised on several fronts:

1. Not enough time for questions
2. Sessions conflict with the rest of the symposia streams (overlapping)
3. DEC felt that they didn't really have the content and format that should have been in the presentations - content and format that was taken out due to time constraints. (Ken Hobday stated: We can do 5 minute sessions also - the session will not really convey the technical information properly or clearly)

Some members felt that the SIG starts off fine. Most things run "smoothly", as Steering Committee members often handle problems on the fly during the symposium. Communication for the SIG tends to become spotty, however, as the symposium progresses, until the Open Steering Committee meeting on Thursday. The possibility of having a Steering Committee Breakfast at least once during the week to go over any problems that surface or adjustments that might need to be made was mentioned.

It was suggested that the SAGs contain some identifier indicating when session notes for a given session are available. This would not be possible for the SAGs in the Preliminary Program, since the deadline for the SAG is weeks before the deadline for the session notes. An alternative suggestion was to add an identifier to the SAGs included in Update.Daily? Earl Cory claimed that the logistics would be overwhelming right now. The issue was tabled for the time being, and it will be brought up at a future date with the individuals who would be involved.
[Ed. Note: L&T will try to include a Session Notes table of contents in its Anaheim Folder, to see if that helps]

L&T Steering Committee members have received complaints concerning the problem of the reprinting of the session notes by the DECUS Store. The L&T session notes ran out sometime Wednesday. Apparently, some orders were taken, and then the notes were reprinted. At the same time, however, some attendees were told there would be NO reprinting of the session notes. This problem will be investigated by the Store Rep and hopefully addressed before the Anaheim symposia.

The Working Group Chairs say the L&T Working Group Business Cards are definitely successful. A suggestion was made to cut down the number of cards to 50 per Working Group, unless the price difference is not considerable.

The issue of SIG overlap was brought up again. Sam wants to encourage the SIG Council to urge SIGs to form more joint Working Groups. The benefit would be that the roadmaps could include streams across SIGs. Abstracts from other SIGs would be required in order to investigate the overlap problem. At present, Sam provides copies of the raw L&T Abstracts to all Working Group Chairs, to enable them to create their roadmaps; for them to include sessions from other SIGs might require working with the abstracts for all 1000 Symposium sessions. Joe Polizzi suggested that perhaps using just the indexes for the full Symposium might be sufficient.

Analysis of Session Evaluation cards needs to be completed prior to the Session Quality meeting on June 10.
[Ed. Note: George Scott, Master of Session Evaluation Card Statistics, succeeded in preparing a session evaluation report before the June 11 meeting of the special team to cull Anaheim sessions, an outstanding accomplishment!]

The possibility of a new L&T logo was discussed again. Members agreed that the current logo was fine, but that a new button is needed. It should be expressive of the symposia or at least have a different representation each year. Lisa Kerby-Rodgers knows someone who has designed logos for several companies. She will examine both the idea for the button and the logo.

The idea of an alternate gift choice for survey completion was discussed in order to encourage the people who already have the L&T corkscrew to fill out more survey forms.

On the issue of the Store items, the idea of having a key ring was trashed because many groups (especially at DEXPO) GIVE key rings away. It was agreed that key rings should be give away items. A thermal mug, or a telescoping pointer with either a light at the end (like the PSS pointers) or one that doubles as a ballpoint pen were suggested as alternatives.

At the end of the meeting, it was announced that if the Working Group chairs wanted to make announcements during a session, they should give copies of the announcements to the Session Chair coordinator PRIOR to the distribution of session evaluation forms to session chairs.

End of the Steering committee meeting, May 19, 1988.

TOWARD A DCL COMPILER

Written by: Dr. William W. Hastings
Mathematics Department
Fordham University

Mailing address: 86-B Forsythia Ct.
Paramus, NJ 07652

Telephone: 201-368-2466

NOTE: I plan to submit this program to the DECUS tape by the end of the summer.

Introduction

Anyone who has written DCL command procedures knows how difficult they are to debug. Since command procedures must be interpreted, their execution can be frustratingly slow.

A DCL compiler would help solve both problems. The compiler would flag errors and generate cross-reference tables for use in debugging. By creating an executable image, execution time would improve, much as a BASIC compiler speeds up BASIC programs.

DCL Compiler is Impossible

Unfortunately, a DCL compiler cannot be written. The reason is simply that the meaning or value of any command name, logical or symbol can be altered dynamically, during the execution of the command procedure. Here is an extreme example:

```
$ INQUIRE  OLD_SYMBOL  "Enter symbol to redefine"
$ INQUIRE  NEW_VALUE   "Enter new value"
$ 'OLD_SYMBOL' :== NEW_VALUE
$ DELETE  'FILE'.DAT;0
```

After seeing the third command, a would-be compiler has a serious problem. For example, in analyzing the fourth command, the compiler must consider the possibility that the symbol FILE has been redefined. Or perhaps the meaning of DELETE has been changed to "TYPE"! For that matter the third command might not be an assignment statement! (For example, the user could response to the first INQUIRE command with "STOP !" , in which case the third command returns the user to command level 0; i.e., the user gets the DCL prompt.)

Well, you say, who would write such a sequence? (No one, I hope!) But the above example only exposes the tip of the iceberg. Any global symbol referenced (but not defined) in the command procedure presents a problem because the symbol can be redefined between compilation and execution. (This observation applies as well to any command name, since the command tables may be altered or the command may be redefined by making the command name a global symbol.)

Consider

```
$ IF 'EXPRESSION' THEN GOTO DONE
$ PRINT 'FILE''EXTRA'
```

Is the value of EXPRESSION the name of a symbol? Is, for example, EXPRESSION := I.GE.N .OR. DONE ? Is EXTRA a file extension or a PRINT qualifier? Along these lines the most troublesome possibility is that the value of a symbol includes a critical punctuation mark (like the exclamation mark following STOP in the first example).

Technical Problems

In addition to the fundamental limitations suggested above, DCL presents the would-be compiler writer with plenty of annoying technical problems. First of all are the anomalies concerning strings. For example the following two assignments are acceptable to DCL:

```
$ A = "abc"def
$ B = "abc -
```

(The result is that A = "abcDEF" and B = "abc -" .) On the other hand the hyphen at the end of the following line does indicate continuation!

```
$ A = " USER  ''F$EDIT( USER_NAME, -
$ B = " TIME  ''F$FAO( "!%T", 0 )'"
```

In the second line, the exclamation point does not start a comment even though it follows the second quotation mark.

Logicals present some of the same problems as symbols. In particular the compiler can not know what logicals are defined or what are their translations.

When is a keyword not a keyword? The following, in which the character string "THEN" appears as a label, logical, symbol, filename and keyword, could be legitimate DCL:

```
$ THEN: IF  THEN  THEN  TYPE  THEN:THEN
```

Finally, suppose a command procedure contains the following two lines:

```
$ AB = ABC + ABCD + ABCDE
$ AB*CDE = 5
```

If the second command is executed before the first, then the four symbols in the first command are all the same symbol. Thus four entries in the symbol table must be collapsed to one.

What is Possible

Even with all of this bad news, many steps performed by a compiler are still possible. The remainder of this article describes a program called DCL_PARSER which parses DCL command procedures. The program recognizes the new features added to DCL in version 5.0 of VMS. For those faced with the task of incorporating these new features into existing procedures, this program should be a handy tool.

The most important step performed by DCL_PARSER is the construction of tables of symbols and labels. By making reasonable assumptions (for example, "IF" has not been redefined), each appearance of a symbol can be correctly identified. Hence it is possible to build cross reference tables of global symbols and local symbols appearing in each SUBROUTINE or the main procedure. With these tables the DCL programmer can see where each symbol is referenced or changed. In particular one has a list of symbols which are referenced but not defined within the command procedure.

An example of the output of DCL_PARSER appears in Figure 1. The procedure is listed with line numbers, which are used in the following symbol tables. This procedure appeared in the February, 1988, issue of The VAX Professional.

The one difficult case for identifying symbols concerns foreign commands. In

```
$ MYSTERY  PARAMETER
```

PARAMETER would usually not be a symbol, although it would be if, for example, MYSTERY = "WRITE SYS$OUTPUT".

The program also can detect the following syntax errors:

Invalid parameter or qualifier
Illegal expression
Unterminated or misplaced SUBROUTINE definition
Multiply-defined or undefined label
Invalid label references
Label appearing with a THEN or ELSE command
Invalid SET/SHOW option
Missing THEN in IF or ON command
Malformed block IF construction
Abbreviated symbols with common prefixes
Other invalid constructions

In addition, DCL_PARSER reports the occurrence of data lines, that is, lines without the dollar sign prefix. (Remember that DCL now requires the dollar sign on every line it reads.)

Again, it should be noted that DCL_PARSER will sometimes be confused. For example

```
$ MYSTERY  A+B+C  D
```

is syntactically correct if MYSTERY = "COPY" , but contains an invalid expression if

MYSTERY = "WRITE SYS$OUTPUT".

Finally, DCL_PARSER generates a syntax tree. That is, each element of the command procedure is a leaf node of a tree. The internal (non-leaf) nodes of the tree show higher level constructions like "file specification" or "qualifier". The syntax tree produced for line 9 of the procedure in Figure 1 is shown in Figure 2.

This syntax tree will be the basis for further development of DCL_PARSER. In particular, it will be possible to check for semantic errors such as inappropriate parameters or qualifiers for a given command. In addition, a flow-of-control analysis is possible. This involves constructing a directed graph whose nodes are indivisible sequences of commands (containing no branches) and whose edges show branches (GOTO, GOSUB, etc) between indivisible sequences. This analysis would allow the program to determine, for example, that a symbol is defined as a constant before it is referenced. In this case the symbol could be replaced by the constant before semantic checks are made.

The Parser Program

This program was written with the aid of two utilities from the UNIX world called LEX and YACC. (In fact, this entire project started as an exercise to learn YACC.) LEX generates a lexical scanner from "regular expressions" which describe the "tokens" of the language. The lexical scanner is a module which accepts a character stream as input and outputs the tokens found. Figure 3 is a partial list of the tokens recognized by the scanner. This list includes only tokens of (possibly) more than one character.

The program YACC is a parser generator. That is, it generates parsing tables from a set of "translation rules".These rules take the form, for example,

```
file_specs :  remote_node DOUBLE_COLON local_file_specs
```

This rule says that a "file_specs" may be a "remote_node" followed by the token DOUBLE_COLON followed by a "local_file_specs". Other rules specify what constitutes a "remote_node" or "local_file_specs".

In Figure 4 is a partial list of simplified transition rules. These rules contain many ambiguities; in particular there is no indication of the punctuation required to separate parameters or to how distinguish command names from parameters. The resolution of these deficiencies requires intervention by the scanner-parser interface (see below). Also missing in Figure 4 are rules of precedence and all semantic actions. There are actually 351 translation rules.

The problem with using YACC is that DCL is not a language which can be described by a set of translation rules of the type accepted by YACC. (Technically, there is no LALR(1) grammar for DCL.) There are several reasons why this is so, of which two cause considerable trouble. First of all, in languages like Pascal or C, whitespace is not a token returned by the lexical scanner. (Whitespace is required only to separate tokens; it is not a punctuation character.) In DCL, however, whitespace is a crucial punctuation in some contexts while in other contexts it is ignored. The problem is that in expressions, for example, the parser requires that whitespace be ignored.

The second problem concerns strings like "THEN" or ".AND." which, as shown above for "THEN", can play several different roles. (The string ".AND." is either a logical operator or part of a directory name as in [DATA.AND.MORE].) In a language like Pascal, the lexical scanner returns special tokens for keywords. Obviously, this approach will cause serious problems for a DCL parser.

These problems, and more, are handled by a special interface between the lexical scanner and the parser. As an example, when the parser is expecting an expression, this interface reduces the three tokens in the string ".AND." to a single token.

At this point I should mention that I could have used the system service TPARSE (except for the fact that I did all of the programming on a PC). The problem with using TPARSE is that symbol substitution and lexical function calls normally take place before invoking TPARSE, which cannot be the case for this project. Consequently, some of the most useful features of TPARSE such as "match a file specification" cannot be used. On the other hand, TPARSE allows backtracking, a real advantage.

Figure 5 shows the overall construction of the program. The main program processes the command line arguments and calls the parser routine. After parsing is completed, the main program calls functions in the symbol table module to print the symbol tables.

The pre-scanner echoes the input and handles continuation lines. The lexical scanner breaks the input into tokens. The only unusual tokens are STRING_SEGMENT and DOUBLE_QUOTES. (See Figure 3.) There are three reasons for this construction. First of all, the characters appearing in a string must be screened for symbol references. Secondly, in most languages, a string begins with a quotation mark and ends with the next one. (Of course, an exception is made for two consecutive quotation marks, which are treated as one quotation mark and part of the string.) In DCL, this is not the case because a string may include a lexical function reference which in turn contains a string as an argument. Thirdly, as mentioned above "abc"def is a valid (or at least acceptable) string.

The heart of the program is the parser. The parser uses the tables generated by Yacc to process tokens supplied by the interface between the scanner and the parser. This interface filters the tokens sent to the parser. For example, it checks for DCL command names and lexical function names. In the other direction, the parser informs the interface what type of token is expected (for example, a component of a file name).

Yacc allows "semantic actions" to be connected with each translation rule. In practice this means that when each unit of a command is recognized, a function in the semantic action module is called to add the appropriate nodes to the syntax tree. In addition, when a symbol or label is processed, a function in the symbol table module is called to enter the symbol or label reference into the appropriate table.

The program is available to anyone interested in seeing whether it performs a useful function. I would like to hear your comments concerning how the program could be improved. Please send any suggestions to me at 86B Forsythia Ct., Paramus, NJ 07652.

FIGURE 1

```
$! File:     DELETE.COM           Author:   Kevin Hartig
$! Function: Add qualifier /VER=(n,m) to DELETE command.
$!           Qualifier deletes file versions from n to m.
$!
$!Concatenate input string and extract filename.
$ INPUT_STRING := ’P1”P2”P3”P4”P5”P6”P7”P8’
$ FILENAME = f$extract(0, f$locate("/",INPUT_STRING), INPUT_STRING)
$ENTER_FILE:
$  if FILENAME .nes. "" then goto PARSE_QUALIFIERS !have FILENAME
$  inquire FILENAME "_File"  !Prompt for filename if one not found.
$  goto ENTER_FILE
$!
$!Separate qualifiers from filename and extract /VER=(n,m)
$!Extract n,m from /VER=(n,m). If /VER= not found, delete normally.
$PARSE_QUALIFIERS:
$ QUALIFIERS=f$extract(f$locate("/",INPUT_STRING),255,INPUT_STRING)
$  QUALIFIERS = f$edit(QUALIFIERS, "collapse,upcase")
$  OFFSET = f$locate("/VER=(",QUALIFIERS) + 6
$  if OFFSET .eq. f$length(QUALIFIERS) + 6 then goto NORM_DELETE
$  LEN = f$locate(")",QUALIFIERS)- OFFSET
$  VERS = f$extract(OFFSET,LEN,QUALIFIERS)    ! VERS equals => n,m
$!
$!Define low and high file version numbers as integers.
$ LOW_VER = f$extract(0, f$locate(",",VERS), VERS)
$ LOW_VER = f$integer(LOW_VER)
$ OFFSET = f$locate("," ,VERS) + 1
$ LEN = f$length(VERS) - f$locate("," ,VERS)
$ HIGH_VER = f$extract(OFFSET, LEN, VERS)
$ HIGH_VER = f$integer(HIGH_VER)
$!
$!Remove /VER=(n,m) qualifier from qualifier string.
$ QUALIFIERS = QUALIFIERS - "/VER=(" - "”LOW_VER’,”HIGH_VER’)"
$!
$!If /VER=(n,m) qualifier was specified, loop through to delete
$!desired files with specified qualifiers.
$ VER = LOW_VER
$DEL_VERSIONS:
$  if VER .gt. HIGH_VER then exit  !Exit when files deleted
$  delete’QUALIFIERS’ ’FILENAME’;’VER’
$  VER = VER + 1
$  goto DEL_VERSIONS
$!
$!If /VER=(n,m) qualifier was not specified, delete normally.
$NORM_DELETE:
```

```
45 $  delete'QUALIFIERS' 'FILENAME'
46 $  exit
```

Global symbols and symbols local to the main procedure

```
F$EDIT()  global symbol
   17  r-value
F$EXTRACT()  global symbol
    7  r-value           16  r-value          21  r-value
   24  r-value           28  r-value
F$INTEGER()  global symbol
   25  r-value           29  r-value
F$LENGTH()  global symbol
   19  r-value           27  r-value
F$LOCATE()  global symbol
    7  r-value           16  r-value          18  r-value
   20  r-value           24  r-value          26  r-value
   27  r-value
FILENAME  local symbol
    7  local definition   9  r-value          10  local definition
   39  r-value           45  r-value
HIGH_VER  local symbol
   28  local definition  29  r-value          29  local definition
   32  r-value           38  r-value
INPUT_STRING  local symbol
    6  local definition   7  r-value           7  r-value
   16  r-value           16  r-value
LEN  local symbol
   20  local definition  21  r-value          27  local definition
   28  r-value
LOW_VER  local symbol
   24  local definition  25  r-value          25  local definition
   32  r-value           36  r-value
OFFSET  local symbol
   18  local definition  19  r-value          20  r-value
   21  r-value           26  local definition 28  r-value
P1  local symbol
 . . .
P8  local symbol
   6  r-value
QUALIFIERS  local symbol
   16  local definition  17  r-value          17  local definition
   18  r-value           19  r-value          20  r-value
   21  r-value           32  r-value          32  local definition
   39  r-value           45  r-value
VER  local symbol
   36  local definition  38  r-value          39  r-value
   40  r-value           40  local definition
VERS  local symbol
```

```
    21  local definition  24  r-value         24  r-value
    26  r-value          27  r-value         27  r-value
    28  r-value


Labels of the main procedure

DEL_VERSIONS  label
        37  local definition      41  r-value
ENTER_FILE  label
         8  local definition      11  r-value
NORM_DELETE  label
        19  r-value          44  local definition
PARSE_QUALIFIERS  label
     9  r-value          15  local definition
```

Figure 1: Sample output of DCL_PARSER

FIGURE 2

```
9 $  if FILENAME .nes. " " then goto PARSE_QUALIFIERS  !FILENAME entered
REGULAR_DCL
        NULL node
        IF_STATEMENT
                EXPR
                        IDENTIFIER  FILENAME
                        RELOP  NES
                        QUOTED_STRING
                                DOUBLE_QUOTES  " "
                                NULL node
                                NULL node
                TRANSFER_STATEMENT
                        TRANSFER_COMMAND  GOTO
                        IDENTIFIER  PARSE_QUALIFIERS
```

Figure 2: Sample syntax tree output by DCL_PARSER

FIGURE 3

Tokens

q_char = [A-Z0-9$_%] | "\"\""
Characters considered part of a quoted string when they follow the closing quotation mark. The vertical bar means "or". Thus a q_char is a letter, digit, percent sign, dollar sign, underscore, or 2 consecutive quotation marks.

DOUBLE_QUOTES : "\"\"" q_char*
"" followed by any number of characters of type q_char

STRING_SEGMENT : "\"" q_char*
A quotation mark followed by any number of characters of type q_char

INTEGER : [0-9]+
One or more digits

DECIMAL_NBR : "%D" [0-9]+
HEX_NBR : "%X" [0-9A-F]+
OCTAL_NBR : "%O" [0-7]+

IDENTIFIER : [A-Z$_] [A-Z0-9$_]*
One or more letters, digits, dollar signs or underscores. The first character may not be a digit.

LABEL: [A-Z0-9$_]+
COLON_DOUBLE_EQUAL : ":=="
COLON_EQUAL : ":="
DOUBLE_EQUAL : "=="
DOUBLE_COLON : "::"
TRIPLE_DOT : "..."

UNKNOWN : no_good+
Any string of characters none of which may appear in a DCL command (except inside a quoted string).

Figure 3: Tokens of more than one character

FIGURE 4

```
command:
             if_command               | on_command
|            set_command              | show_command
|            write_command            | read_command
|            transfer_command         | inquire_command
|            wait_command             | call_command
|            other_dcl_command        | foreign_command
|            invoke_procedure         | assignment_statement

if_command: IF expression THEN command

on_command: ON on_keyword THEN command

on_keyword: ON_KEYWORD | quoted_symbol

set_command: SET set_parameter parameter_part
             /* E.G. SET PROTECTION=W:R FILE.DAT */

set_parameter:
             set_symbol qualifier_part
             /* E.g. TERMINAL/DEVICE=VT200 */
|            set_symbol colon_or_equal qualifier_value qualifier_part
             /* For example PROTECTION=G:RE/DEFAULT */

set_symbol: SET_KEYWORD | quoted_symbol

write_command: WRITE qualifier_part logical expression_list

transfer_command: TRANSFER_COMMAND label_reference

inquire_command: INQUIRE qualifier_part symbol message_part

call_command:
             CALL qualifier_part label_reference parameter_part
             /* E.g. CALL/OUT=CHECK.LOG CHECK_VALUES 'P1 'P2 */

other_dcl_command: DCL_COMMAND qualifier_part parameter_part

foreign_command:
             symbol limited_qualifier_part foreign_arguments

foreign_arguments: parameter_part | expression_list

invoke_procedure: '@' filespecs limited_qualifier_part parameter_part
             /* E.g. @INFO/OUTPUT:USERS.LIS USERS */
```

```
assignment_statement:
            assignment_opening  assignment_operator  expression
|           assignment_opening  string_assign_op  unquoted_string

assignment_opening: symbol | star_symbol  | symbol  selector
            /* E.g.  BLANKS[0,80]  */

star_symbol: symbol  '*'  symbol

selector:  '['  expression  ','  expression  ']'

parameter_part:
            parameter_list          | parameter_list  '@'  filespecs
|     '@'  filespecs          | empty

parameter_list: single_parameter  | parameter_list  single_parameter

single_parameter:  unqualified_parameter limited_qualifier_part

unqualified_parameter:
            symbol                      | filespecs
|           unsigned_constant           | multi_filespecs
|           date_time_value

qualifier_part:
            regular_qualifier_part
|           regular_qualifier_part  '@'  filespecs

regular_qualifier_part:
            qualifier_list
|           qualifier_list  prime_first
|           prime_first
            /*  E.g. in  $ SET MESSAGE 'SAVE_MESSAGE'  */
|           empty

qualifier_list:
            '/'  qualifier
|           qualifier_list  '/'  qualifier

qualifier:
            symbol
|           symbol_list  colon_or_equal  qualifier_value
            /*  E.g.  /SELECT=SIZE=(MIN=10,MAX=100)  */

symbol_list:  symbol  | symbol_list  colon_or_equal  symbol

qualifier_value:
            symbol  | filespecs  | constant  | date_time_value
|           '('  value_list  ')'
```

value_list: single_value | value_list ',' single_value

single_value: constant | qualifier
/* E.g. in /KEY=(SIZE:4,POS:1) */

Figure 4: Translation Rules

FIGURE 5

```
                                                                ________
                                                               |semantic|
                                                          ---->|actions |
 ________      ________       _________                   |    |________|
| pre-   |    |lexical|     | scanner-|        _________   |        |
|scanner |--->|scanner|---->| parser   |---->| parser  |   |        |
|________|    |_______|     |interface|      |_________|  __|__
                            |_________|          |        |symbol|
                                                 |----->|table |
                                                          |______|
```

Figure 5: Overview of DCL_PARSER

X3 PUBLIC REVIEW AND COMMENT PERIOD

X3 Announces the Second Public Review and Comment Period on
Draft Proposed American National Standard X3.160-198x,
Programming Language Extended Pascal

Washington D.C. -- Friday, 15 July 1988.

X3, the Accredited Standards Committee on Information Processing Systems, announces the second public review and two-month comment period on the draft proposed American National Standard, X3.160-198x, Programming Language Extended Pascal. The public review period extends from 12 August 1988 to 11 October 1988.

Extended Pascal is an upward-compatible extension to the presently- existing standards for Pascal. It has been developed in an international effort between the American Joint X3/IEEE committee, the British Pascal committee, International Working Group on Pascal, and the ISO/JTC1 Subcommittee 22. As such, Extended Pascal is being put forth as an American, British, and International standard. This standard is intended to coexist with the existing Pascal standards until such time as these standards are no longer needed.

Extended Pascal is a modern version of Pascal, the classic language for teaching programming concepts and programming languages. In this tradition, Extended Pascal contains new features that reflect the modern software engineering techniques of modularity, protectability, and abstraction. In addition, Extended Pascal is a vibrant language containing many of the features that have been added (often in different ways) to existing Pascal implementations.

The major features in Extended Pascal that distinguish it from the classic language defined by Jensen and Wirth are: modules with separable definition and interface parts (including initialization and finalization of modules), a generalized type definition mechanism (schemata) which permits the static and dynamic selection from a "set of types", a secure method for initializing variables of a given type (even if they are outside of the module where the type is defined), and the ability to define private variables which have restrictions placed on their access.

In addition, Extended Pascal contains many pragmatic features which make it a more practical and usable language than "classic" Pascal. Some of these are: string types of both dynamic and fixed length, short-circuit logic operators, exponentiation, direct-access files, external file binding, complex numbers, constant expressions, record and array value constructors, non-decimal integer constants, time and date functions, and numerous other features.

Extended Pascal also includes many simple conveniences which have become standard practice, such as an underscore character in identifiers, relaxed order of declarations, and an otherwise clause in the case statement and variant records.

This draft standard is available for public review and comment for a two-month period ending October 11, 1988. Copies may be obtained from GLOBAL ENGINEERING DOCUMENTS, INC. by calling 800-854-7179. The International number is +1-714-261-

1455 and the FAX number is 714-261-7892.

Note: Because the public comment period only lasts two months, it is advisable that all requests be made by telephone so that you will have more time to submit comments within the designated period for public comment.

Single Copy Price: $ 35.00 International Copy Price: $45.50

For more information contact:

Thomas N. Turba
Chair, X3J9 Pascal

Phone: 612-635-2349
CSNET: Turba@SP.Unisys.com
TELEX: 297072
FAX : 612-635-2003

APL WORKING GROUP MEETING

by Bob Van Keuren

The APL Working Group Meeting in Cincinnati was held on Thursday, May 19. There were eight people present.

We discussed possible future sessions for APL. Some ideas that came up were calling external routines from APL, and graphics.

A plea was made for newspaper articles.

The group discussed the APL character set and the terminals available for APL: the VT100, 200, 240, HDS. One person noted that on the HDS the characters were larger and clearer. It was mentioned that some people object to the "funny characters", but what's getting popular now? Graphics terminals with icons. And what is an icon? A "funny character"!

One person mentioned a problem with MASS-11.

The Wish List was discussed at some length. Here is a brief list of wishes mentioned at the meeting.

- Set default mode for error messages to display without numbers.

- De-VAX the syntax. Example:)DROP name instead of)DROP name.apl;*

- Library numbers (IBM syntax?) This one led to some hot discussion pro and con.
- Write to library with password.
- Add session editor to APL.
- Max core: turn off, or use system default.
-)PUSH privileges
- Dump stack if APL blows up.
- The editor interface is weak, compared to other implementations. Beef up the TPU link. For example, on Erase Character.
- Session Manager with full screen editor. Keep a record of where you are.
- There is a question of which way to go: TPU, SMG, windows, ...
- IF..THEN..ELSE syntax. (What, keywords in APL??!!)

Various kinds of APL applications were discussed, including farming, bonds, music, and trading.

The group talked about APL standards. There is a meeting at SDSC headquarters on Monday, May 23.

Jim H. Logue offered to be our representative to ACS.

Samuel Harbold from Digital asked for a copy of the wish list, when it is assembled. He indicated that DEC has a commitment to APL, and that the budget is increasing.

The group suggested beefing up the session "What is APL?" We discussed more activities we could have for Anaheim. One suggestion that was adopted was to have a button made up. There hasn't been one since the "Special Character" button two years ago or so.

The group continued assembling a mailing list.

Report on the April 29-30 X3J12 ANS DIBOL Committee Meeting

This was the first meeting since the resubmission of the proposed standard after the public review period. For those unfamiliar with how the standards approval process works, here are the high points:

1. Approval must be obtained for the formation of the committee. This was done on November 28, 1984. The official committee designation of X3J12 was assigned at this time.

2. The committee membership is open to anyone interested. Noone can be barred from the committee. Voting membership is automatically granted to those who either attended the first committee meeting or who attend two successive committee meetings. Committee members are considered to have their voting membership in jeopardy if they have missed two of the last three meetings.

3. The full committee has met eight times, including this one, since the first meeting on February 21-22, 1985. The committee is solely responsible for the technical content of the proposed standard.

4. The committee operates under the administrative directive of the X3 secretariat, a part of the Computer and Business Equipment Manufacturers Association (CBEMA). X3 sponsors a public review period for the document, during which anybody may comment on the contents of the standard. X3 ensures only that all relevant technical questions have been addressed and handled and that any internal inconsistencies within the document have been resolved.

5. After successful closure of the public review, the proposed standard is submitted to the Bureau of Standards Review (BSR). This is generally no more than a formality. Official acceptance of the standard was due on May 2.

6. Finally, the entire document is submitted to the technical editor who does the final edit for style and form. This process usually takes about sixty days. Once complete, the standard has completed its length rounds through administrative channels and is available as a public document to anyone who wished to purchase it.

New Business

Having reviewed the approval procedure, the committee began to address its future. Since the committee has completed its initial charter by producing a standard, it must obtain a new charter. Much of this meeting was therefore spent in discussing what the committee's new goals should be.

This puts the committee in uncharted waters so to speak. The expressed purpose of the committee to date was to produce a "state of the DIBOL market" standard. We are now faced with forging new territory and have the challenge of producing a "state of the computer market" standard. This represents a significant departure from the previous goal, and will doubtless result in far less unanimity amongst members.

There was serious discussion concerning whether or not a new revision of the standard should even be attempted. There was strong sentiment that the committee had accomplished its initial purpose, and that it should simply let the work speak for itself. A motion was even made to this effect, but it was defeated by a 3-8 vote.

Despite this, there is strong sentiment that to justify the existence of the committee and to best serve the interests of the DIBOL community, that simply adding language elements that make DIBOL more like all other languages is not sufficient; that something must be done to the language that will draw more users to it, that somehow distinguishes it from other languages.

To this end, the following proposal on the goals of the committee was passed:

The objectives of the proposed X3J12 technical committee are:

1 To address the existing enhancement list (X3J12/85-037) et al.

2 To address any ambiguities existing in the proposed standard X3.165-1988.

3 Removal of selected "grandfathered" portions of X3.165-1988.

4 To evaluate enhancements which will address the state of the computer market. Such areas of investigation may include:

- graphical interfaces
- data base enhancements
- coordination with other standards (SQL, X-WINDOWS, PHIGS, etc)
- program development tools

Having determined our basic goals, the committee started to tackle some of the easier issues.

Removal of Grandfathered Language Elements

The easiest issue to address was the removal of all of the "grandfathered" portions of the first standard. All such features of the language were considered loathesome by the committee for one reason or another, but in the interests of not breaking existing code with the first version of the standard, were kept in. These elements are:

1 An unsatisfied END terminates the procedure division

2 The physical end of file terminates the procedure division

3 If .MAIN is omitted, the routine name is implementation dependent

4 Implementations may restrict the routine name to six characters

5 PROC will be accepted in lieu of .PROC

6 Implementations may restrict the subroutine name to six characters

7 SUBROUTINE will be accepted in lieu of .SUBROUTINE

8 Implementations may restrict COMMON names to five characters

9 Implementations may restrict COMMON field names to five characters

10 Trailing unary operators in the data division are allowed

11 Name is optional in subroutine argument definitions

12 GO TO will be accepted in lieu of GOTO

13 OFF ERROR will be accepted in lieu of OFFERROR

14 ON ERROR will be accepted in lieu of ONERROR

15 Implementations may restrict the length of name on XCALLs

16 The second argument may be specified on DELETE

17 SI will be accepted in lieu of I:I on OPEN statements

18 SU will be accepted in lieu of I:U on OPEN statements

19 Submode on OPEN statements is not required

20 The third argument may be specified on STORE

21 The first argument may be specified on XCALL DELET

22 The first argument may be specified on XCALL RENAM

Before anyone gets up in arms over these removals from the standard on the grounds that their code will break, remember that the individual implementors may choose to keep these language elements in their next versions. It's just that code written with these elements will not be ANS DIBOL compliant.

Wish List Items

A list of approximately 50 enhancements desired by committee members was assembled and assigned to various committee members to present to the committee to be acted upon at its next meeting. The majority of these represent language elements already implemented in either VMS DIBOL or by DISC's DBL. There were two general issues around which sub-committees were formed: error handling and I/O.

Major Enhancements

The broader issues represented by item four of the stated committee goals was addressed by a far-ranging general discussion. Few members of the committee boasted expertise in any one of these areas, and committee members volunteered to make presentations regarding the current state of the computer market in these areas so that the committee can make more informed and directed decisions regarding these issues.

An Opportunity to Participate

This new evolution of the committee gives an excellent opportunity for non- committee members to get involved and have their input listened to. In particular, the wish list items and major language extensions require the input from a broader-based public than the first version of the standard.

To reiterate, the first version of the standard was more marketing-driven than technically-driven. As such, any suggestions regarding a radical change from DIBOL-83 were generally put on hold until passage of the first standard.

So now is the time for all techies, wizards and other denizens of the CPU to arise and give their input. The committee welcomes new members. However, if that is too ambitious for you, you may still have your suggestions considered by the committee. Here's how to ensure full consideration:

1. Mail to me your suggestion:

    Kenneth Schilling
    2314 Mira Vista Ave
    Montrose, CA 91020

2. "The format for submission of all enhancement proposals is:

    1. The name, address and phone number of the person making the submission.

    2. A description of the change desired.

    3. Rationale and justification for the change, including:
        - goals
        - pros and cons
        - how the proposed change fits into the DIBOL philosophy as stated in the preface to the standard.

    4. Specific format and changes the standards document.

    5. Any implications of the change to both developers and users, including, but not limited to the breaking of existing code and language performance."

This wording is directly from a motion passed in the last committee meeting. Unfortunately, the standard itself will not be available until mid-summer. As soon as I know how it can be obtained, I'll see that it gets published in the SIG newsletter.

Committee Meeting Schedule

Also by motion, in order for a proposal to be on the agenda for the next committee meeting, it must be presented in writing to committee members at least 30 days prior to the meeting. Here is the current meeting schedule:

Date	Location
Sept 22-24, 1988	Milwaukee, Wisconsin
March 30 - Apr 1, 1989	Baltimore, Maryland
Sept 14-16, 1989	Belfast, Ireland

If you can attend, great! I assure you that you input will be welcome. This is NOT a committee of stuffed shirts overimpressed by their own knowledge. It IS a committee of dedicated DIBOL users who wish to see the language thrive.

If you cannot attend, be sure to get your proposals to me in time to distribute before the 30 day deadline.

I look forward to hearing from you.

Kenneth Schilling

History of the X3J12 ANS DIBOL Committee

The history of the ANS DIBOL effort has been somewhat different from other standards efforts in that is has been driven more from a marketing perspective than a technical one.

The committee membership was unusual in that it was evenly split between language implementors, application developers and users. This, its relatively small size of 11 members and an unusual (for this type of effort) degree of agreement on goals and willingness to compromise has allowed the DIBOL standard to be developed faster than any other language standard to date.

In particular, I'd like to state up front that Digital, as represented by extremely ably by Ken Beers, has been far more cooperative than any of us imagined; and that it has been that cooperation more than any other factor that has allowed this first version of the standard to be created.

DIBOL was already far more "standard" in terms of its similarities across different operating systems and hardware platforms than most of the COBOL products available. What had hurt DIBOL and its users was its lack of recognition as a viable alternative to other languages in the programming of business applications. It was strongly felt that producing an ANS DIBOL would legitimize the language in the eyes of many users unfamiliar with it and thus create business opportunities for both language implementors and software developers. This was particularly true in the area of selling to government entities, which seldom will purchase a software product that isn't written in a standard language.

Thus, the primary effort to date has been to produce a "state of the DIBOL market" standard, rather than a "state of the art" standard. The committee used the DIBOL-83 language reference manual as its starting point. When controversial issues were broached, the prime requisite was that no existing code would be broken. If the committee felt that an existing aspect of the language had to change, then every effort was made to ensure that the change would be caught by the compiler, as opposed to producing a program that unknowingly ran differently than before. With the exception of a very few obscure elements, this policy was successfully adhered to.

This is not to say that there haven't been substantial improvements made. There were a number of conceptual holes in the language, things that "everybody knew", but that weren't thoroughly defined. These were the source of many of the differences between various implementations of the language, as well as many of the "gotchas" that would plague programmers. As one example, the committee discovered that it was impossible to accurately define how a continuation line worked until the definitions of physical lines and logical lines had been resolved.

One of the major results of the work to date, therefore, has been to provide a thorough conceptual framework for the language. The committee feels, not without a touch of hubris, that its work goes further than any other standard in nailing down the tough issues which standards such as COBOL have never fully resolved. The area of I/O in particular falls into this category. One interesting bit of fallout from this intense effort has been that there is no special glossary to the standard. All technical terms used within the standard can be found in the American National Dictionary for Information

Processing Systems (ANDIPS), which is an ANSI document.

The structure of a DIBOL program underwent considerable scrutiny, the results of which are the addition of several new compiler directives: .MAIN, .PROC, part of the standard, although future versions of the standard will most likely not support them. This addition has several very nice implications for both the language implementors and the users. It resolves the double use of END, and it tidies up the SEND and RECV statements.

The entire area of I/O has had a major face lift. The user of DIBOL will find that many previously hazy areas have been clearly stated. The exact nature of sequential, relative, indexed, print file and character I/O are clearly laid out. Special care has been taken to define "context" in each case. Context here refers to what is the current record and what is the next record. The work done here resolves some of the thornier issues of how the DISPLAY, WRITES, READS and ACCEPT statements work for terminals versus other I/O devices.

One major enhancement has been the addition of multi-key ISAM and segmented key support. This was a hotly debated issue, since Digital maintained that, if this were included in the standard, Digital would not be able to offer an ANS DIBOL for CTS-300. This was a very key issue, since it had been Digital's intention to produce an ANS compliant DIBOL for all its implementations. Despite the potential drawbacks of this, the committee determined that we could not, in good conscience, produce a standard without these features.

Error handling also received a lot of attention. The most important result is the creation of error mnemonics instead of error numbers as part of the standard. Now, instead of trapping for error #1, end of file, the programmer can trap for error number $ERR_EOF. This was considered a major victory, as it solved the problem of which error numbers to allocate to this and future standards, and which to allocate to the language implementor's for specific non-standard implementations of the language. It also has the side benefit of producing more readable code and making it easier on the programmer.

A number of new statements have been added, most of which have been drawn from VMS DIBOL version 3. These are DECR, EXIT, EXITLOOP, NEXTLOOP, REPEAT and XRETURN.

The number of external subroutines that are part of the standard has been reduced considerably, the feeling being that many of the existing ones were highly implementation specific and should not be part of the standard. Only one new standard subroutine has been added, FILEC. This provides for a standard method of creating ISAM files from within a DIBOL program, and was considered to be a much needed addition to the language.

The above represents a distillation of over 20 days of committee meetings spread out over a period of nearly three years. The committee members all feel justifiably proud of their work, and will now focus on language extensions for the next version of the standard.

Ken Schilling

GLOBAL ACCESS

MMP

MUMPS SIG DECUS U.S. CHAPTER

"If you don't want it in the paper, don't let it happen."

DECUS

GLOBAL ACCESS

Volume 2, No. 2 September, 1988

MUMPS means you never have to say you're sorting.

$VIEW(Editor)

Have you ever heard of the "Law of Mean Objects?" It comprises several corollaries to Murphy's Law which reflect the tendency of inanimate objects to behave in ways which maximize the negative impact of their actions. For example, the probability of a traffic signal changing to red approaches one as your distance from it approaches zero. And, coins roll away from you with a velocity proportional to the square of their value. Well, the Communications Committee (that illustrious DECUS Operating Unit which is responsible for bringing you this splendid example of editorial excellence each month) conducted a meeting last weekend, and the Mean Objects conducted a war.

In relation to the whole committee, the Objects' major weapon was the weather. I drove out of Central New York in a 1/2 inch per hour monsoon (the storm drains were spouting columns of water a foot high into the streets, knocking manhole covers off into the path of traffic), which played tag with me all the way to Massachusetts, and then camped there for the entire weekend. I arrived at midnight; some who flew didn't make it until 3:30AM, and one spent the first night in Logan Airport. To make matters worse, on the second night the storm caused a power failure. This rotated everyone's alarm clocks into the twilight zone, and made the schedule that morning more than a little ragged.

In my own case, the Objects enlisted my car as the agent of their wrath. Sixty miles from home, the car broke down. The mechanic who towed me off the Interstate tried to find the parts, but struck out. I worked a little harder, and managed to find a set of parts. However, when the mechanic tried to install them, they didn't fit. At that point, he gave up in despair, and I managed to get somebody else to tow the car home.

In spite of the hassles, the meeting was very productive and, overall, relatively enjoyable. I just hope that next year we can pick a weekend when the Mean Objects are on vacation.

$DATA

It got away from me! At the time I wrote the last issue, I completely forgot that the Symposium will be early, and that I had to put the schedule in THIS issue. However, dawn broke over Marblehead in time enough to get it in, so here it is:

General Interest

What is MUMPS? --An Introductory Overview	MU010	Mo	9:00AM
DECUS Symposium On-Line Abstract Keyword Index	MU001	Mo	10:00AM
Voice Synth. Xface for a MUMPS Application	MU006	Mo	7:30PM
Large MUMPS Systems Under VMS	MU022	Mo	8:30PM
MUMPS Source Code Management	MU028	Tu	5:00PM
MUMPS Development Committee Report (ANSI X11)	MU021	Tu	5:30PM
MUMPS Users' Group Roundtable	MU019	We	12:00M
MUMPS Portability Panel	MU026	We	3:00PM
VAX MUMPS Tuning Panel	MU025	We	4:00PM
MUMPS SIG Business Meeting	MU018	We	5:00PM

Tutorials

Beginning ANSI MUMPS Programming	MU011	Mo	1:00PM
Intermediate ANSI MUMPS Programming	MU003	Mo	2:00PM
MUMPS Programming Workshop	MU004	Mo	3:00PM

Public Domain Software

Rule Man, a Heuristic Expert System	MU005	Mo	9:30PM
MUMPS Submissions to the DECUS Library	MU015	Tu	2:00PM
Intro to VA Kernel and File Manager S/W Tools	MU027	Tu	3:00PM

DSM Specific

VAX DSM Product Update	MU016	Tu	9:00AM
MUMPS Clustering/Networking Considerations...	MU007	Tu	10:00AM
Coexistence/Migration Features DSM-11/VAX DSM	MU017	Tu	11:00AM
DASL--A 4GL Application Generator Clinic	MU008	Tu	1:00PM
Interfacing MUMPS to FMS	MU012	Tu	4:00PM

Third Party Software

Medical Query Language (MQL): A Query Language and Report Generator	MU009	Tu	4:30PM
GT.M, A Novel VAX MUMPS Implementation	MU024	We	9:00AM
Micro to VAX Distributed Systems	MU020	We	10:00AM
ANSI SQL and ANSI MUMPS Rel. DB Capabilities	MU013	We	11:00AM

Please observe that we have been careful not to schedule MUMPS sessions on Thursday and Friday.

Democrats Pick MUMPS Reservation System

Delegates gathering in Atlanta in July for the Democratic National Convention were greeted by a quick and efficient MUMPS-based reservation system, developed and installed by Registrar Data Group Corporation (RDG) of Owings, MD. Selected by the Democratic National Committee, RDG's Housing Bureau Reservation System managed about 40,000 room reservations, and unlimited bookings for multiple meetings in over 100 hotels.

The powerful R/SCAN-M databse system was devloped specifically for the association industry to handle meeting requirements, housing bureau services, membership records, and exhibitor space management.

Unlike other major U. S. cities, Atlanta did not have a computerized housing bureuau. RDG has permitted the Democrats to give the license for their housing package to the city of Atlanta after the Convention.

--Excerpted from MUMPS NEWS, June, 1988, vol. 5, no.3

[The Steering Committee would like to thank the MUMPS Users' Group for granting us permission to reprint items from their newsletter. --Ed.]

$HOROLOG

September 24	Submission deadline for November newsletter
October 17-21	Fall '88 Symposium; Anaheim, CA
February 8-10	DEXPO East '89; New York, NY
February 20-24	Canadian '89 Symposium; Vancouver, BC
May 8-12	Spring '89 Symposium; Atlanta, GA

$ORDER("Speak to")

Lately it has become faddish among government officials, corporate officers, and others who consider themselves of great import, to use the verb *to speak* in a twisted fashion: "the Secretary will *speak to* the issue of nuclear disarmament." They seem to think it makes them sound sophisticated; I think it sounds pompous and foolish. Just try substituting *talk to* in the sentence above, and you will see what I mean. Besides, both my magic internal compass and my mundane external dictionary state quite clearly that, while you can *speak of, speak about, speak upon,* or even *address* a topic, you can only speak to an entity (preferably another human being). Fortunately, fads tend to be of finite duration, so I will just find a way to endure until this one goes the way of hula hoops, tail fins, and leisure suits.

$NEXT

For those who will not be able to make the Fall Symposium, the November issue will contain a description of the MUMPS submission that Mike McIntyre is putting together for the DECUS Library.

$NEXT($ORDER)="different than"

$RANDOM

"What you have here is an Administration that has set its hair on fire and is trying to put it out with a hammer." --Senator Alphonse D'Amato

OFFICE

AUTOMATION

OA

IN THIS ISSUE...

FROM THE EDITOR...

Drum Roll Please.... Ladies and Gentlemen, and ALL-IN-1 users everywhere: Announcing in this month's issue an article from DEC on "The Next Release" of ALL-IN-1!!!

I just couldn't resist doing that. After so much talk about the mysterious 'future unannounced release' we are pleased to be able to share with everyone the highlights of a session given by DEC in Cincinnati that outlines the major changes in the next version.

We also have an OA SIG Tape update from our very busy Tape Coordinator, Roger Bruner. Add to that two terrific technical articles: One on Document Privacy for those of you concerned with document security, the other, some shortcut steps for entering commands interactively.

I want to thank all of you who have sent me articles in the last two months, WOW! I have so many great articles right now it will take me a few months to get them all published. But please, don't let that stop *you* from sending in an article. The more articles we have waiting to be published, the higher the quality of our portion of the newsletter, so keep up the good work everyone.

Regards,

Therese LeBlanc
OA Newsletter Editor
275 London Place
Wheeling, IL 60090

ALL-IN-1 ... FUTURE FUNCTIONALITY

Bob Malay, Digital Equipment Corp.

The following is a summary of the session "OA Update Seminar for ALL-IN-1: The Next Release" which was presented at the Cincinnati Symposium in May, 1988.

NEW / IMPROVED FUNCTIONALITY

PRODUCTIVITY IMPROVEMENTS

- READ
  Scroll Forwards / Backwards
  Next / Previous Screen Key
  GOLD Up / Down
- BACKGROUND FORMAT / PRINT
- INDEXING
  Work on single or multiple items
  Remain within the index

SYSTEM MANAGEMENT

- INTERFACE CHANGES
  Menu options grouped more logically
- NON-PRIVILEGED SYSTEM ADMINISTRATOR
  Management of User Accounts
  Maintenance of System Distribution Lists
  Scheduling of Housekeeping Events
  Empty Wastebaskets
  Archival of documents
  Reorganize accounts / file cabinets
  Update new / old Mail addresses
  Cancel scheduled Housekeeping procedures
  Monitor ALL-IN-1 Exception Reports
  Initiate ALL-IN-1 Shutdown
  Site Definable
- MAINTENANCE OF USER ACCOUNTS
  Create User Account - Batch
  Delete User - removes all ALL-IN-1 references
  Rename User Account - Single step operation
- METERING
  Number of times metering called
  CPU usage
  Direct I/O
  Buffered I/O
  Page Faults
  Application specific metered sources
- IMPROVED HOUSEKEEPING
  Automatic Scheduling and Re-scheduling
  No job clashing
  Logs Simplified
  More versatile File Cabinet Verify and Repair

- SPLIT MULTIPLE SHARED DOCUMENT ATTRIBUTES FILE
  Maximum of five areas
  Each area is the same as the SDAF
- MANAGEMENT OF SHARED AREAS
  Tools
  Shared Area Manager
  Message Quotas based on total blocks
  Archiving

ARCHIVING

- USER
  Control which documents/folders to include or exclude
  Request retrieval
- SYSTEM MANAGER
  Creates archive area(s)
  On-Line and open
  On-Line and closed
  Off-Tape
  Control timing of archive / retrieval process
  Force archive
  Exempt users
- SYSTEM ADMINISTRATOR
  Institute an off-schedule run
  Cancel the next scheduled run

WPS-PLUS

Columnar movement
Status Line
Insert / Overstrike
Enhanced Footnoting
Table of Authorities
Word Indexing
Optional Grammar Checker

TIME MANAGEMENT

- NEW INTERFACE
  Meetings/ Appointments — -> EVENTS
  Reminders/Action Items/To-Do — > TASKS
  Multi-Node Scan
  Editable calendars

MAIL

MAILbus
Core component of ALL-IN-1
ALL-IN-1 can utilize the Message Router Gateways
ALL-IN-1 takes further advantage of new functionality - Distributed Directory Service - addressing by full name rather than account name
Mail Directory (Distributed Directory Services)
DDS not used
A1 Prime
DDS Prime
Recommendation - A1 Prime
Support store / forward of unknown document types
Usability Improvement
Cover Note can be specified or not on a forward

File -Message, -Attachment, -Text allow keyword entry during filing
Auto-forward / Destination address will be announced during LOGIN

INTERNATIONALIZATION

Multilingual Support
Directory structure changed to allow language dependent and language independent files to bestored
System logicals extended for multilingual system support
Eight-bit character support
International date / time formats

DOCUMENTATION

- ALL BOOKS HAVE BEEN RE-ISSUED AND A NUMBER REORGANIZED
  User's Reference
  WPS-PLUS Books
  System Manager's Guide
  Administration Guide
  Management Guide
  Mail Management Guide
  Application Programming and Integration
  Application Programming: Getting Started
  Application Programming: Guide
  Application Programming: Reference
  Application Programming: Index

CUSTOMIZATION MANAGEMENT

- SEPARATION OF CUSTOMIZATIONS FROM THE ALL-IN-1 BASE KIT
  Base Kit area
  Customized Area
  Development Area
- STRUCTURED MANAGEMENT OF CUSTOMIZATION TO THE SYSTEM
  Develop new elements
  Modify Base Kit Elements
  Test Elements
  Track Changes

Easier identification of customizations
Customizations maintained across upgrades
Customizations have priority

PRE-INSTALLATION TOOL

VMSINSTALable
Identifies customizations of previous ALL-IN-1 versions
Creates a Customization Directory Structure (CDS)
Run before ALL-IN-1 Upgrade

CUSTOMIZATION MANAGEMENT SUBSYSTEM

Provides structured approach to manage customizations
Provides records of modifications
Provides audit trails for modifications

PERFORMANCE

- GOAL
  Comparable Levels with ALL-IN-1 V2.2
  Printing / Reading
  User Productivity Gains - Scroll, Indexing

"ALL-IN-1 IS A JOURNEY ... NOT A JUST A DESTINATION"

OA SIG TAPE UPDATE

Roger Bruner, Foreign Mission Board

Even though the Foreign Mission Board no longer distributes the OA SIG TAPE, we still get lots of calls about how to order it.

That's great! We're glad the tape is of such interest to you and that it has some of the things you tell us you need!

But wouldn't it be great if you could contribute something to the next tape (due after Anaheim) — something someone else may need? It doesn't have to be anything fancy or elaborate. It might even be your own variation of a procedure you've seen elsewhere. There is an extensive (but not exhaustive) listing in last month's newsletter of suggested ideas for tape submissions.

We have already received Dale Coy's updates to his previous submissions, including his WPE editor, which you need to see and use to appreciate. We have been trying WPE out for the past few months and are considering the possibility of its use in a setting where users accustomed to WPSPLUS will be editing in a non ALL-IN-1 environment. For them, WPE may just be the answer!

I intend to include several command procedures we use here whenever we modify the MEMRES or OAFORM libraries. There is a "long" version which runs on one node of the cluster and a "short" version to run on the other nodes. Not only is it a time-saver, but it also prevents human error!

Another procedure I hope we can include is one which checks the size of the current Shared Directories and makes it real easy to either reassign OA$SHARE_LOW & OA$SHARE_HIGH to existing directories which have thinned out some or to create new Shared Directories.

These are just a few examples of what I hope the new tape will contain. But there's plenty of room for your submissions, too.

Remember, there is an October 31 deadline. Send your tapes to:

Roger Bruner
MIO
Foreign Mission Board
Box 6767
Richmond, VA 23230

If you have questions, please call me at 804/353-0151, extension 606.

DOCUMENT PRIVACY IN THE ALL-IN-1 ENVIRONMENT

George B. Bone

How secure is that document you just placed in your ALL-IN-1 file cabinet? Well, you didn't file it using a shared folder ([<username>]<folder-name>), so it must be private, right?

Not Quite. Although it is true that ALL-IN-1 will not let others access a document that has not been filed as 'shared', the protection of that document at the VMS operating system level is a function of the UIC protection and Access Control List (ACL) assigned it. If you haven't modified the default, the protection of a file in a directory is S:RWED, O:RWED, G:RE, W:none. This means that members of your 'group' will have READ access to your document if they go through the VMS level. Of course, the document is stored in sub-directories and is named ZRLKFJUJV.WPL or some such gibberish, so the fact that someone could read your document doesn't bother you because they can't find it, and won't know what it is if they do find it.

Or will they? Check the protection of a file in your ALL-IN-1 directory called DOCDB.DAT (use DIR/SEC DOCDB.DAT). If the default hasn't been modified, your Group will have READ access to this file. Now try to display it using the TYPE command. Go ahead, it won't hurt.

Oh my. Looky there. There is a listing of the folder, the document title, and the file name (including sub-directory). Everything that someone would need to find a document in your file cabinet. In fact, at our site, we have written a utility that will list this information in a more readable format. Of course, we only use it when someone has gone on vacation and has forgotten to file a needed document as shared.

It's okay, though. If you were to try to TYPE this file (the ZRLKFJUJV.WPL file), it would be all gibberish and beeps and 'oh, shoot, I have to reset the terminal'. We're safe.

Again, not quite.

On the Word Processing (WP) menu there is an option called Document Transfer (DT). On the DT menu is an option called Receive from VMS (RV). When called, this option prompts for the name of the VMS file (remember the TYPE DOCDB.DAT we did?), then prompts for a document folder and title. When it is finished, that document (your document!) has been copied into the file cabinet (someone else's file cabinet!). It is now readable and printable.

Incidentally, a similar operation may be performed using the GOLD G(et) function while in the WPS-PLUS editor. Same tune, different harmony, same end. Your document read by somebody it wasn't intended for.

But so what? After all, we're all in the same group! We have no secrets!

What if you are a supervisor writing performance evaluations?

What if you are a personnel clerk preparing a report containing PRIVACY ACT information?

Hmmm.

So, what do we do about this? An easy fix (comparatively) would be to modify the protection on all of the files in your file cabinet (including DOCDB.DAT and DAF.DAT) so that group and world have no access. This would mean setting the default protection on each of the [.DOCx] directories to group and world none, also, so that future documents would be protected.

This would also mean that your shared folders won't work anymore. This could be bad, depending on how much you use them.

What to do? Another fairly easy solution is to add the following named data items to your WP menu (and the appropriate items on the layout screen):

```
;;SHP;;

get oa$dcl="set protection=group:re " oa$curdoc_filename\get oa$display=
"document " oa$curdoc_docnum " has been set to shared protection"\force

;;NSHP;;

get oa$dcl="set protection=group " oa$curdoc_filename\get oa$display=
"document " oa$curdoc_docnum " has been set to no shared protection"\force
```

```
;;DSP;;

get oa$function="spawn dir/sec " oa$curdoc_filename \wait
```

SHP, when invoked, sets the protection of the current document such that members of your group may read it. NSHP revokes that ability (sets the protection to no access for group). DSP simply displays the protection of the current document. Further enhancements could provide a means to set up specific access control lists for your documents, but that is a bit beyond the scope of this article.

Protecting the privacy of documents created in the ALL-IN-1 environment is not something that happens automatically. It's easy to be lulled into complacency by the file cabinet software, but the file cabinet protection can be easily subverted by using standard VMS commands. Ensuring that your documents are protected at the VMS level as well as the ALL-IN-1 level will keep you just a little saner, and your users a little happier.

Happy ALL-IN-1ing!

A LAZY MAN'S GUIDE TO ENTERING ALL-IN-1 COMMANDS INTERACTIVELY

Bob Wilkinson - Naval Underwater Systems Center
Bill Sherman - A&T Technical Services

It seems that here at NUSC, we take interactive programming seriously. No matter how much forethought we put into a new program, it takes at least 600 iterations of: $edit [stuff.test.project]fiddle.scp to get anything to really run. We also seem to find ourselves doing things like: <form auto view named_data and <oa$menu_show_stack a lot, i.e. if it takes a lot of accurate typing to make it work, we do (and redo) it a lot!!!

The inconveniences of constantly retyping commands started us thinking in terms of PC amenities like Superkey or macros, and wishing that ALL-IN-1 had something similar. UDP's are nice, but a bit clumsy, and putting things in named data on form DEFAULT means re-linking ALL-IN-1 every time you need something new. Eventually, the light dawned! We could store command strings in permanent symbols, and access them through permanently assigned GOLD function keys defined on form DEFAULT as follows:

```
;;.GOLD 1;;

GET OA$FUNCTION = $user_func_1

;;.GOLD 2;;

GET OA$FUNCTION = $user_func_2
.
.
.

;;.GOLD 9;;

GET OA$FUNCTION = $user_func_9
```

DEC doesn't appear to be using these particular GOLD sequences anywhere right now, and although these commands won't work *inside* a WPS-PLUS document, our primary need was for program debugging and other developer tasks. You write a "macro" by interactively loading the desired command string into the appropriate permanent symbol:

```
<GET $user_func_1 = "oa$menu_show_stack"
<GET $user_func_2 = "form auto view named_data"
```

We also wanted another scratch pad that was faster to access and less volatile than the regular one. This same method gave us a quick solution:

```
<GET $user_func_9 = "edit 'memo.wpl'"
```

This gave us a WPS-PLUS format "scratch pad" that works for recording ideas practically anywhere except *inside* a WPS-PLUS document.

This idea could be extended with a ARG form that would let you look at all your assignments at once and/or edit them on the screen. We've found this little trick to be quite a time saver, and hope others will find it useful.

PERSONAL COMPUTER SPECIAL INTEREST GROUP

NEWSLETTER

PC

TABLE OF CONTENTS

PRO Section

Rainbow Section

PRO Section

Using an ST225 Drive with P/OS v2.0A

By John Preston, Centre for Nuclear Sciences, University of the West Indies, Kingston Jamaica, West Indies

I was first introduced to the possibility of using the ST225 drive on my Professional 380 by two articles that appeared in the DECUS Newsletter in June '87 and February '88. Having purchased one, sometime later I proceeded to discover the differences in the driver code between P/OS v2.0 and P/OS v2.0A. My 380, having been purchased within the last two years was bestowed with the most recent disk driver ROMs and thus I proceeded to simply connect the ST225 and install the Operating System. I was anxious to see what the outcome would be without any tinkering.

Well, after I turned on the computer and the self test ran, I got a picture of the computer high-lighting the disk driver board. I had read earlier that without a preformatted disk then the controller would have a problem recognizing the disk. However, this did not inhibit the further installation of P/OS and thus I continued the normal formatting and installation procedure. After an uneventful hard disk format, I discovered that only 10 megabytes was available. What to do? I proceeded as my predecessors had, searching through an octal dump of the POS.SYS file, looking for the head, cylinder and total block sizes for the RD51 (10 megabyte disk). I found two different locations with this information. Without having a DW driver listing, I was unable to see where the initializations for the different disks began and ended. I reasoned however that if the system saw the disk as 10 megabytes then I could possibly change the information for the 10 megabytes to configure it as 20. After a number of false starts, I was successful and got 39295 blocks available.

I changed the following locations in POS.SYS:

157:734 from 046077 to 114577
1041:460 from 000461 to 001146
1041:464 from 046077 to 114577

This solution may not be the correct way to go about it, but it works for me. I created a file in FORTRAN fo 36,000 blocks for testing and had no problems. If anybody has P/OS v2.0A and wants to use a more economical disk upgrade than DEC's own, this method works.

PROgramming Quickie - Converting Quad Word Dates to/from ASCII

By Bart Z. Lederman, Brooklyn, NY

Editor's Note: This Quickie originally appeared as an article in the October 1985 issue of these Newsletters. I have abridged the article and added to it somewhat so that it is specific to the PRO.

The quad word date format used as the DATE data type in PRO-DATATRIEVE can be easily read by DATATRIEVE, but there are times when the data must be processed in another language (sad, but true). For example, you might want to maintain, query, and report a large amount of data with DATATRIEVE, but want to do number crunching on it with FORTRAN for some functions which are not easily implemented in DATATRIEVE like Fourier Analysis, Analysis of Variance, Matrix inversion, or whatever. Most PRO languages don't have any provision for dealing with quad word integers, particularly dates expressed as clunks, and especially when the order of most significant to least significant word goes in the opposite direction from normal P/OS integers.

If you want to convert a date in ASCII characters to a quad word value in clunks, there are several methods. The routine to convert a quad word binary time to an ASCII character string is called $CDTTA, and it used to be in Appendix C of the RMS-11 Macro-11 Reference Manual (AA-H683A-TC) supplied with RMS-11 Version 1.8; but it is not in the manual for version 2.0, the current version of RMS (the version you use regardless of which version of P/OS or PRO/Toolkit you have). I don't know why it's not documented anymore, but it is still supplied. If you have old RSX manuals (see, there is a reason for keeping all of that old stuff heaped up in your office), you can look up the routine there. The source listing in the old manual is slightly out of date in that some of the other routines called have been replaced by standard System Library routines, but the basic mode of operation is unchanged. Knowing that there was one undocumented conversion routine aroused my curiosity, and when I started looking around I found a routine which goes the other way, converting an ASCII character date (and time) to a quad word integer in clunks. For those of you who don't have the old manual and for the routines that were never documented, I will give a description of how to use the routines from several languages (MACRO, FORTRAN, BASIC, etc.).

If you are writing programs in MACRO-11, you can call the routines directly. The procedure is just like other system library formatting routines such as $CBTMG, $CBTA, etc. Addresses of buffers are loaded into registers, and the routine is called.

If you are programming in FORTRAN then you cannot call these system routines directly. To solve this, I wrote my own subroutine which makes the $CDTTA and $CATDT routines available to FORTRAN programs. I am including the entire subroutine later in this article. The actual routine is so simple that it is shorter than the documentation, and most of that is argument error checking. If you were willing to assume that the people using the routine would always pass exactly the right arguments, the routines would only be three or four instructions long, but I think it's better to check anyway, just in case. The CDTTA routine has an optional flag so you can choose if you want the date, time or both, and times with and without seconds. This flag is set as shown in the MACRO routine documentation for R2, also at the end of the article. If you leave out the flag, the routine will convert the date portion only, and it is a limitation of the system routine that it always converts the year as four digits (i.e., 1985 rather than just 85) so you should always allocate enough space for it. You don't need to know MACRO yourself to use this routine. Assuming you typed in the macro source with the file name QUADAS.MAC, simply assemble it with the command:

```
$ MACRO QUADAS
```

with PRO/Toolkit and include QUADAS.OBJ when you task build. I haven't passed back the conversion error indicator, but you can easily test for errors yourself as follows: before conversion, zero or blank out the output area. If after calling the conversion routine the output area is still zero or blank, the conversion failed.

The most common cause of error is a syntactically incorrect date (such as asking for 30-FEB-85 to be converted to binary). When calling CATDT, the date must be in the format:

```
        HH:MM:SS NN/DD/YY or
        HH:MM NN/DD/YY or
        HH:MM:SS DD-MMM-YYYY or
        HH:MM DD-MMM-YYYY
```

The routine will test to see which version of date is used by looking for "/" and "-" characters and will check to see if the time includes seconds (it can be HH:MM only). Also, the year here can be two digits (assumed to be between 1900 and 1999) or four digits. IMPORTANT: if you use the 20-JUL-85 date format the month MUST be in UPPER CASE. The conversion routine does not convert lower case to upper case (though it is my opinion that it should). If you are interested in dates only, just put in 00:00 as the time and it will work from midnight as DATATRIEVE does. In calling the CDTTA routine, almost any positive number stored as a quad word integer should translate to a date without error. I have included a sample program calling these routines from FORTRAN in this article.

If you are using BASIC-Plus-2, then you can also use this FORTRAN callable routine. You will have to use the

```
CALL xxx BY REF (arg, arg)
```

format to pass the values properly; the BASIC manuals should tell you how to do this.

I have limited experience with other DEC languages, but a check of some manuals indicates that they also use the FORTRAN calling convention. If they can call system services such as ASNLUN, CLREF, WTQIO, etc., then they can call these routines. COBOL-81 should work with the construction

```
CALL "xxx" USING arg arg
```

and DIBOL should work with an XCALL. PASCAL can also call routines if they are defined as external. In all cases you should consult the appropriate language documentation.

Some DECUS languages such as "C" and some of the PASCALs should also be able to do this, or could be modified to call the system routine directly, though I haven't yet looked into this myself. "C" sometimes has the facility to place values and addresses into known registers, which would allow you to call the routines in the same manner as MACRO would, otherwise the FORTRAN calling convention should be used.

Subroutine to make Quadword/ASCII conversion callable from FORTRAN:

```
        .TITLE QUADAS
        .IDENT /V1.0/
;
; B. Z. Lederman 22-Jun-85
;
; This FORTRAN callable subroutine makes the $CDTTA and $CATDT routines which
; convert between Quadword Binary date/time values in clunks (such as values
; from VMS, DATATRIEVE, and System Accounting) and ASCII strings available to
; FORTRAN-77 programs and other languages using the FORTRAN
; calling convention.
;
; These routines use R0, R1, R2, and R4. They don't save or restore any
; registers because the FORTRAN compiler assumes they will all be destroyed.
;
; Errors are handled in the simplest manner. If the routine is called with
; the wrong number of arguments, it simply does nothing and returns. If the
; conversion fails, the system routine returns with carry set, but FORTRAN
; can't check this. The easiest way to check for errors is to zero your output
; area before calling and check it after calling: if it's still zero, the
; conversion failed. The most likely cause is an incorrect date format to
; CATDT: nearly any positive integer passed to CDTTA will result in a date.
; Note that the routines have no way of checking to see that you have
; allocated sufficient space for the output data, so if you haven't, one of
; your other variables will be clobbered.
;
;
; CDTTA: Convert Quad Word to ASCII date/time
;
; CALL CDTTA( IN, OUT, [FLAG])
;
; IN is a four word Integer*2 array (or equivalent) which holds
; the 64 bit binary time.
;
; OUT is a byte array (at least 11 bytes long) to hold the
; converted date. It must be this long because the year is
; four digits (as in 12-JUN-1985).
;
; FLAG is an optional argument to specify the items converted
; (see documents on $CDTTA): if absent, only the date (no ; time) is converted.
;
.PSECT          $CODE2, RO, I
CDTTA::         MOV   (R5)+, R4       ; get number of arguments passed
                BEQ   50$             ; if no arguments, quit
                CMP   R4, #2          ; are there at least two arguments?
                BLT   50$             ; quit if no
                MOV   (R5)+, R1       ; retrieve address of input data
                CMP   #-1, R1         ; was this a dummy argument?
                BEQ   50$             ; no dummy arguments allowed here
                MOV   (R5)+, R0       ; retrieve address of output area
                CMP   #-1, R0         ; was this a dummy?
                BEQ   50$             ; quit if yes, no dummies here either
                CMP   R4, #2          ; were there only two arguments
                BEQ   20$             ; branch if yes (third is optional)
                CMP   R4, #3          ; double check for 3 arguments
                BNE   50$             ; not valid unless 2 or 3 arguments
                MOV   (R5), R4        ; get address of flags
                CMP   #-1, R4         ; is it a dummy?
                BEQ   20$             ; yes, fill in default flags
                MOV   (R4), R2        ; use the callers flags
                BR    30$             ; skip our flags
20$:            MOV   #2, R2          ; flag conversion of date only
30$:            CALL  $CDTTA          ; call system routine
50$:            RTS   PC              ; all done

; CATDT: Convert ASCII time and date to Quad Word:
; (Must have a time, even if it's 00:00)
;
; CALL CATDT( IN, OUT)
;
; IN is a byte array which holds the time and date to
; be converted
;
```

```
; OUT is a four word Integer*2 array (or equivalent)
; which holds the 64 bit binary time.
;
 CATDT::        CMP    (R5)+, #2        ; were there two arguments?
                BNE    150$             ; quit if not exactly two arguments
                MOV    (R5)+, R0        ; retrieve address of input data
                CMP    #-1, R0          ; was it a dummy argument?
                BEQ    150$             ; quit if dummy
                 MOV   (R5), R1         ; retrieve address of output area
                CMP    #-1, R1          ; was this a dummy?
                BEQ    150$             ; quit if dummy
                CALL   $CATDT           ; call system routine
150$:            RTS   PC               ; all done

; GDTIM: Get the System Date/Time as a quad word
;
; CALL GDTIM( OUT)
;
; OUT is a four word Integer*2 array (or equivalent)
; which holds the 64 bit binary time.
;
GDTIM::         CMP    (R5)+, #1        ; was there exactly one argument?
                 BNE   250$             ; quit if no
                MOV    (R5), R1         ; retrieve address of output area
                CMP    #-1, R1          ; was this a dummy?
                BEQ    250$             ; quit if dummy address
                CALL    $GDTIM          ; call system routine
250$:            RTS   PC               ; all done

                .END
```

```
      PROGRAM TST
C
C show conversion from ASCII to quad word date to ASCII
C
      INTEGER QUAD(4), FLAG
      BYTE DATEIN(20), DATOUT(20)
C
C Clear out the date strings.
C
10     DO 20 I = 1, 20
             DATEIN(I) = ' '
20    DATOUT(I) = ' '
C
C Prompt the user for a date
C
      WRITE (5, 30)
30    FORMAT(' Enter a date string [HH:MM(:SS) MM/DD/YY or' /
     1 ' HH:MM(:SS) DD-MMM-YYYY]' /
     2 '$ or RETURN for the current system date: ')
      READ (5, 40, END = 999) I, DATEIN
40    FORMAT( Q, 20A1)
C
C Re-set the quad word date
C
```

```
        QUAD(1) = 0
        QUAD(2) = 0
        QUAD(3) = 0
        QUAD(4) = 0
C
C Check the number of characters input
C
        IF (I .EQ. 0) THEN
C
C Get the system date/time as a quad word
C
                CALL GDTIM( QUAD)
C
            ELSE
C
C Convert the ASCII input to a quad word
C
                CALL CATDT( DATEIN, QUAD)
 C
        ENDIF
C
C Show the conversion
C
        WRITE (5, 50) QUAD
50      FORMAT( 4(3X, I7))
C
C Convert back the other way (default flags)
C
        CALL CDTTA( QUAD, DATOUT)
C
        WRITE (5, 60) DATOUT
60      FORMAT (1X, 20A1)
C
C Demonstrate different formats with conversion flags
C
        DO 100 FLAG = 0, 15
C clear the output area
                DO 70 I = 1, 20
70              DATOUT(I) = ' '
C
                CALL CDTTA( QUAD, DATOUT, FLAG)
C
                WRITE (5, 80) FLAG, DATOUT
80              FORMAT (1X, I3, 1X, 20A1)
C
100     CONTINUE
C
        GOTO 10
C
999     CALL EXIT
C
        END
```

Description of MACRO Interface to Library routines:

$CBTTA - Convert binary Quad word date/time to ASCII string
Input: Output:
R0 = address of output buffer
R0 = points to next byte in output buffer after converted string
R1 = address of quad word to be converted
R2 = conversion flag:
bit 0 = 0 -> convert date portion
bit 0 = 1 -> do not convert date
bit 1 = 0 -> convert time portion
bit 1 = 1 -> do not convert time
bit 2 = 0 -> time in 24 hour format
bit 2 = 1 -> time in AM/PM format (might not be functioning)
bit 3 = 0 -> convert time in seconds
bit 3 = 1 -> do not include seconds in time
Call:
CALL $CDTTA

$CATDT - Convert ASCII string to binary Quad word date/time
Input: Output:
R0 = address of input string
R0 = points to next byte in input buffer after converted string
R1 = address of quad word to receive value
R2 = not used
Call:
CALL $CATDT

$GDTIM - Retrieve system date/time as binary Quad word
Input: Output:
R0 = not used
R1 = address of quad word to receive value
Call:
CALL $GDTIM

Rainbow Section

102 Laptop to Rainbow--a Revisitation
By Tom Warren, PC SIG Session Notes Editor

Anne-Marie Pollowy Toliver reports using her Radio Shack Model 100 laptop/notebook computer to transfer files to her Rainbow. I want in this note to update this information based on the Model 102. Anne-Marie gave settings for Poly-Com (see below) and her laptop. She commented on needing to modify the Inmac cable to fit her RS-232 port. When I got my cable, I made no modifications to either the cable or the RS-232 except to change to longer screws.

She also commented on the slow transfer from the 100 to the Rainbow--a single line at a time overwritten. So far, I have not solved this problem. Perhaps a higher speed of sending from the 102 to the Rainbow will help.

Also in Poly-Com, "Select" "R" opens a file you name, and "Select" "C" closes the file. Once I had my file on a Rainbow disk, I could import it to either WordPerfect ("Text-In/Out") or Samna Word III ("Translate"). Then, I could reformat or whatever I wanted to do.

When I sent a file from the Rainbow to the 102, the text scrolled through on the 102's screen. One problem I did have was in naming files: DOS takes 8 + 3 letter/number characters; the 102 takes 6 + 2, and if you weant to do word processing on the 102, the extension must be "DO" because that's the extension the text processor reads.

Is it worth the hassle to transfer files from the 102 to the Rainbow? Yes. I can work many more places with the 102 and then load into the Rainbow for finishing. Also, I don't have to turn the Rainbow on in the morning if I want to do a quick memo. I highly recommend the 102 as a sound investment for Rainbow owners. (Actually, I can also load into my PC-XT at the office and can carry files back and forth--a kind of portable I-Drive. I do have to be careful of file size because I normally have only 25K of free memory in the 102.)

SPECIFICATIONS;

RAINBOW--using Poly-Com:
7 Stop bits
1 Parity
Method: Full Duplex to Host
Xon-Xoff
TX Speed 300 bps
Throttle 0 cps
from host Xon-Xoff
Console Mode ANSI
New Line CR LF
User Defined Keys Answerback
Host prompt None
EOF String None
Method Blind
EOL Sync None
EOL Delay 0 ms
CR Action Cr Cr

TANDY MODEL 102
Set Telecom to 37I1E, 10pps

After getting evertything set, I open a file on the Rainbow ("Select" "R"), use TERM[inal] for the 102 and set the line width to 64 (Anne-Marie suggested a much longer line length, but 64 means less reformatting for me.)

After sending the file, press "BYE" on the 102 and "Y[es]" for "Disconnect"? On the Rainbow, close the file ("Select" "C") and you're ready for a word processor.

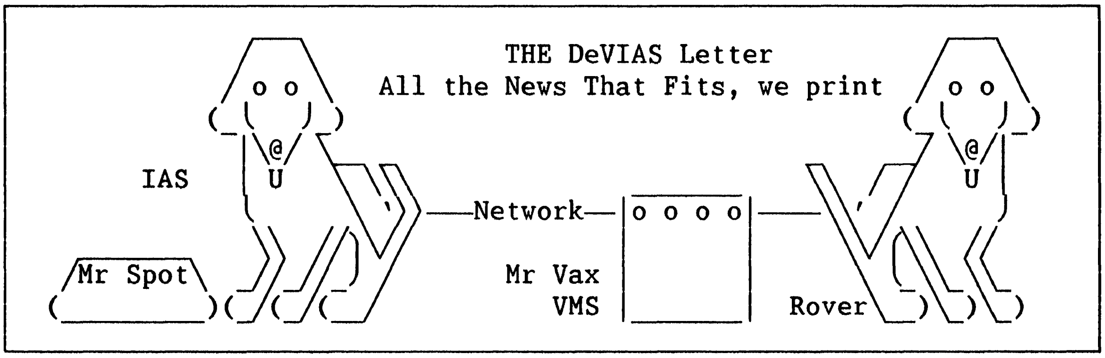
Inmac Cable Order Number 1964, null modem, M-F connectors, 8 foot. I found that I got the cable within two days after placing the order (the company has 15 regional offices). Their phone is 1-800-826-8180.

CITATION

Toliver, Anne-Marie Pollowy. "From Laptop to the Rainbow," PC-SIG NEWSLETTER, 1, No. 7 (March, 1986), PC-71 to PC-73.

THE DEVIAS LETTER

RSX/IAS



FROM THE EDITOR'S KEYBOARD

The following editorial is solely the opinion of the author, and does not represent the views of DECUS or of Digital Equipment Corp. Responses to the Editor's remarks are heartily welcomed.

A funny thing happened to me on the way to last month's issue. Yours truly, old "Never been sick a day in my life...", ended up in the hospital with pneumonia and some other lung infection, (the doc still isn't sure what else I had.) Anyway I am back in the saddle, (and feeling a lot better than I have in a long time. Just wish I had not lost a months worth of summer.)

Needless to say, two weeks in the hospital and two more convalescing at home blew last month's issue, It also cut down on what I could prepare for this one, especially since I had to prepare for the annual CommComm meeting this past weekend. Things should be back to normal next issue.

I have just been reading the newsletter survey filled out at Spring DECUS. (Being Newsletter Chair occasionally has its advantages, I get first read on goodies, etc.) We had 3 out of 60 readers say they read the DeVIAS letter. A very unscientific extrapolation of this data suggests that 5% of readers still are involved with IAS. That translates out to about 325 newsletter readers. If we really force things and count the three readers who checked off IAS in the "weakest" section, (but did not check of IAS in the sections read question,) that ups our percentage to 10% of the readers who notice the DeVIAS letter.

We also garnered one "Top section" vote, but also got THREE "Weakest section" votes. The three people who rated us weakest, however, did not list the DeVIAS letter in as their "most interested" sections. They also checked off as weaknesses "Poor content" and "Too small an issue" Ok IAS hackers and users, you can do something about both those issues. Send in some good articles, and make this newsletter "Bigger and Better." (I could probably pursuade myself to let me go over page count for an issue or two!)

IN THIS ISSUE

Return with us now to the thrilling days of yesteryear. You IAS oldsters can flex your memory and you newer members can see what it was like to be an early IAS (and RSX11D) user. Take the "Real RSX Programmers Quiz".

Our program of the month is an update to an old program. It us used to fix a file clobbered when your long running program (or the system) crashes with a 2000 block output file with 0 blocks used.

CONTRIBUTION GUIDLINES

Contributions of articles, SPR's, letters, etc. will be accepted in any form, (including notes jotted on gravy-stained tablecloths.) They will be more happily accepted in one of the following formats:

Paper submissions will always be accepted. Publishing may be delayed until the editor gets some time at the keyboard to convert them to our current format. We can accept submissions by FAX. Call for info.

Contributions may be submitted on tape, (800,1600, 3200 or 6250 BPI,) DEC-tape II, and DecMate or RT11 floppies. We're not fussy, we'll even accept paper tape or cards. We can read any IAS/RSX, RT11, VMS format. Any media sent to us will be promptly returned.

We have 2400/1200 baud modems on our IAS system and our VAX, with KERMIT for electronic submission. Give the editor a call @ (312)-791-8075 (preferably later in the day,) to obtain access info, etc. You can also submit over DCS, by sending mail to BORGER.

If you have a problem you would like to submit to the Devias Demon, send it to the Editor at the following address. Answers to problems from members (or anyone) should also be sent to the Editor at:

Frank R. Borger
Michael Reese Medical Center
Department of Radiation Therapy
Lake Shore Drive at 31st St
Chicago, IL 60616

TEN YEARS AGO THIS MONTH

The August and September issues of the Multi-Tasker contained:

An SPR submitted by "Kenneth Olson" of TMI Systems Corp. (Guess DEC's president still moonlighted in the old days.) Seriously, that user must have taken a lot of ribbing if he ever attended DECUS.

A real system hacker article described how to add a new system directive to RSX11M. Unfortunately the authors, (probably in self defense,) were not specified in the article.

The newsletter also contained the first version of the famous R French program to recover a file locked due to a program terminating while it had write access to the file. The program properly set end of file and closed the file, so your 4 hour batch run would not be wasted if the system or the program crashed.

THE REAL RSX11D PROGRAMMER'S QUIZ

Frank R. Borger
Michael Reese Medical Center

Currently RSX and IAS programmers have a plethora of utilities and rich operating systems to make their lives easier. Disk backups are dependable and can be done almost automatically, memory and disk storage are cheap, layered product software is plentiful. After a hard day working with screen based editors, super debuggers, etc. programmers probably relax with Perrier and Tofu.

In the early days of RSX11D (as in Dad, not Mom or Son,) life was not so easy. Hardware and software resources were slim, and when they finally left work, programmers probably resorted to something harder that Perrier.

In recognition of the 7th anniversary of the DeVIAS letter, and the 14th anniversary of the Multi-tasker, this article will present a light hearted retrospective of the early days of RSX11D, looking at the problems early users faced with backups, crashes, etcetera. It is presented in the form of a Quiz, so get your pencils in hand.

Question #1:
Although 32K words was sufficient to run RSX11D as a single user system, DEC recommended more memory to support 2 more terminals. How much more did DEC suggest?

1/ 32K words of memory.
2/ 16K words of memory.
3/ 8K words of memory.

Question #2:
(With appologies to the DECUS commercialism policy,) How much did that additional 8K of memory cost?

1/ $1,700
2/ $3,700
3/ $5,700

Question #3:
When we first received RSX, we got a version 2 tape. How were we saved from having to work with that primitive system, and ended up starting out on version 4?

1/ Our TU10 was down, and we couldn't read the tape.
2/ The tape was bad, and by the time DEC got us a replacement tape, the next release had come out.
3/ The software specialists couldn't get to us.

Question #4:
When the DEC software specialists walked us through our first sysgen and edited [11,17]SYSGEN.CMD, they tried to purge the file but it failed. WHY?

1/ PIP did not yet have the /PU switch.
2/ The file protection was set RWE:RWE:R:R
3/ The file owner was [1,1], and the DEC software specialists were trying to delete it from [11,17].

Question #5:
The first issue of the Multi-Tasker contained a fix to a similar problem. Users reported that they could not delete later versions of RSX.SAV via PIP RSX.SAV;n/DE. What was this first "Bug fix" published in the Multi=-Tasker?

1/ Set UIC to the owning UIC of RSX.SAV
2/ Delete the file right after booting.
3/ Execute PIP via a : "MCR>REQ ...PIP/UIC=[11,17]

Question #6:
What other information about the above "bug" did the newsletter include?

1/ We don't know if an SPR has been sent.
2/ Will be fixed in the next release...
3/ That's not a bug, that's a feature.

Question #7:
The first issue of the newsletter listed some of the new features of the new release of RSX11D. Pick the actual new ones from the following list:

1/ Checkpointing, (swapping)
2/ Support for parity memory.
3/ Support for line printers.

Question #8:
Another new feature was the first disk compression program, (not to be confused with DSC.) What was the major problem with this utility?

1/ It was very slow.
2/ It did not move tasks.
3/ it could not handle large disks.

Question #9:
What announcements appeared in the first "Notes from the RSX Underground"?

1/ A fast FORTRAN compiler that generated inline code, and didn't have to call MACRO to finish the job.

2/ A PDP11 writeable control store.
3/ RSX11M.

Question #10:
About this time, a DEC CPU handbook listed the software operating systems available for the PDP11. How many of the following were available in 1975?

1/ PTS-11 Paper Tape System.
2/ CAPS-11 Cassette Programming System. (Audio cassettes)
3/ RT-11
4/ RSTS/E
5/ RSX-11M
6/ RSX-11D
7/ IAS

Question #11:
Which backup method did real RSX programmers use?

1/ DSC
2/ Preserve
3/ Shut down your system and run diags.

Question #12:
When Larry decided that shutting down the system to do backups was wasteful, he dumped part of a rollin tape and deduced its format, (a short header followed by disk blocks, blocked 12 blocks to a record.) He then wrote a version of ROLLIN that ran online under RSX. What langage did he use?

1/ FORTRAN
2/ BASIC
3/ MACRO

Question #13:
When a REAL RSX11(D) programmer writes their first major program, what is it?

1/ A large data base.
2/ A Driver for an electrostatic lineprinter/plotter.
3/ A Graphic storage scope driver for the then unsupported (and still poorly supported,) Lab Peripheral System

Question #14:
What was the major stumbling block for the above task?

1/ Until the handler was working, you only had a 30cps, 80 column decwriter for listings.
2/ DEC gave no examples of DMA output.
3/ DEC did not provide any sources for any handlers, at all, except the TT handler, (which was a monstrosity.)

Question #15:
What was the first known program to overwrite the system disk boot block with an ascii error message?

1/ The Card Reader Handler
2/ The Message Output handler
3/ Crash Dump Analyzer

Question #16:
What did Cy Mead have to do to recover from the above ?

1/ Use another disk to do a BOO/WB
2/ Do a full sysgen from DECTAPE and Cards.
3/ Continue using a backup copy of the system disk.

Question #17: Under version 4 RSX, what was a major limitation to MCR?

1/ It was a single-user task.
2/ There were no indirect mcr facilities.
3/ It had rudimentary error facilities.

Question #18:
Under version 4 of RSX, what was wrong with PIP and EDI.

1/ They tended to hog the system.
2/ They were single user.
3/ They tended to crash the system.

Question #19:
When we created multi-user versions of popular programs, why did we have to create copies of the tasks, instead of just saying "INS PIP/TASK=...POP"?

1/ Install did not have the "/TASK=......" switch.
2/ The system checked the disk indicator and the starting block number, and wouldn't let you install a task twice.
3/ There was no [0,0]SWPFIL001.SWP

Question #20:
On an early RK05 disk drive, what was the lamp next to the "ONCYL" lamp on the right?

1/ WT PROT
2/ RDY
3/ GRETEL

Question #21:
Under version 4 RSX, what type of CPU allocation scheme could you use.

1/ Round robin scheduling.
2/ Heuristic scheduling.
3/ Random scheduling.

Now for the answers. Note that some of the questions can be answered as "all of the above". Give yourself one point for each correct answer.

Answer #1: 3

DEC suggested another 8K of memory.

Answer #2: 3

Yes, $5,700 for 8K words of memory.

Answer #3: 2

The tape was bad, and it took them 8 weeks to come up with a replacement.

Answer #4: 3

Although PIP did not yet have a purge switch, the DEC specialists were trying to use the "/DE" switch. They didn't know about the difference between File owner UIC and file directory UIC.

Answer #5: 3

Users used to single directory DOS, still had no idea about the dual UIC nature of Files-11.

Answer #6: 1

Yes, they did consider SPR'ing the "bug."

Answer #7: All of the above.

With this release, you could checkpoint tasks, detect memory problems, and use your line printer.

Answer #8: 2

Although it was slow, (2 hours or so to compress a 2.5M-byte RK05, near forever for an RP03,) its main bug was that it would not move any tasks, so your disk still ended up fragmented by all the task images.

Answer #9: All of the above.

And yes, as usual, DEC declined all knowledge of RSX11M.

Answer #10: All of the above.

I believe this was the widest spread of operating systems offered for the PDP11, paper-tape all the way up to IAS.

Answer #11: 3

DSC was DEC's first reasonable backup. (PREserve belonged in one.) With version 4, there was no method to do backups. But, if you shut down your system and ran MAINDEC diagnostics, you could use ROLLIN, (although it was totally undocumented,) to backup RK's and RP's.

Answer #12: 1

None of us were proficient in Macro yet, and BASIC / RSX11D was only a dream. Since we were old FORTRAN hacks...

Answer #13: 2

The first thing I had to write was a driver for an electro-static printer plotter. (The second thing I had to write was a driver for the storage scope.)

Answer #14:

All of the above. Since I had practiced writing a driver for the above mentioned printer/plotter for DOS, we transferred the listing to a DOS disk to do listings.

Answer #15: 1

We have had the MO handler and CDA overwrite the boot block, but CY Mead, (founder of CARTS, the Chicago Area Real Time Society,) was the first to have this happen.

Answer #16: 2

Cy only had one RK and Dectape, so he had no method to do disk backups, or to rewrite the bootblock. He had to do a full sysgen from Dectape, plus re-compile and link all his programs from command decks thru the card reader.

Answer #17: All of the above.

1/ As long as MCR was prompting at one terminal, no one else could use it. (An early program in the SIG newsletter contained a program to let everyone use MCR.)

2/ There were no indirect MCR facilities. (We ended up writing a rudimentary one.)

3/ One of our early SPR's complained that "F012" was a error message for the actual problem of "Task too large to load."

Answer #18: 2

Under version 4, there was no such thing as a multi-user task. Our work-around was to create copies of the task image, and install them with new names, (EDI, EDO, and EDU, PIP, POP, and PUP etc.)

Answer #19: 3

There was no swap file, all tasks swapped back to their original disk space. Obviously you could not have two versions of the same task swap to the same disk area.

Answer #20: 3

A group we new relabled the fault light so that they had "ONCYL" and "GRETEL"

Answer #21: None of the above.
If you were the first person to request a task executing at any given priority, subsequent users with tasks at the same priority were always behind you in the ATL. (We ended up writing a very simple program, which just checked for tasks with the same priority, and pulled the first one out of the ATL and put it back in again. The system, when it reinserted the task, put it at the end of tasks of the same priority.)

Scoring:

0-7	Go back to using your IBM PC.
8-14	Welcome to the club.
15-21	You're learning, (the hard way.)
21-28	An old time RSX user.
over 28	Want to take over my job?

THE PROGRAM OF THE MONTH CLUB

Our ten years ago today column mentioned the first listing of the Recover program. Here is the lastest version. It does more than unlocking a locked (record based) file. It unlocks the file, and proceeds to read records until it reaches the last readable record of the file. (Some garbage may occur near the end.) It then sets the file header accordingly so that the file can be read normally.

This version has been updated to function correctly with multi-header files, and to let one also set the last record number of a fixed length file. Unfortunately it currently does not support operation on ODS-2 files under RSX11 mode on VAXen.

```
           .TITLE     RECOVER
           .IDENT     /MRH001/
           .SBTTL     INTRO PAGE
;
; MCR TASK TO RECOVER A FILE LOCKED AS A RESULT OF TASK ABORTION
;
; ORIGINAL AUTHOR     R B FRENCH NOV 76     THE BOEING CO
;
; MODIFICATIONS                  F BORGER   MAR 77     MINOR CHANGES
;                                F BORGER   2/81       ACCOUNT FOR MULT. HEADERS
;                                F BORGER   ?          ADD SWITCHES
;
;CALLING SEQUENCE
;
;MCR>REC[OVER] FILENAME[/SWITCH]
;
;ALLOWABLE SWITCHES : /LA:NNN    /AD:NNN
;                     LA = SET LA(ST) RECORD # TO NNN
;                     AD = SET LAST RECORD TO # IN FILE + AD  (AD CAN BE NEGATIVE)
;                     (THE ABOVE SWITCHES ONLY WORK FOR FIXED LENGTH RECORDS)
;                     AD SWITCH INTRODUCED CAUSE LA AND AD CAN ONLY BE 16 BIT VALUES
;
;RECOVER WILL UNLOCK THE FILE (IF LOCKED) UNLESS ANOTHER TASK HAS WRITE
;ACCESS. IT WILL THEN SCAN THE FILE AND REWRITE FILE ATTRIBUTES TO
;REFLECT THE ACTUAL SIZE OF THE FILE (RATHER THAN 0)
;
;TASK BUILD PROCEEDURE
;
;RECOVER/MU=RECOVER,LB:[1,1]EXEC.STB/SS
;/
;TASK=...RCV
;UNITS=2
;ASG=TI:2
;LIBR=SYSRES:RO
;//
;
           .MCALL     GCMLB$,GCML$,CSI$,CSI$1,CSI$2,DIR$,QIOW$
           .MCALL     FINIT$,FSRSZ$,FDOP$A,OPEN$U,CLOSE$,EXIT$S
           .MCALL     FDRC$A,FHDOF$,GET$,CSI$SW,CSI$SV,CSI$ND
;
           FHDOF$     DEF$L                 ;DEFINE FILE HEADER OFFSETS
AC0=%0                                      ;DEFINE FLOATING POINT REGS USED
AC1=%1
AC2=%2

    .SBTTL INTRO CODE
RECOV:     FINIT$                ;INITIALIZE FSR
;
;READ AND DECODE COMMAND LINE
;
START:     GCML$      #GCLBLK               ;GET A COMMAND LINE
           BCC        10$
           CMPB       #GE.EOF,GCLBLK+G.ERR ;IF ERROR WASN'T END OF FILE
           BNE        SERR                  ;THEN IT'S A SYNTAX ERROR
           EXIT$S                           ;IF END OF FILE NORMAL EXIT
10$:       CSI$1      #CSIBLK,GCLBLK+G.CMLD+2,GCLBLK+G.CMLD ;CHECK SYNTAX
           BCS        SERR                  ;THIS IS ALSO A SYNTAX ERROR
           CSI$2      #CSIBLK,OUTPUT,#SWITCH           ;TRY TO DECODE ONE FILE NAME
           BCS        SERR                  ;THIS TOO IS A SYNTAX ERROR
;HAVE A GOOD FILE SPEC
           MOV        #FDB,R0               ;FBD ADDRESS -> R0
           MOV        #FDB+F.FNB,R1         ;FNB ADDRESS -> R1
           MOV        #CSIBLK+C.DSDS,R2 ;DATA SET DESCRIPTOR BLOCK -> R2
           CLR        R3                    ;ALL SET TO PARSE THE FDB
           JSR        PC,.PARSE  ;SO DO IT
           BCS        FERR                  ;BR IF FILE ERROR
           JSR        PC,.FIND   ;NOW LOOK FOR THE FILE
           BCC        READHD                ;BR IF FILE FOUND
           CMPB       #IE.NSF,FDB+F.ERR ;WAS ERROR NO SUCH FILE ?
           BNE        FERR                  ;IF NOT SAY FILE ERROR
           JMP        NOFILE                ;IF SO TELL THE GUY (PERSON THAT IS)
SERR:      JMP        SYNERR                ;SOME BRANCH AIDES
FERR:      JMP        FCSERR
USEERR:    JMP        FILUSE
EMPERR:    JMP        FILEMP
;
;READ FILE HEADER, UNLOCK IF NECESSARY (BUT CHECK ACCESS COUNT FIRST)
;
```

```
READHD:    DIR$       #READ                 ;READ FILE HEADER
           BCS        FERR                  ;ANOTHER FCS ERROR
           BITB       #100,HEADER+H.UCHA ;IS FILE LOCKED ?
           BEQ        COUNT                 ;DON'T HAVE TO DO NEXT IF IT ISN7T
           MOV        #RDSTAT,READ+Q.IOPL+2 ;SET UP TO READ STATISTICS BLOCK
           DIR$       #READ                 ;DO IT
           BCS        FERR                  ;LOTS OF FCS ERRORS CAN HAPPEN
           TST        STAT+10               ;IS FILE CURRENTLY IN USE ?
           BNE        USEERR                ;IF IT IS TELL THE OPERATOR
           BIC        #100,HEADER+H.UCHA ;OK TO CLEAR LOCKED BIT NOW
           DIR$       #WRITE                ;WRITE CHARACTERISTICS BLOCK BACK
           BCS        FERR                  ;CHECK THAT THAT WAS OK

    .SBTTL SCAN HEADER MAP AREA TO RECONSTRUCT ACUTAL SIZE
;
COUNT:     CLR        R4                    ;CLEAR BLOCK COUNTER
           MOV        FDB+F.FNB+N.FID,FIDSAV          ;SAVE FIRST HEADER FID
           MOV        FDB+F.FNB+N.FID+2,FIDSAV+2
COUNT1:    MOVB       HEADER+H.MPOF,R1 ;MAP AREA OFFSET -> R1
           ASL        R1                    ;MAKE IT BYTE (NOT WORD) OFFSET
           ADD        #HEADER,R1 ;R1 NOW POINTING AT MAP AREA
           MOV        R1,R2                 ;AND R2
           MOVB       M.USE(R2),R2          ;NUMBER OF POINTERS -> R2
           BIC        #177400,R2 ;CAUSE ITS A BYTE
           BEQ        EMPERR                ;FILE EMPTY IF ZERO
           ASR        R2                    ;MAKE R2 WORD OFFSET
           ADD        #M.RTRV+1,R1          ;START OF POINTERS IN R1
10$:       MOVB       (R1),R3               ;BLOCK COUNT-1 -> R3
           BIC        #177400,R3 ;CAUSE ITS A BYTE
           INC        R3                    ;NOW HAVE BLOCK COUNT
           ADD        R3,R4                 ;ADD IT TO COUNTER
           ADD        #4,R1                 ;BUMP TO POINTER
           SOB        R2,10$                ;LOOP TIL DONE
           MOVB       HEADER+H.MPOF,R1 ;MAP R1 TO MAP AREA AGAIN
           ASL        R1
           ADD        #HEADER,R1
           MOV        M.EFNU(R1),R2         ;CHECK FOR EXTENSION FILE ID
           BEQ        COUNTX                ;IF NONE, WE HAVE ALL THE BLOCKS
           MOV        R2,FDB+F.FNB+N.FID ;ELSE CHANGE THE FILE ID
           MOV        M.EFSQ(R1),FDB+F.FNB+N.FID+2
           DIR$       #READ                 ;AND READ THE EXTENSION FILE HEADER
           BCC        COUNT1                ;COUNT SOME MORE IF READ WAS OK
           JMP        FCSERR                ;ELSE REPORT ERROR
;
COUNTX:    CMP        FIDSAV,FDB+F.FNB+N.FID ;DID WE LOOK AT EXTINSION FILE HEADER ?
           BEQ        OPEN                  ;BRANCH IF NOT
           MOV        FIDSAV,FDB+F.FNB+N.FID
           MOV        FIDSAV+2,FDB+F.FNB+N.FID+2
           DIR$       #READ                 ;READ ORIGINAL FILE HEADER AGAIN
           BCC        OPEN                  ;BR IF OK
           JMP        FCSERR                ;ELSE REPORT ERROR

    .SBTTL OPEN THE FILE AND PUT ACTUAL ATTRIBUTES VALUES IN FDB
;
OPEN:      OPEN$U     #FDB                  ;OPEN THE FILE (FOR UPDATE)
           BCS        FERR                  ;ANOTHER FCS ERROR
           CLR        FDB+F.HIBK
           MOV        R4,FDB+F.HIBK+2 ;UPDATE BLOCK COUNT
           CLR        FDB+F.EFBK
           MOV        R4,FDB+F.EFBK+2 ;AND LAST BLOCK (SINGLE PREC SHOULD BE OK)
           BITB       #R.FIX,FDB+F.RTYP ;ARE RECORDS FIXED LENGTH ?
           BEQ        5$                    ;IF NOT, SKIP NEXT
           JSR        PC,CALEFB  ;CALCULATE END OF FILE BYTE
           BR         10$                   ;AND DON'T DEFAULT END OF FILE BYTE TO 1000
5$:        TST        RECFLA                ;DID OPERATOR SPEC # OF RECORDS TO SET
           BEQ        6$                    ;IF NOT WE'RE OK
           MOV        #NFXMSG,QIOW+Q.IOPL ;ELSE TELL THEM WE CAN'T DO IT FOR
           MOV        #NFXLEN,QIOW+Q.IOPL+2 ;VARIABLE LENGTH RECORDS
           DIR$       #QIOW                 ;BUT DO THE REST
6$:        TST        FDB+F.FFBY ;IS END OF FILE BYTE DEFINED ?
           BNE        10$                   ;BR IF IT IS
           MOV        #1000,FDB+F.FFBY ;ELSE DEFAULT IT
10$:       TST        FDB+F.RSIZ ;IS RECORD SIZE DEFINED ?
           BNE        11$                   ;SKIP IF SO
           MOV        #1000,FDB+F.RSIZ ;ELSE DEFAULT IT FOR NOW
;
;NOW CLOSE AND REOPEN FILE SO ACP KNOWS ABOUT NEW SIZE
;
```

```
11$:       MOV        FDB+F.FNB,-(SP)       ;SAVE FILE ID
           CLOSE$     #FDB
           MOV        (SP)+,FDB+F.FNB       ;RESTORE FILE ID
           BCC        111$                  ;BR IF OK
           JMP        FCSERR
111$:      OPEN$U     #FDB
           BCC        112$                  ;BR IF OK
           JMP        FCSERR
112$:      BITB       #FD.RWM,FDB+F.RACC ;IS IT BLOCK MODE ACCESS ?
           BNE        CLOSE                 ;IF SO JUST CLOSE FILE
           BITB       #R.FIX,FDB+F.RTYP ;FIXED LENGTH RECORDS ?
           BNE        CLOSE                 ;BR IF THEY ARE

    .SBTTL FOR NON-FIXED RECORDS, SCAN TO DETERMINE MAX RECORD SIZE
;
           CLR        R1                    ;CLEAR TEMP ACCUM
SCAN:      GET$                             ;MAIN LOOP GET RECORD
           BCC        RECCHK                ;IF NO ERROR CHECK MAX REC LENGTH
           CMPB       #IE.EOF,FDB+F.ERR ;END OF FILE ?
           BEQ        40$                   ;IF SO, WRAP IT UP
           MOV        FDB+F.NRBD+2,R2 ;OKAY GET POINTER TO LAST RECORD
           CMP        R2,FDB+F.BDB          ;IS IT IN BLOCK BUFFER OR USER BUFFER ?
           BLT        10$
           SUB        FDB+F.BDB,R2          ;SUB START OF BLOCK BUFFER
           SUB        #S.BFHD,R2 ;ACCOUNT FOR BUFFER HEADER
           BR         20$
10$:       SUB        #HEADER,R2 ;JUST ACCOUNT FOR HEADER
20$:       SUB        #2,R2                 ;OFFSET FOR COUNTER
           MOV        R2,FDB+F.FFBY         ;START OF BAD RECORD IS FIRST FREE BYTE
           MOV        FDB+F.VBN,FDB+F.EFBK  ;CHANGE END OF FILE BLOCK
           MOV        FDB+F.VBN+2,FDB+F.EFBK+2 ;
40$:       MOV        R1,FDB+F.RSIZ         ;NEW RECORD SIZE IN FDB IF NECESSARY
;PUT CORRECT ATTRIBUTE VALUES IN FDB
CLOSE:     CLOSE$     #FDB                  ;CLOSE THE FILE
           JMP        START                 ;AND TRY FOR MORE COMMANDS
;SMALL LOOP TO KEEP TRACK OF MAX RECORD SIZE
RECCHK:    CMP        R1,FDB+F.NRBD         ;WAS ANY PREVIOUS RECORD LARGER OR SAME ?
           BGE        SCAN                  ;IF SO JUST KEEP LOOKING
           MOV        FDB+F.NRBD,R1         ;ELSE REMEMBER NEW SIZE
           BR         SCAN                  ;AND TRY AGAIN

    .SBTTL SUB TO CALCULATE END OF FILE BYTE NUMBER FOR FIXED LENGTH RECORDS
;          MODIFIED FOR DOUBLE PRECISION CALC SO CAN HAVE >32767 BYTES
;
CALEFB:    MOV        R4,-(SP)   ;SAVE USED REGS
           TST        RECFLA                ;OPERATOR SPECIFING NUMBER OF RECORDS ?
           BEQ        1$                    ;IF NOT DEFAULT TO MAX NUMBER FILE CAN HOLD
           MOV        RECNUM,SR0+2          ;NUMBER OF RECORDS TO SR0
           CLR        SR0
           CLR        RECFLA                ;IN CASE WERE DOING MULTIPLE COMMANDS
           BR         2$                    ;AND JOIN SEQUENCE LATER
1$:                                         ;CALC # OF RECORDS = MAX THAT WILL FIT
           CLR        SR0                   ;ASSUME NO FILE > 32767 BLOCKS
           MOV        FDB+F.EFBK+2,SR0+2
           SUB        #1,SR0+2   ;NUMBER OF FULL BLOCKS IN SR0,SR0+2
           MOV        #1000,SR1+2           ;BYTES/BLOCK -> SR1
           CLR        SR1
           JSR        PC,MUL                ;GET # OF BYTES IN SR0
           SETI                             ;CHANGE TO INTEGER MODE
           LDCID      FDB+F.FFBY,AC1        ;CONVERT # OF BYTES IN LAST BLOCK
           ADDD       AC1,AC0               ;ADD THOSE TO # OF RECORDS
           SETL                             ;BACK TO LONG MODE
           STCDL      AC0,SR0               ;AND GET THEM BACK
           CLR        SR1                   ;GET RECORD SIZE IN SR1
           MOV        FDB+F.RSIZ,SR1+2
           JSR        PC,DIV                ;CALC # OF RECORDS
2$:        TST        ADJFLA                ;ADJUST # OF RECORDS ?
           BEQ        22$                   ;BR IF NOT
           CLR        ADJFLA                ;CLEAR FLAG FOR POSSIBLE NEXT TIME
           SETD                             ;SET MODES IN CASE WE HAVEN'T CALLED
           SETL                             ;MUL OR DIV YET
           LDCLD      SR0,AC0               ;# OF RECORDS -> AC0
           SETI                             ;BACK TO INTEGER MODE
           LDCID      ADJNUM,AC1 ;# ADJUSTMENT -> AC1
           SETL                             ;BACK TO LONG MODE
           ADDD       AC1,AC0               ;ADJUST # OF RECORDS
           STCDL      AC0,SR0               ;AND GET ANSWER BACK
22$:       CLR        SR1                   ;GET RECORD SIZE AGAIN
```

```
           MOV        FDB+F.RSIZ,SR1+2
           JSR        PC,MUL                ;SR0 = TOTAL BYTES
           MOV        #1000,SR1+2           ;BLOCK SIZE -> SR1
           CLR        SR1
           JSR        PC,DIV                ;QUOT (SR0) = # OF FULL BLOCKS
                                            ;REMAIN (SR1) = FIRST FREE BYTE IN NEXT BLOCK
           MOV        FDB+F.HIBK+2,R4       ;CALC MAXIMUM ENDFILE BLOCK
           TST        SR1+2                 ;DID LAST RECORD JUST FIT BLOCK ?
           BNE        23$                   ;BR IF NOT
           ADD        #1,R4                 ;ENDFILE BLOCK CAN BE 1 GREATER THAN LAST
23$:       ADD        #1,SR0+2   ;ADD 1 TO BE FIRST FREE BLOCK
           ADC        SR0
           CMP        SR0+2,R4   ;IS CALCULATED ENDFILE BLOCK IN THE FILE
           BLOS       3$                    ;BR IF OK
           DIR$       #BADREC               ;REPORT BAD RECORD #
           BR         4$                    ;AND DON'T REDO RECORD NUMBER INFO
3$:        MOV        SR1+2,FDB+F.FFBY ;WRITE 1ST FREE BYTE TO FDB
           MOV        SR0,FDB+F.EFBK        ;WRITE THE ENDFILE BLOCK BACK TOO
           MOV        SR0+2,FDB+F.EFBK+2
4$:        MOV        (SP)+,R4
           RTS        PC

    .SBTTL 32-BIT MULTIPLY & DIVIDE ROUTINES
;
;ROUTINES TO PERFORM MULTIPLY & DIVIDE USING FPP CAUSE OTHERWISE
; CALEFB ROUTINE BLOWS UP IF TOTAL # OF BYTEST > 32767
;
;MULTIPLY SR0 BY SR1, LEAVE RESULT IN SR0
;
MUL:       SETL
           SETD                             ;SET CALC MODE
           LDCLD      SR0,AC0               ;LOAD VALUES
           LDCLD      SR1,AC1
           MULD       AC1,AC0               ;GET ANSWER
           STCDL      AC0,SR0               ;PUT IT BACK
           RTS        PC
;
;DIVIDE SR0 BY SR1, LEAVE RESULT IN SR0, REMAINDER IN SR1
;
DIV:       SETL                             ;SET CALC MODE
           SETD
           LDCLD      SR0,AC0               ;LOAD VALUES
           LDCLD      SR1,AC1
           LDD        AC0,AC2               ;SAVE DIVIDEND IN AC2
           DIVD       AC1,AC0               ;RESULT IS IN ACO
           STCDL      AC0,SR0               ;CONVERT ANSWER TO INTEGER
           LDCLD      SR0,AC0               ;GET BACK INTEGER PART
           MULD       AC1,AC0               ;MUL BY DIVISOR
           SUBD       AC0,AC2               ;SUB FROM DIVIDEND
           STCDL      AC2,SR1               ;CONVERT REMAINDER TO INTEGER
           RTS        PC                    ;AND WE'RE DONE
;
SR0:       .WORD      0,0                   ;FLOATING POINT SAVE
SR1:       .WORD      0,0

    .SBTTL ERROR ROUTINES
;
NOFILE:    MOV        #NFLMSG,QIOW+Q.IOPL
           MOV        #NFLLEN,QIOW+Q.IOPL+2
           BR         COMERR                ;JOIN MAIN SEQUENCE
;
FILEMP:    MOV        #FLEMSG,QIOW+Q.IOPL
           MOV        #FLELEN,QIOW+Q.IOPL+2
           BR         COMERR                ;JOIN MAIN SEQUENCE
;
FILUSE:    MOV        #FLUMSG,QIOW+Q.IOPL
           MOV        #FLULEN,QIOW+Q.IOPL+2
           BR         COMERR
;
FCSERR:    MOV        #FCSMSG,QIOW+Q.IOPL
           MOV        #FCSLEN,QIOW+Q.IOPL+2
           MOV        #FCSMS1,R0 ;SET TO INSERT FCS ERROR CODE
           MOVB       FDB+F.ERR,R1          ;INTO MESSAGE
           CLR        R2
           JSR        PC,$CBDSG  ;AS A SIGNED DECIMAL NUMBER (NEGATIBE)
           BR         COMERR
;
SYNERR:    MOV        #SYNMSG,QIOW+Q.IOPL
```

```
           MOV        #SYNLEN,QIOW+Q.IOPL+2
           BR         COMERR
;
COMERR:    DIR$       #QIOW
           JMP        START                 ;AND TRY AGAIN
;
;THE QIO DPB
QIOW:      QIOW$      IO.WLB,2,1,,,,<SYNMSG,SYNLEN,0>

    .SBTTL  ERROR MESSAGES
SYNMSG:    .ASCII     <12>/REC -- SYNTAX ERROR/<15>
           SYNLEN=.-SYNMSG
FCSMSG:    .ASCII     <12>/REC -- FCS ERROR /
FCSMS1:    .ASCII     /    /<15>
           FCSLEN=.-FCSMSG
FLUMSG:    .ASCII     <12>/REC -- FILE IS BEING ACCESSSED/<15>
           FLULEN=.-FLUMSG
FLEMSG:    .ASCII     <12>/REC -- FILE IS EMPTY/<15>
           FLELEN=.-FLEMSG
NFLMSG:    .ASCII     <12>/REC -- NO SUCH FILE/<15>
           NFLLEN=.-NFLMSG
NFXMSG:    .ASCII     <12>/REC -- CAN'T SET # OF RECORDS, NOT FIXED LENGTH/<15>
           NFXLEN=.-NFXMSG
           .EVEN
;
BADREC:    QIOW$      IO.WVB,2,1,,,,<BADMSG,BADLEN,40>
;
BADMSG:    .ASCII     /REC -- SPECIFIED NUMBER OF RECORDS TOO LARGE FOR FILE/
           BADLEN=.-BADMSG
           .EVEN

    .SBTTL FILE BUFFERS & SUCH
;COMMAND LINE GOODIES
GCLBLK:    GCMLB$     1,REC,HEADER,2
           CSI$
           .EVEN                            ;IS THIS NECESSARY ?
CSIBLK:    .BLKB      C.SIZE                ;CSI BLOCK
                                            ;FILE READ/WRITE AREA
READ:      QIOW$      IO.RAT,1,1,,,,<FDB+F.FNB,RDHDR>  ;READ FILE ATTRIBUTES (OR STAT)
WRITE:     QIOW$      IO.WAT,1,1,,,,<FDB+F.FNB,WRCHAR> ;WRITE FILE CHARACTERISTICS
RDHDR:     .BYTE      -12,0                 ;READ HEADER CONTROL BLOCK
           .WORD      HEADER
           .WORD      0
RDSTAT:    .BYTE      -11,12                ;READ STATISTICS CONTROL BLOCK
           .WORD      STAT
           .WORD      0
WRCHAR:    .BYTE      3,1                   ;WRITE CHARACTERISTICS CONTROL BLOCK
           .WORD      HEADER+H.UCHA
           .WORD      0
HEADER:    .BLKB      1000                  ;BUFFER FOR FILE HEADER
STAT:      .BLKW      5                     ;BUFFER FOR STATISTICS BLOCK
;
;FILE IO MACROS AND BLOCKS
           FSRSZ$     1                     ;MAKE ROOM FOR ONE LUN ONLY
FDB:       FDBDF$                           ;DEFINE FDB
           FDOP$A     1,CSIBLK,C.DSDS
           FDRC$A     FD.PLC,HEADER,1000
;
;COMMAND STRING INTERPERTER SWITCH
;
SWITCH:    CSI$SW     LA,1,RECFLA,,,SWVAL   ;SPECIFY SWITCH ITSELF
           CSI$SW     AD,1,ADJFLA,,,SWVAL1
           CSI$ND
SWVAL:     CSI$SV     DECIMAL,RECNUM,2      ;SPECIFY DECIMAL RECORD NUMBER
SWVAL1:    CSI$SV     DECIMAL,ADJNUM,2      ;SPECIFY DECIMAL ADJUST NUMBER
           CSI$ND
;
FIDSAV:    .WORD      0,0                   ;SAVE 1ST FILE HEADER FID HERE
RECFLA:    .WORD      0                     ;FLAG FOR OPERATER SPEC'D NUMBER OF RECS
RECNUM:    .WORD      0                     ;NUMBER OF RECORDS SPEC'D BY OPERATOR
ADJFLA:    .WORD      0                     ;FLAG FOR OPERATOR SPEC'D ADJ TO REC #
ADJNUM:    .WORD      0                     ;NUMBER TO ADJUST MAX RECORD # BY
           .END       RECOV
```

THE mini-tasker

DECUS
RT-11 SIG NEWSLETTER

RT-11
SIG
DECUS
U.S. CHAPTER

RT

RT-11 MINITASKER
September, 1988

Contents

****

From the Editor:
This month's issue is a little sparse, and the October folder is empty. Please

You'll find a few more bugs (and fixes!) for the RT-11 implementation of DECUS C in this issue. Thanks to James Whitlock for sending those. Stephen Bart sent an article that not only reminds us of the golden days of RT-11 hacking, but provides some useful tidbits about using high memory. John Hamilton's letter is typical of requests I get at least twice a month about reading and writing IBM PC diskettes under RT-11. I have a couple of comments about that on the last page. If anyone has done this successfully, please drop me a note for publication.

All newsletter contributions will be gratefully accepted by:

John M. Crowell
RT-11 Newsletter Editor
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

JOHN F. HAMILTON JR.

OSO SQUARE II APTS.
1515 ENNIS JOSLIN RD. - APT. 128
CORPUS CHRISTI, TEXAS 78412

512-992-4298

07-JULY-88

JOHN M. CROWELL,
RT-11 NEWSLETTER EDITOR,
MULTIWARE INC.
2121-BSECOND ST.,
SUITE 107
DAVIS, CA.

DEAR MR CROWELL:

1- AS A SEMI-RETIRED EXECUTIVE I NECESSARILY HAVE TO OPERATE MY OWN MICRO-11/73 (RUNNING UNDER RT-11) AS THERE ARE NO RT-11 DECUS PEOPLE AROUND HERE. SO I NEED SOME HELP FROM SOME OF YOU SEASONED EXPERTS.

2- FIRST I WOULD APPRECIATE BEING ADVISED AS TO THE PRODUCT(S) YOUR COMPANY SELLS, AS I AM NOT TOO SURE BUT THAT YOU HAVE A PRODUCT THAT ENABLES THE INTER-FACING OF RT-11 AND IBM FORMATTED DISKS/DATA.

3.0- I HAVE THE NEED TO BE ABLE TO MOVE DATA FROM MY RT-11 FORMAT TO IBM (OR COMPATIBLE) EQUIPMENT VIA DISK. I SOLICIT ANY ADVICE AVAILABLE.

IN REVIEWING THE RT-11 MANUALS, I FIND THE FOLLOWING:

> 3.1- UNDER "INITIALIZE/INTERCHANGE"; - A DISK CAN APPARENTLY BE FORMATTED AS AN IBM EBCDIC DISK.
>
> 3.2- UNDER THE VERSION 5.4 UTLITY "FILEX", SEC:7.3.2, FOR TRANSFERRING FILES BETWEEN RT-11 AND INTERCHANGE DISKETTE"; - IT TELLS HOW TO PERFORM SUCH TRANSFERS.

4- WHILE I TOOK A PROGRAMMING COURSE WITH IBM SOME 35 YEARS AGO (THE 2ND PROGRAMMING COURSE EVER TAUGHT IN INDUSTRY BY IBM AND IN MACHINE LANGUAGE ONLY), I AM NOT CERTAIN WHETHER IBM USE THE SAME FORMAT (3471 FOR EBCDIC) FOR THEIR "PC'S" THAT THEY DO ON THEIR LARGER MACHINES.

IF ITEM 3 ABOVE MEANS RT-11 TO IBM CONVERSION IS POSSIBLE AT THE HOME SITE, I WOULD IMAGINE THERE ARE ANY NUMBER OF RT-11 USERS THAT WOULD BE INTERESTED IN KNOWING OF IT.

I WOULD WELCOME ANYTHING YOU MIGHT KNOW ABOUT THIS.

RESPECTFULLY,

John F Hamilton Jr

JOHN F HAMILTON JR

Stephen Bart
Physics Dept.
Bldg. 510A
Brookhaven National Lab
Upton, NY 11973

John M. Crowell
RT-11 Newsletter Editor
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

John,

I recently returned from my first DECUS symposium, in Cincinnati, and was quite impressed by the convention in general and by the **RT-11** sessions in particular. The **RT-11** group seemed especially friendly and I found those sessions more comfortable than those for that other Very Monstrous System. Milton Campbell's remarks at the wrap up session prompted me to write and relate the following technique I have used for many years to free up some precious low memory on our **RT-11** system.

In the Medium Energy Physics group at Brookhaven Lab we do experiments that require our data acquisition system to transfer as much incoming data to mag tape as possible and analyze whatever fraction that we can. The data tapes are later analyzed fully on a VAX 11/750. The acquisition system consists of a PDP 11/45 running RT11FB V5.4D and was originally written around the time of the RT11 V02C to V03 transition on an 11/40. The system utilizes a Macro foreground program to receive the data and transfer it to tape while a predominantly Fortran-77 background program analyzes whatever there is time for. I say the foreground program receives the data because the hardware is actually interfaced to a front end processor which transfers the data into the 11/45 via direct memory access. The front end processor is now a diskless J-11 running **RT-11**. The important point that should be noted here is that the data is received by DMA and is transfered to tape via DMA. Data buffers are typically 2 KW in size and double buffering is used to allow a buffer to be filled while another is transfered to tape. With this buffering scheme, 4 KW is required for buffering, the operating system requires around 5 KW and the foreground code requires 2 KW minimum (to allow the USR to swap over it). This uses a significant fraction of the 28 KW of low memory. The backgound analysis program is fairly large and complex and thus, as most PDP 11 users are, we are somewhat memory constrained. At some point during the **RT-11** V04 days I hit upon the idea of using an interesting quirk of the PDP 11 architecture to move the data buffer out of low memory.

In the early PDP 11 days (does anyone remember DOS?) prior to widespread use of memory management it was quite common to have machines configured with 32 KW of memory of which only the lower 28 KW was accessible. What happened to the other 4 KW? It was there on the UNIBUS but one could not access it without a memory management unit (if the machine had one) because virtual addresses above 160000 were translated into I/O page references. However, an apparently little known fact is that this memory IS accessible to DMA devices without using the extra two bits of addressing found in the CSR of most NPR devices. Thus, this memory could be accessed with the SJ and FB device drivers and I could use this "dead" area for data buffering!

The advent of the memory management unit which could access all eighteen bits of UNIBUS address space and the XM monitor which could program the MMU provided access to this memory (as well as memory with higher addresses). However, this 4 KW block of memory still has the unique property of not being in the normal non-XM address space but yet is accessible to the non-XM (DMA) device drivers.

When I tried to implement the idea, a problem occured because **RT-11** was unhappy I was trying to use a .SPFUN write with a buffer address out of the job's memory space. It took me several hours of looking through the uncommented monitor listings to find the section that checked this address. The routine, which begins at the label CHKSP:: in the monitor source file RMONFB.MAC, checks that an address, which has been pushed onto the stack, is a valid address for the requesting job. The code segment is reproduced below.

```
.SBTTL     CHKSP - CHECK USER ADDRESS FOR BOUNDS AND EVENNESS
.IF EQ     MMG$T
CHKSP::    MOV  R2,-(SP)
           MOV  4(SP),-(SP)
           BEQ  2$
           BIT  #<USRRN$!CSIRN$!BATRN$!CMPLT$>,@CNTXT
           BNE  2$
           BIT  #1,@SP
           BNE  3$
           MOV  JOBNUM,R2
           BNE  1$
           TST  KMONIN
           BNE  2$
      1$:           <----------   CMP  (SP),#160000
           ASL  R2                 BHIS     2$
           ADDR $JBLIM,R2,ADD
           CMP  @SP,(R2)+
           BHIS 3$
           CMP  @SP,@R2
           BLO  3$
      2$:  TST  (R5)+
      3$:  TST  (SP)+
           MOV  (SP)+,R2
           RTS  R5
```

The address which the routine checks is the buffer address from the .SPFUN, .READX or .WRITX request. The routine is typically called with a calling sequence which looks like the following:

```
           MOV  ADDRESS,-(SP)
           JSR  R5,CHKSP
           BR   ERROR
```

If the routine executes the TST instruction at label **2$:** the branch to the error routine is not executed and (assuming there are no other problems) the I/O request will be queued. If the address checked by the routine is in error, the `TST (R5)+` instruction is not executed and the routine returns to branch to the ERROR label. I chose to make my "hack" by adding a test at the label **1$:** which checked for an address of 160000 or higher and allowed such values. This entails some risk since it might allow a job to write random data into the I/O page with a driver for a programmed I/O device. However, I decided the benefits justified the risk. One could also eliminate the test against the job's high limit but this incurs an additional risk and I decided against it. This routine is apparently stable since it has not changed since V04 and the modification has become a routine part of my upgrade procedure; I simply edit in the change to RMONFB.MAC (keeping the original as a backup, of course) prior to running SYSGEN. With this simple modification, I was able to use the memory between 28 KW and 32 KW as the data buffer. A serious deficiency with this technique is that the data buffer cannot be accessed by the CPU without a memory management unit but it works quite well for simply moving data from one DMA device to another. Also, a sufficiently intelligent front end processor can be used to overcome this problem.

A logical question to ask is "Why not use XM and avoid all this trouble?". In our case, there were a couple of reasons. As I mentioned, this system was designed and written at about the same time XM was introduced and in the early days XM was not very reliable. Furthermore, we all ready had written software which programmed the memory management unit directly to satisfy our most important requirement: a large, quickly accessible, storage area for histograms generated by the data analysis. It was a simple matter to slide this area up to begin at 32 KW and free up the space from 28-32 KW. I have recently considered converting the system to XM but I was a bit troubled by a remark I heard in Cincinnati which asserted the XM PLAS directives were "slow". This would have serious implications for us and would need to be studied before implementing such a conversion. I also have to consider that the PDP 11/45 will be phased out over the next few years to be replaced by a VAXSTATION 3200. Fortunately, however, **RT-11** will live on in our J-11.

Regards,
Stephen Bart

TO: John M. Crowell
RT-11 Newsletter Editor
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

FROM: James D. Whitlock
Woodcraft Components
7111 Forest Pine
San Antonio, TX 78240
(512) 681-0439

FOR THE DECUS C FLIT GUN

Case of the missing edit.

The CLIB FSEEK function wrecks a lot of programs under RT-11 (not RSX) because the true edit 11 fix got left out of the DECUS C V2 distribution. Prior distributions have it. No doubt, author was not using the most recent version of fseek when making other corrections.

The following code fragments correct this problem.

```
        .title  fseek   Reposition file pointer (seek)
.iif    ndf     rsx     rsx     =       1      ;Assume rsx11m
.iif    ndf     rmsio   rmsio   =       0      ;Assume FCS-11
.if ne rmsio
.ident  /RMS014/                        ;!!JDW fix up ident
.iff
.ident  /000014/                        ;!!JDW fix up ident
.endc
 ...
 ...    <Fix up edit history>
 ...
; 000010 08-Jun-83 RBD  Don't clear V$BCNT (on RSX) if file open
;                       in "u" mode. Forces a spurious read.
; RMS011 01-Jan-84 RBD  Add conditional support for RMS-11(V2) on
;                       RSX and (particularly) P/OS.  It's a hack
;                       ... see 'bugs'.
; RMS012 20-Jan-84 RBD  Add call to RMS $REWIND if file has banged
;                       into EOF.
; RMS013 07-Feb-84 RBD  Well ... I've finally learned to use RFA
;                       access on RMS-11.  I thought it worked
;                       only on relative and indexed files ...
;                       No more IRAB/IFAB hacks.
; 000014 24-Sep-86 JDW  OLD 000011 13-Oct-83 MM
;                       Clear r0 on exit GOT LOST, SO PUT BACK IN.
;                       IF NOT IN, COOKIE AND COOBLD AMONG OTHERS
;                       ARE WRECKED UNDER RT11. ;!!JDW
;
; Normal return
;
 ...
 ...    <Reinsert the missing edit 11>
 ...
        bic     #vf$eor,v$flag(r4) ;Clear error and eof flags    ;07
;!!JDW The valuable edit 11 got skipped
        CLR     R0                 ;Clear r0 on exit      ;!!JDW  ;14
40$:    jmp     cret$              ;back we go.
        .end
```

TO: John M. Crowell
RT-11 Newsletter Editor
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

FROM: James D. Whitlock
Woodcraft Components
7111 Forest Pine
San Antonio, TX 78240
(512) 681-0439

FOR THE DECUS C FLIT GUN

DECUS C does not consider a lonely formfeed or vertical tab as a record. Often, files end with a formfeed. If these files are archived for instance, the formfeed is lost. Formfeeds followed by a legitimate record or cr-lf are not lost, because the formfeed rides along as the first character of the next record. In an effort to keep final formfeeds, the CLIB module FGETS.MAC has been altered with no ill effects under RT-11.

The following code fragments give the changes to FGETS.MAC.

```
        .title  fgets   Read a string from a file
;!!JDW  .ident  /000002/
        .ident  /000003/                ;!!JDW Fix up ident
;
 ...
 ...    <Fix up edit history and equates>
 ...
; Edit history
; 000001 10-Mar-80 MM   Conversion for the newer library
; 000002 06-Aug-81 RBD  Terminate long fgets() lines correctly
; 000003 16-Feb-88 JDW  Allow final FF or VT in files to be
;                       gotten ;!!JDW
;
        .psect  c$code
nl      =       12              ;Newline
vt      =       13              ;!!JDW vertical tab             ;03
ff      =       14              ;!!JDW formfeed                 ;03
 ...
 ...    <Modify code>
 ...
20$:    movb    r0,(r3)+        ;Out the byte
        cmp     r0,#vt          ;!!JDW                          ;03
        beq     40$             ;!!JDW                          ;03
        cmp     r0,#ff          ;!!JDW                          ;03
        beq     40$             ;!!JDW                          ;03
        cmp     r0,#nl          ;Line feed?
        bne     10$             ;no, continue
30$:
        movb    (sp),-1(r3)     ;maybe overwrite the newline.   ;02
40$:                            ;here for full lines            ;02
        clrb    (r3)            ;yes, terminate the string
        mov     c$pmtr+0(r5),r0 ;and return buffer
        jmp     cret$
        .end
```

TO:

John M. Crowell
RT-11 Newsletter Editor
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

FROM:

James D. Whitlock
Woodcraft Components
7111 Forest Pine
San Antonio, TX 78240
(512) 681-0439

FOR THE DECUS C FLIT GUN

Here are some observations and experiences concerning the Lempel-Ziv-Welch file compression/decompression utilities and the Fall 1985 RSX SIG tape containing the new DECUS C V2 Language System distribution. This tape is no doubt from whence DECUS C V2, that is found on the Spring 1986 RT SIG tape, came.

File compression is used on some files on the RSX SIG tape so that the entire distribution can fit on a single 2400′ magtape in BRU format. Notably, these are runoff files and archive files. The compression/decompression utilities are included so that the compressed files may be unbundled and restored to their original form. Compressed files are most often identified by the letter 'Z' as the middle character of the file extention. For example, the LZW sources in the archive file LZPDP11.ARC would become LZPDP11.AZC when compressed. (RSX has up to nine characters in the file name).

Here's where the trouble begins. There are about 15 such compressed files on the RSX SIG tape; all of which have unknown file attributes. This means you can copy them, but can't do anything else -- like decompress them.

Assuming the files are ok, the first thing is to make them legit. Since I am not an RSX guru, I used the following brute force method:

1. Using RSX, initialize an RX02 floppy as Files-11 with the MFD, etc. at the end of the diskette. Then contiguously copy the problem file to the RX02 floppy. It helps if the floppy has been formatted with a background of zeros so you can find the end of the file if need be.

2. Next use RT-11 to copy the floppy as part of a device to a legit RT-11 file. The RSX Files-11 file will always start at block 3, and the length will be that as shown in the RSX directory.

For example, the file ASM80.AZC is 28. blocks long and would be handled as follows:

```
COPY/DEV/FILE DY0:/ST:3./EN:30. DK:ASM80.AZC
```

The resulting ASM80.AZC file is a legitimate RT-11 file and can be decompressed under RT-11. For RSX-11M users, the file must now be FLX'ed back into the RSX system in image mode. The file will now have normal RSX-11M file attributes. It can then be decompressed under the RSX-11M environment.

Now for the decompression. About half the files were compressed in binary mode and the rest in ascii mode. All files properly decompressed with no missing data. However, there is more trouble here in the potential sense.

When the LZCOMP compression utility is built using the DECUS C compiler and the required new CPP preprocessor, and run on a PDP-11, the ASCII mode simply does not work under RSX-11M or RT-11. And this mode is very important to RSX-11M. The problem is, that the last buffer of data does not get thru the process. This means that anywhere from 1 byte to 511 bytes will be missing. Since the SIG tape compressed files were all ok; but ascii mode does not work on a PDP-11, I assume the files were compressed with 12 bits on a VAX and the utility was built using VAX-C; and that environment flushes the final buffer automatically.

Since compressed files are occuring on RT-11, RSX-11M, and VAX SIG tapes, it seems to be important that the LZW utilities work completely on those respective operating systems. Also, any files compressed on a 32-bit machine, VAX or otherwise, should be compressed with 12 bits, if targeted as an RSX-11M or RT-11 tape submission. 12 bits is the upper limit for the 16-bit PDP-11. Higher compression renders the file useless.

The following are the code changes to the LZCOMP sources to get ASCII mode to work completely for RT/RSX.

Files affected: LZCMP2.C, LZIO.C

Code fragment for lzcmp2.c

```
...
...     <LZCMP2.C>
...
/*
 * Set hash code range bound.
 */

hshift = 0;
for (hash_code = (long) hsize; hash_code < 65536L; hash_code <<= 1) {
    hshift++;
}
hshift = 8 - hshift;

/*++ Part 1 of 2 that fixes ascii mode final buffer not being processed. ++*/
/*++**********************************************************************++*/

/*++ For ascii mode, this processes final buffer until an all one's byte ++*/
/*++ is recognized.  This seems safe for text mode.  Can't imagine an    ++*/
/*++ all one's byte in normal ascii text.                                ++*/
/*++**********************************************************************++*/

        while ((c = GET(in)) != ((binary) ? (unsigned) EOF : '\377')) {
```

```
/*++********************************************************************++*/

/*++ The above method seems easiest.  Using 'if (binary) {...}' on a    ++*/
/*++ partial while statement doesn't work; and the program would become ++*/
/*++ too big if two complete while statements (one for binary true and  ++*/
/*++ one for binary false) were used, as was done with lzio.c.          ++*/
/*++                                                                    ++*/
/*++ Another method is:                                                 ++*/
/*++                                                                    ++*/
/*++ char       XYZZY;                                                  ++*/
/*++    if (binary) {                                                   ++*/
/*++            XYZZY = (unsigned) EOF;                                 ++*/
/*++    }                                                               ++*/
/*++    else {                                                          ++*/
/*++            XYZZY = '\377';                                         ++*/
/*++    }                                                               ++*/
/*++                                                                    ++*/
/*++    while ((c = GET(in)) != XYZZY) {                                ++*/
/*++                                                                    ++*/
/*++********************************************************************++*/

    in_count++;
    hash_code = (long) (((long) c << maxbits) + code);
    i = (c << hshift) ^ code;           /* XOR hashing          */
    if (HASH(i) == hash_code) {         /* Found at first slot? */
        code = CODE(i);
        continue;
    }
...
...     <Continue with rest of code>
...
Code fragment for lzio.c

...
...     <LZIO.C>
...
# else
        i = fread((char *) s->bstart, 1, s->bsize, stdin);
        if (ferror(stdin)) {
            perror(infilename);
            exit(IO_ERROR);
        }
# endif
#endif
/*++ Part 2 of 2 that fixes ascii mode final buffer not being processed. ++*/
/*++ fread shuts us down too soon.                                        ++*/
/*++**********************************************************************++*/
        if (binary) {
                if (i <=0) {
                    return (EOF);
                }
                else {
                    s->bp = s->bstart;
                    s->bend = &s->bstart[i];
        #if UCHAR
                    return (*s->bp++);
#else
                    return (*s->bp++ & 0xFF);
#endif
                        }
        }       /*++ END binary TRUE ++*/
/*++**********************************************************************++*/
        else {
            s->bp = s->bstart;
            s->bend = &s->bstart[MAXIO];                /*++*/

#if UCHAR
            return (*s->bp++);
#else
            return (*s->bp++ & 0xFF);
#endif

        }       /*++ END binary FALSE ++*/

/*++**********************************************************************++*/
}
        ...
        ...     <Continue with rest of code>
        ...
```

Another problem that isn't as severe. The DECUS C preprocessor CPP can't handle numbers greater than 32767 in comparisons in directives. The first thing we run into is big numbers in LZ.H and LZCMP1.C. Thus the hash table size is computed all wrong for PDP-11 size machines. This doesn't seem to have any affect on the performance of the program tho.

Code fragment for lz.h

```
...
...     <LZ.H>
...
 * Algorithm Tuning parameters:
 * #define USERMEM   <n>        Memory available to compress.
 *                              If large enough, a faster algorithm is used.
 * #define SACREDMEM <n>        Don't use this part of USERMEM.
 * #define BITS      <n>        Maximum number of code bits.
 * #define MAXIO     <n>        Output buffer size (squeeze memory if needed).
 *
 * Edit History:
 *
 * Alterations by:      James D. Whitlock
 *                      Woodcraft Components
 *                      7111 Forest Pine
 *                      San Antonio, TX 78240
 *                      512-681-0439
 *
 * This version is from F85 RSX-SIG Tape.
 *
 * Use decus_cpp to preprocess all sources first.  They contain macros with
 * arguments.  Also continuation lines in "directives" are not supported by
 * decus C.
 *
 * Various changes have been made to correct minor bugs and cosmetics.  They
 * are usually identified by double plus sign tracer comments.
 *
 * 04-27-87
 * PDP11's and decus_cpp handle preprocessor directives as 2's complement
 * signed words.  These sources have been altered to handle comparisons of
 * numbers greater than 32767 in the directives, used as a basis of
 * evaluation, or conditional code inclusion, by associating the big numbers
 * with 32 bitters, and the little numbers with 16 bitters. (brilliant huh?)
 *
 * 05-28-87
 * RT-11 can now pass multiple arguments at the DCL level using the
 * [ arg arg arg ... ] construct.  lzcmp1.c and lzdcm1.c have been altered
 * to handle this..
 *
 * 06-01-87 (biggest bug)
 * In normal text mode, the final input buffer doesn't get processed.  This
 * is now fixed.  Code alterations are in lzcmp2.c and lzio.c.  See these
 * files for further explanation.
 *
 * Keep in mind that under RT-11, text mode strips the CR; and under RSX-11M,
 * text mode strips both the CR and LF.  Often, arch, archc, archx, or RSX-11M
 * FLX will fix this right up, depending on direction.
 */
...
...
...
```

```
/*
 * Set derived tuning parameters.
 */
#ifndef USERMEM
# define USERMEM 0
#endif
#ifndef pdp11   /*++ fixes 2's complement comparisons under decus_cpp ++*/
# if USERMEM >=                 (433484 + SACREDMEM)
#  define PBITS         16
# else
#  if USERMEM >=                (229600 + SACREDMEM)
#   define PBITS        15
#  else
#   if USERMEM >=               (127536 + SACREDMEM)
#    define PBITS       14
#   else
#    if USERMEM >=              ( 73464 + SACREDMEM)
#     define PBITS      13
#    endif
#   endif
#  endif
# endif
#else                           /* Smaller systems                      */
# define PBITS          12
#endif          /*++*/
...
...     <Continue with rest of code>
...
```

Code fragment for lzcmp1.c

```
...
...     <LZCMP1.C>
...
/*
 * This routine is used to tune the hash table size according to
 * the file size.  If the filesize is unknown, fsize should be
 * set to zero.
 */

typedef struct TUNETAB {
    long        fsize;
    code_int    hsize;
} TUNETAB;

static readonly TUNETAB tunetab[] = {
#if HSIZE >= 5003                       /*++ At least this for pdp11 ++*/
    {   1 << 12,         5003   },      /*++ This should cough out   ++*/
#endif
#if HSIZE > 9001
    {   1 << 13,         9001   },
#endif
#if HSIZE > 18013
    {   1 << 14,        18013   },
#endif
/*++ Watch out for signed 2's complement of 35023 on pdp11 and decus_cpp
     as a fall-thru bogey.
     Seems harmless though, if the next conditional gets included.
++*/
#ifndef pdp11           /*++ Fixes 2's complement size on pdp11. ++*/
# if HSIZE > 35023
    {   1 << 15,        35023   },      /*++ This originally coughed out. ++*/
    {     47000,        50021   },
# endif
#endif                  /*++*/
    {         0,            0   },
};
...
...     <Continue with rest of code>
...
```

****

Editor's note:

The problem of reading (and even worse, writing) disks for the IBM PC under RT-11 is one I'm asked about constantly. No doubt many of you are too. The problem is not really in the MS-DOS file structure, which, strange as it is, is not all that difficult to manage. The real problem is in the hardware. The RX50 and RX33 diskettes used in RT-11 have a different format than those used by PC's. If you have a PC disk drive interfaced to your RT-11 system, and a device handler to talk to it, reading and writing PC disks is a snap. (Well, almost.) There are enough handlerholics out there so that even if you don't have the RT-11 device handler, it wouldn't be hard to find someone to write one. If anyone has solved either or both problems - interfacing the drive or writing the handler, write up something about it and send it to the newsletter. If it's a product you sell, send me somehthing anyway. I'll figure out how to let the readers know about it without violating any of the DECUS commercialism rules.

UNISIG

UNI

A Word from the Editor

As you may or may not be aware, there.is a rather long lead-time for the publication of these newsletters. So I write this shortly after the July issue has been delivered. Regular readers remember that July's issue contained an offer of money - actual hard currency, of the United States of America. All you had to do was send me a self-addressed, stamped envelope, along with a note saying you had read the Unisig newsletter. With publication of this issue of the Unisig newsletter, I formally withdraw this offer. Not because I was deluged with SASE's. On the contrary, the offer had a built-in expiration (only the first twenty respondents were to get the money), and an avalanche of responses would have been no problem. But, incredibly, I received **no responses!** This was an offer of **free money.** For the cost of one or two stamps (depending on whether or not you used your company's postage meter to stamp the envelope containing the SASE), at most **50 cents,** you were virtually assured of receiving **one dollar.** That's at least a 100% return on your investment. And it isn't even illegal! I am baffled by the dearth of responses to this offer. Perhaps you felt that by the time I received your SASE, 20 people would have beat you to it and you would have thrown away perfectly good stamps (or stamp). The return on an investment is usually somewhat proportional to the risk involved. But to gamble one or two quarters on twenty strangers getting their acts together and sending in SASE's of their own? Minimal risk, big payoff (percentage-wise, that is). All I can say is, I was rather disappointed.

But I didn't have much time to get depressed, because my system manager finally got us attached to the Usenet (thanks, Lisa!). In case you haven't heard about the Usenet, it is a global network providing (to quote from Kurt Reisler's Net.News abstract in the Fall 87 Session Notes) "vast amounts of technical information (and misinformation), news (and rumors), sources, and bug fixes." This month we have two articles that I pulled from "the net." The first is **Great Moments in UNIX System History** by Andy Tannenbaum. I found it interesting, and useful when playing "Unix Trivial Pursuit." The other article from the net is **Our LeXicon** - an amusing collection of X Windows puns. Not from the net but from our own Jim Livingston, is **So There is Another ULTRIX Shell?** The first in a short series of articles about that "other shell." And we have another installment in the series that tries to answer the question "What does the Steering Committee do?" This month we hear from Bill Cheswick about the role of the Session Notes Editor. Thanks to Bill and Jim for sending things in. And how about you? The offer of the dollar is off, but you can still send me a note or an article. Please send hardcopy to :

Sharon Gates-Fishman
NDC Systems
730 E. Cypress Ave.
Monrovia CA 91016

or e-mail to:

amdahl!cit-vax!ndc!sgf

Great Moments in UNIX System History

1969		Thompson fiddles with filesystem ideas on Multics
		Thompson ports "space travel" from MULTICS to
		GECOS to PDP-7, hacks UNIX filesystem and...
		UNIX V1 written in assembly by Thompson for PDP-7&9
1970		Thompson intends to write FORTRAN using TMG, writes B instead.
		After being denied for $.5M for a research DEC-10, Ossanna
		asks for $65K for a new PDP-11/20 for text processing research.
		Ordered in May, comes in August
		Runs UNIX in core until disk comes in December
		Kernighan suggests name UNIX
1971	Feb	UNIX V2 PDP-11/20 in assembly language on unprotected hardware
		Work begins on C
	Summer	BTL patent dept runs text processing on 11/20 UNIX
	Nov	UNIX V1 manual
1972	Jun	UNIX V2 manual
		UNIX gets pipes
1973	Feb	V3 Manual
	Feb	UNIX V3 PDP-11/34 40 45 60 70 multiprogramming
		C runs on Honeywell 6000, IBM 370
	Oct	UNIX first mentioned in public at ACM 4th SOSP
		Snyder Portable C MIT Masters Thesis
		PWB/UNIX conceived - RJE SCCS PWB/MM
		MERT designed for PDP-11/45
1974	June	V4 Manual
	July	CACM UNIX paper "The UNIX Timesharing System"
		UNIX V5 university release
1975	May	UNIX V6 manual
1976		UNIX V6 commercial release
		LSX LSI-11 UNIX (Minimum Configuration ~$7000)
		Feasibility study for UNIX under VM/370 at Princeton
1977	early	Work begins on Interdata 8/32 UNIX port at Bell Labs
		Interdata 7/32 UNIX port at at U Wollongong Australia
		UNIX V6 commercial sublicenses
		USG UNIX Generic Issue 3 (UNIX V6 based)
	May	PWB/UNIX 1.0 (UNIX V6 based) Bell Release
	Jun	John Lions' commentary on UNIX V6
	late	MERT Release 0
1978	Spring	Johnson and Ritchie Interdata (later PE) port complete
	Jul	Bell System Tech Journal UNIX issue, Part 2 Vol. 57 No. 6
		current: USG UNIX, PWB/UNIX, UNIX V6, Research UNIX
		UNIX TS 1.0 (UNIX V7 based) Bell release
		UNIX 32V VAX port (UNIX V7 based)
1979	Jan	UNIX V7 manual
	Apr	UNIX RT (supersedes MERT)
	Jun	PWB/UNIX 2.0 (UNIX TS based) Bell release
	Dec	3BSD UNIX VAX virtual memory
1980	Feb	UNIX TS 1.3 VAX & PDP-11 Bell release
	Apr	UNIVAC 1100 UNIX TS 2.0+ Bell release
	Jun	UNIX TS 2.0 VAX only Bell release
	Jun	UNIX 3.0 (replaces TS and PWB) Bell release
	Oct	4.0BSD UNIX
1981	March	UNIX/370 Bell Release
	Mar	3B UNIX 3.0 Bell Release
	Jun	4.1BSD UNIX
1982	Jun	UNIX 5.0 Bell release
1983	Aug	4.2BSD manual
	Dec	UNIX System V Release 2 manual

So There is Another ULTRIX Shell?

by Jim Livingston

Most users of ULTRIX very likely use the Berkeley C shell (*csh(1)*) as their command interface, if only because it is the default shell found in the system's basic /etc/passwd -- remember that *csh* is a Berkeley development, and that ULTRIX is a Berkeley UNIX derivative.

In this and the next few newsletters, I'd like to describe another user interface you will find in ULTRIX: the Bourne shell (*sh(1)*), named after its author, Steven Bourne; it predates the C shell a good bit. It has been the shell most often used in AT&T UNIX implementations, at least until the Korn shell (*ksh(1)*) became available. I'll leave *ksh*'s description for another time; suffice it to say that it provides a marked increase in features over *sh*, and is different in "feel" from *csh*.

First, we might note that *sh* is much faster to invoke than *csh*; this is partly because the former is a good deal smaller -- 60K to 100K bytes, depending upon local hacks -- than the latter. Among the reasons for the difference in size (and thus performance) between these shells is the difference in their features. In general, *csh* has more features built into it than *sh* -- a fact often cited derisively by some UNIX purists.

A more radical UNIX zealot of my acquaintance has even been heard to say that *sh* is implemented in the spirit of UNIX, while *csh* exhibits the dreaded "creeping featurism". The "spirit of UNIX", in this context, goes roughly like this: a given command should do just one thing very well, and the purpose of the shell is to make it an efficient operation to connect the output of one command to the input of the next, and thereby to create a "pipeline" of commands to do some larger job.

As additional commands are needed, one builds them up from pipelines of existing commands; one does not rewrite the command interpreter to give it more features, even though, as a user-level process, it's not an especially big deal to incorporate a new shell into your system. Above all, one minimizes the number of commands implemented internally to the shell.

The rationale for this view, of course, is that individual commands can be fine-tuned and specialized as needed, while the shell stays small, clean, and fast.

A second, and major difference between *csh* and *sh*, is that the former has "job control" and the latter does not. This means you can control processes running in the background -- apart from simply killing them -- via signals entered from the *csh* command line. Both shells will start background jobs for you; *sh* will let you kill them, and *csh* will let you do that, plus switch your console from one to another. Both shells will give you information on the processes you have running from your terminal.

As I experience it, job control is mostly superfluous when you have an X-window equipped, UNIX workstation. If you want to start a process and switch from it to another, you just pop up a new *xterm*, and run your new process there.

A third difference between *sh* and *csh* is in the former's command language syntax. The syntax of *csh* is, by design, much like the C language, while *sh* has a syntax reminiscent of ALGOL. I'll give some examples of this syntax later. First, I'd like to show a few sample commands that will give you something of the flavor of *sh*. Most of the commands available from *sh* are the same as those reached via *csh*; commands are generally written to run under either shell.

Readers are most likely familiar with the common directory listing command *ls*. That is, you say (at the *sh* default prompt "$")

```
$ ls
```

and you get something that looks like (in a small directory)

```
addnum    hier
areacode  how.many
$
```

This is just a list of the file names (as in VMS' DIR/BRIEF). It's quick, and the information is limited. Adding options to the command will get variations on that theme. For example,

```
$ ls -l
```

will yield "long" directory entries (more horizontal space than this)

```
-rwxrw-r-- 1 jwl  7 Feb 9 00:27 addnum
-rwxr-xr-x 1 jwl 36 Jan 19 1987 areacode
-rwxr-xr-x 1 jwl 64 May 27 1987 hier
-rwxrw-r-- 1 jwl  5 Feb 9 00:13 how.many
$
```

In future issues, I'll carry on the discussion with *sh*'s environment, and how it's used to help you work more efficiently.

The Unisig Steering Committee

Another in the continuing series that attempts to explain what the members of the Unisig Steering Committee do.

Bill Cheswick - Session Notes Editor

A Session Notes editor collects and edits the slides and viewgraphs used in a DECUS session. The editor's job mostly involves beating and poking speakers to get camera-ready copy to the editor before a deadline. Since the deadline is long before the conference (a month or more), the editor often relies on combatting procrastination. I typically get a third to a half of the sessions covered with session notes.

The notes arrive in a variety of forms: camera-ready, TeX or troff, or even in postscript. The editing process is fairly simple: I throw away empty or useless slides. For example, people often include a viewgraph at the end of their talk containing the word "Questions."

Though the typography varies, it is still helpful to assemble a unified package. Page layout trades readability for expense. How many pages should be displayed per page? I have avoided numbering each page of the notes: I want them ordered by session number, and page numbering has to wait until all submissions have arrived. I chose to put tabs on the edge of each page with the session number on it. Each tab is positioned according to its session number. (Unfortunately, the printers are not able to reproduce the entire page, so some of the tab has been lost in past issues.)

Then the notes are shipped off to the DECUS office for printing. On my first two attempts at session notes, I had them printed myself. This gave me two more weeks to collect articles, which improves the products dramatically. Unfortunately, it is also quite a bit more expensive (DECUS gets a MUCH bigger discount than I) and (as I discovered in Cincinnati) it's against DECUS rules. I am afraid the Anaheim notes will be legal, but much less satisfying to the purchasers.

Our LeXicon

Xtradition -- the way X things have always been done
Xpensive -- CPU-bound Xprograms
Xample -- more Xservers than Xclients
Xized -- being converted to the X cult
Xpending -- while the Xclient is in sleep state
Xstatic -- non-animated Ximages
Xposed -- multi-viewed Ximages
Xsighted -- an Xdisplay first seen from afar
Xtensive -- Xprograms under a deadline
Xacting -- Ximages used for drama
Xpatriot -- Xprograms with a flag-and-country theme
Xseeding -- Xprograms which leave little resident pieces
Xterminate -- kill an X terminal emulation window
Xpress -- quickly iron out wrinkles in an X application

NEWSLETTER OF THE VAX SYSTEMS SIG

Pageswapper

Our Mascot

Table of Contents

Volume 10 Number 2

Contributions

Contributions and suggestions for this newsletter are constantly needed. Articles, letters, technical tips, or anything of interest to our SIG are greatly appreciated. The editor prefers submissions be made electronically. Magnetic tape and hard copy will be accepted, but may delay publication.

Please do not submit program source. It is difficult to typeset and is better distributed on the VAX SIG tape. Please do not submit "slides" from DECUS Symposia presentations or other meetings. They are generally a very incomplete treatment for those readers of the *Pageswapper* who are not so fortunate as to be able to travel to Symposia. Please DO write articles based on such slides.

Send your contributions to:

ARPAnet/CSnet/NSFnet: ctp@cs.utexas.edu
UUCP: ctp@cs.utexas.edu.uucp ({harvard,ihnp4,uunet}!cs.utexas.edu!ctp)
BITNET: CTP@UTADNX
CompuServe: 75226,3135
DECUServe/DCS: POOLE

or if you must, use the U. S. Mails:

Clyde T. Poole
The University of Texas at Austin
Department of Computer Sciences
Taylor Hall 2.124
Austin, Texas 78712-1188

VAX Spring 1988 SIG Tapes Released

Glenn Everhart, VAX SIG Tape Librarian

The S88 VAX SIG tapes have been sent to the initial distribution channels, and LUG librarians should be able to acquire them by contacting appropriate distribution people. The tapes were put together somewhat quickly this year to avoid a delay when my office moves, but have a large and varied software collection. Thanks to all who helped contribute to the collection.

Tapes have been sent to the NLC, the DECUS library, and several tree contacts including the "Other Media" working group chair (Bob Uleski), the RSX regional tree heads (Jim Neeland, Tony Scandora, and Barton Bruce), and several others active in the tree (Dave Schmidt and Tom Gerhard chiefly).

Individuals may obtain the tape via their LUGs or directly from the DECUS library. The complete distribution is one reel at 6250, or four (4) reels at 1600 BPI. The DECUS library packaging has the first three reels labelled as the VAX SIG tape, S88 and the fourth tape labelled as the RSX/PC/VAX collection, S88. This permits each of these packages to fit on one TK50 and simplifies book-keeping there. All NLC and tree contacts have been sent the single-reel 6250 format tape, which is the easiest to deal with.

LUG librarians should contact the nearest NLC regional distributor and arrange to send tape(s) to obtain the SIG tapes. People on the trees should use your old tree lists to contact others higher up. Skip levels if necessary. We will TRY to get a tree list out soon via mail.

Any LUG librarians are asked to send your name, address, and phone number to me to help validate the list. Please do so even if your name exists on the list already. Send to:

Glenn Everhart
VAX SIG Librarian
25 Sleigh Ride Rd.
Glen Mills, PA 19342

The NLC contacts are listed after the end of the SIG tape contents summary.

VAX SIG Spring 1988 SIG tape Contents

The following are the contents of the Spring 1988 VAX SIG Tapes. This year there are three tapes in the VAX collection plus one tape in a mixed VAX/PC/RSX collection which will be described separately.

VAX88A1 Tape

`[.ABEL]` Editing interface built on top of EVE. ABEL is the output of an L&T SIG effort to merge the many EVE supersets of recent years. From Jef Kennedy.

`[.ANUNEWS]` Geoff Huston's Un*x NEWS rewrite for VMS. Allows any VMS site to participate in Usenet NEWS networks and manage newsgroups.

`[.ATTC]` Cluster management tools. From Martin Brunecky.

`[.BRYANT]` TPU enhancements (APLTPU). GBLSUBS – global substitute strings. GBLSEARCH – search for pattern in files.

`[.BZL]` LSE templates for RUNOFF. Spell dictionary with most MACRO-32 instructions defined. From Bart Lederman.

`[.COY]` DM – Directory manager, windowing dir. management. SD – revised Set Default. WPE – WPS-Plus emulator for TPU. COLORS – Reset VTx4x colors. SHOWME – system or user status report. VAXNotes enhancers. Improved MON PROC/TOPC display. From Dale Coy.

`[.CSDHBO]` XSHOW – Show many things across cluster. (Files, holders of IDs, terminals, users, etc. From Jonathan Welch.

`[.DINKIN]` NAME – Find what terminal server and port a terminal is on.

`[.DMILLER]` TPU enhancement. ALOCWAIT – allocate/wait. BBS – full function bulletin board system. CB – VAX CB radio simulator. ETAPE – Convert to/from EBCDIC etc. tape. From Dale Miller.

`[.DOLGEN]` Procedure to reorganize a whole disk and maintain master indexed files created with exact positioning. From William Baker

`[.DUMPER]` Utility to read TOPS-10 and TOPS-20 DUMPER tapes on VMS. Handles versions 5 and 6 DUMPER tapes now too. From James Harvey.

`[.FINGER]` Local or remote SHOW SYSTEM/USERS, name server, and mail querier. Major update of mail items. From Rand Hall.

[.FLOWERS] Security audit on wildcard files. Checksums on SYS$SYSTEM. Account summary. Check login time. Allow use of old shared images.

[.GAMES] Cribbage and poker games. Various authors.

[.GHC] Rewrite of LIST/RSX for native mode VMS. Super multicolumn lister, many options. From Gerson Cohen.

[.GNUSOFTWARE] Many GNU utilities including BISON (a YACC superset), AWK, Emacs, C++, Assembler, GDB, and Gnu C. Gnu C binaries for VMS are present also, providing a FREE, NATIVE-MODE C compiler for VMS. This compiler generates code which is generally superior to VAX11 C's. It is however a beta version (until it finishes compiling all of Berkeley Un*x) and may still have some bugs. Even so it is quite usable.

VAX88A2 Tape

[.CAROSSO] Kevin Carosso's latest PTY driver for VMS V4 and V5 plus a working PHOTO program using it.

[.DISM32] Disassembler, converts .EXE to .MAR. Really works! From Andy Pavlin.

[.ISPELL] Interactive spelling corrector for VMS. Extremely fast and powerful. From John Sherling.

[.JLP] William Wood's LIST file display/search utility updated for newer VMS versions. ZMODEM code for VMS. From Jon Pinkley.

[.JMEADOWS] STATUS monitor/finger type user/system display program. Will work over DECnet. Also SDL un-do'er for C and re-releases of VERB, FILE, and FIND for decompiling DCL tables, changing RMS attributes, or finding files in INDEXF.SYS

[.JPRIGOT] Master_Dist generates a mailing list of everyone in the UAF file. From J. Prigot.

[.JSCLUG] MTG – meeting announcement bulletin board. USERLIST – several new format lists from SYSUAF.DAT. VC – VAX Calculator. Mail utilility – examine if someone has read your mail. Calendar of events program. Compress someone else's mail. Front end to make remote network mail address fruit-salad invisible to users. WHOIS manager. PRIOMGR – adjust priority of compute bound jobs. From William Baker.

[.LILUG] IDXTEX – Create index files from .IDX files. Grant ACL to others easily. WIN – multiple windows on screen. Test if your mail was read. REMINDOLD – Tells if you have reminders outstanding. Group quota system. From John Hasstedt.

[.LOMASKY] The "ultimate" SET DEFAULT. Password change enforcement. SYSUAF report enhancements. From Brian Lomasky.

[.MAQDEV] Queueing to/from global sections between processes, on either VMS or RSX11M+ and over DECnet. User written system services and queue managers. From Neal Schmidt.

[.MCM] Code Management System to handle changes to any kind of source material. Also adds a BUILD facility. (a kind of free CMS/MMS). From Wren Hunt.

[.MNVAX] Force user to change his password. Become invisible to others on the system. Text library handling command procedure, and editing/runoff control program. From MNVax LUG.

[.MOPAC] LSE Templates for Rdb/VMS and slides for two DECUS talks. From Jim McIlvaine.

[.NET_WHAT] FAST Cluster and network wide SHOW USERS (and more) type command. From Andrew Potter.

[.NEWLON] Create orderly SYSTARTUP.COM files, from J. Newlon.

[.NEWLZW] Current versions of LZW compress/decompress utilities for VMS, from Martin Minow.

[.NFREED] AUTOLOGOUT – logout inactive terminals. CRON – issue commands based on time. EVEEDT – EDT keypad on top of EVE. NODECHECK – allow and disallow access to various nodes in cluster driven by identifiers. Convert ASCII to Postscript. (Also does .MEM files). MAKE – from Todd Aven. Like Un*x MAKE (works off MMS files.). From Ned Freed.

[.NIELAND] BULLETIN – Mark London's BULLETIN, a VAXNotes - like utility. CMD – DCL buffer recall utility. MANUAL – online VMS manual file. MENU – SMG based menu driver. MODIFY – global search/replace utility. SPELL – correction to Vassar SPELL utility. ZDEC – zero device error counter. From Ted Nieland.

[.NSWC] NSWC Runoff updates, from Al Zirkle

[.PAGESWAPPER] *Pageswapper* back issues through May 1988. From Larry Kilgallen.

[.PTIUTIUL] Generate a hard BREAK signal on VAX terminal. From Howard Halstead.

[.RAE] MLR Macro language, structured MACRO-32. From Rod Eldridge.

[.RNOTODOC] DSRtoSDML – Convert RUNOFF files to VAX Document format. Output in LN03, Postscript, or plain text.

VAX88A3 Tape

[.DTRSIG] Datatrieve SIG items: Convert VMS Accounting to DTR format. Get DTR to access AI1 logging/data files. Corporate Phone directory. User defined DTR functions. Back newsletters. Additional DTR Plots. Command line recall. Symposium sessions. System manager utilities. From Bart Lederman and the DTR SIG.

[.RCAS88] AnalytiCalc spreadsheet update; it now can call unchanged subroutines. DBAG – Source only for dBase III clone. FORCE – force commands on other terminals. MENU building aids. PLOTIT – presentation graphics for AnalytiCalc or standalone. Search multiple words. Command frontend to VMS. NETS88 – MANY invaluable items from ARPAnet mail including BOSS (multiple sessions at a single terminal) and matching PTY drivers and MUCH, MUCH more. From Glenn Everhart

[.RPI] MODPARAMS manager – build MODPARAMS.DAT for a cluster. Screen Dump filter (VAXstation to print). SETUP – avoid zillions of DCL symbols at login. WATCHER – idle job killer; works with LAT. From Matthew Madison.

[.SDE] Software Development Environment complete procedure using CMS and MMS. From Kevin Angley.

[.SUM] SUM – Show Users More; shows users and LAT location info. From Michael Kimura.

[.TPUSPELL] Spell checker in TPU; check words in the editing buffer interactively. From Dilip Jain.

[.WENTZ] ALL-IN-1 system management utilities. BECOME another user utility. Utilities for developing BLISS source code. SHOW PROCESS /CONTINUOUS which shows all quotas also. More. From Eric Wentz.

[.WORLTON] DOCUMENT – get documents from source files. GPLOT – routines to make it easier to use VAX GKS. MODIFY – replace strings, detab, etc. files; global search/replace. From T.G. Worlton.

[.XEVE] Extended EVE with features from many of the EVE extended editors of earlier tapes. Spelling checker. From Tom Wolfe

RSX/PC/VAX Tape

[VAX88A4...] Directory Tree

[.SOFTWARE_TOOLS] Software Tools User Group multinetwork mailer, in C, last PD version. Does multinetwork, store/forward mail and gateways, and has its own user interface if desired (much more powerful/complete than VAX Mail). From Ken Adelman.

[.TECOC] Complete rewrite of TECO in C for VMS, by Pete Siemsen.

[.PCSIG_DECUS] Large collection of MS-DOS and CP/M utilities contributed by the DECUS PC SIG. Many for Rainbow but many for IBM PC or generic MS-DOS or CP/M also.

[.RCAS88.AMIGA] Several interesting Amiga packages including a port of the RIM relational DBMS and micro Emacs version 3.9i with sources. MicroEmacs runs on Amiga, VAX, PC, and several other machines.

RSX88A Tape directories

Submitter	[UIC]	Contents
Alan Frisbie	[265,20]	MCE Command Line Editor from Hans Hamakers, Rotterdam, The Netherlands.
Alan Frisbie	[373,310]	Files-11 ODS1 & ODS2 specifications, including RMS-11 Internals manual.
Thomas Wyant	[351,144]	LBC – Logical Block Copy (disk-file, disk-disk, file-file).
Thomas Wyant	[351,145]	TEM – Terminal Emulator.
L.M. Cartwright	[300,201]	11M+ Queueing & Routing Software.
William Crocker	[300,210]	LZYCLI – Command-line recall/edit and VMS-style symbols.
Frank Borger	[311,*]	LN03 control, COOKIE, DAMMIT, and other sayings; ECR – Command-line editing for IAS.

The following are the NLC regional tape copy distributors. The top level point of contact for the VAX SIG tapes is:

Robert N. Perry
Tektronix, Inc.
P. O. Box 500
M/S 50-454
Beaverton, Oregon 97077
503-627-5410

The following people are the regional NLC tape distributors. Note that if you are a LUG librarian, YOU must contact THEM and arrange to get copies of the tape. Tapes are NOT automatically sent out. You should expect to furnish tape and possibly postage.

National LUG Council Tape Copy Regional Distributors

Southern Region: Frank Bush, Tennessee Tech. University, Computer Center, Cookeville, TN 38505, 615-372-3972

Northeastern Region: Dennis P Costello, Cornell University, G02 Knight Lab, Ithaca, NY 14853, 607-255-2329

Central Region: John Livengood, 4645 N. Hillcrest, Wichita, KS 67220, 316-526-9380

Northwest Region: Steve Lorentzen, United Administrators Inc., 201 Queen Anne Ave. N. #100, Seattle, Wa. 98109, 206-284-4316

Eastern Region: Rick Sharpe, Toledo Edison, 300 Madison MS: 3180, Toledo, OH 43652, 419-249-5000 x7470

Southwestern Region: Don Shehi, Maricopa Community College, 3910 East Washington St., Phoenix, Az. 85034, 602-267-4404

The 1988 Spring VAX SIG Tape, A Review

Earl Ake, Science Applications Industries, Inc

Duane Baker, North West Ohio Computer Association

David Hitner, Projects Unlimited

M. Edward (Ted) Nieland, Systems Research Laboratories, Dayton, OH

Jonathan Pinkley, Westinghouse Electric Corporation

This is a review of the Spring 1988 VAX SIG Tape (Cincinnati). This review does not cover every aspect of the tape, as the reviewers do not have all the compilers or utilities to check out everything. This review has been done in hope of alerting people to what is on the tape and what you should consider running. This is not a complete review of the tape, but a partial review.

If you find this review helpful and you would like to see such reviews of SIG tapes continue, please leave a message for NIELAND on DCS or send E-MAIL to TNIELAND@WPAFB-AAMRL.ARPA on the Internet. People interested in helping with future reviews can also send messages to the above addresses.

[.ABEL]

Abel is an editing interface built on top of Digital's EVE editor. ABEL makes use of the CALL_USER interface built in to EVE.

[.ANUNEWS]

This is version 5.2 of the NEWS program. VMS NEWS is a computer-based conferencing system. VMS NEWS allows the user to view all postings made by other users of NEWS, and also allows the user to contribute to the conference by posting an item into NEWS. News items are classified by into newsgroups (postings with similar subject matter) and all postings referring to the same conference are logically grouped together for presentation within a newsgroup (this is analogous to VMS Mail's use of folders).

NEWS implements the Standard for Interchange of USENET Messages (RFC1036), and as such has many similarities with the Unix(tm)-based rnews software set. However NEWS is not a port of this rnews software to VMS and the program has been designed with the aim of implementing the functionality of rnews using data and file structures that can be executed efficiently on a VMS system.

The implementation of this standard under VMS implies that a VMS site can be configured into the USENET news network, and will accept USENET news batches from neighboring USENET sites, and generate batches for transmission to downstream sites.

NEWS also implements the Network News Transfer Protocol (NNTP), as documented in RFC977. This implementation uses DECNET as the transport layer, and is implemented as a DECNET object. The NEWS program can be configured to automatically access a NNTP server over DECNET, implementing a distributed news database.

[.ATTC]

This area is a section devoted to LAVC management. The author has submitted many management procedures developed and used on his own LAVC. Documentation is excellent; a professional-quality runoff document is included which contains a table of contents, sections with bullets and numbered points, definitions, philosophy, graphs, a few examples, and most importantly, restrictions and caveats.

The management routines include items such as cluster-wide queue initialization, logical name setup, disk mounting facilities, cluster watch "overlord" daemon, startup routines, and a cluster-wide login file for convenience. These routines give the "local" system manager the option of rebooting the local node into a stand-alone mode and back, automatically reconfiguring the cluster as necessary. Also included are facilities to keep a "spare" - a system can be designated as an emergency backup for the master boot node, and routines are included to keep the spare's disk files up-to-date so that the cluster can be resumed with minimal impact. While the reviewer was unable to test the routines due to not having a LAVC, if the submitted routines are as good as the documentation, this could be an excellent aide to make a LAVC manager's life easier.

[.BRYANT]

This submission includes three TPU utilities: APLTPU, GBLSUBS, GBLSEARCH.

APLTPU This is an EDT emulator under TPU with many features borrowed from EVE and EVEPlus. This submission is an update from the Dallas VAX SIG TAPE.

GBLSUBS This utility allows you to replace all occurrences of a string with another string in a list of files. Parameters of /CONFIRM, /EXACT and /QUERY are supported.

GBLSEARCH This utility allows you to SEARCH files for a string and to edit the files containing that string. Parameters of /CONFIRM, /EXACT and /NAMES are supported.

[.BSL]

[.BZL.LSE]

LSE macros that will help you to write RUNOFF and MACRO-32 code using the LSE editor.

[.BZL.SPELL]

Contains a DECSpell personal dictionary which has defined in it the instructions, directives, and placeholders used in the VAX Macro-32 Language Sensitive Editor template that was also submitted.

[.COY]

This area contains a set of useful user utilities, many with graphic user interfaces to help a user more easily cope with the flood of information available on a VAX. These of the submissions contain logical graphs, using the SMG facility, to enable the user to visualize the operation being performed. Even if a user has no use for these submissions, the structure of some of the screen displays is instructive as a model for inclusion in that person's own programming style. This area contains resubmissions (with improvements) of the author's DM, SD, COLORS, and WPE programs , which were submitted to previous VAX tapes. Some new submissions are also included - SHOWME, VAXNOTES, and TOPC.

DM is a screen-based directory management package. It allows the user to use one and two keystroke commands to provide most of the DCL file-affecting commands, such as delete, copy, purge, print, edit, etc. SD is an executable set default facility, using the screen manager. SD allows the user to easily move around the directory structure, remembering the last directories used and allowing easy return to those areas.

COLORS is a package for changing the colors on VT241's and VT340's.

WPE is a WPS-PLUS implementation using the TPU editor.

SHOWME is a non-privileged utility for showing system information.

VAXNOTES is an area that has some useful programs to help the user more efficiently use DEC's vaxnotes.

TOPC is a more useful rendering of the MON/TOPCPU display.

[.CSDHBO]

This submission is and extended SHOW Command procedure for Clusters. It implements SHOW USERS, SHOW DEVICES, and SHOW HOLDERS. Also a SHOW TERMINAL to show LAT Terminal information.

[.DINKIN]

This submission is a command procedure that determines the terminal server and server port a particular LTxxx terminal is connected to.

[.DOLGEN...]

A replacement to a submission on the Fall 1987 VAX SIG tape. This is a disk reorganizer to place files in an optimum location on the disk. It places INDEXF.SYS, directories, and user files in the middle on the disk. Long life and short life files are placed at either end of the disk.

[.FINGER]

This is a VMS version of the UNIX Finger program with a lot of VMS extras. If you are not using this program you should be. Finger offers the following:

- Customizable SHOW SYSTEM that includes showing the image name.
- Showing LAT terminal locations.
- Showing the number of unread MAIL messages of a user.
- Displaying the date and topic of unread messages sent to a user by the user executing FINGER.
- Displaying a user created PLAN file so that the user can display information that he wants other's to know about.

FINGER does not work correctly under V5.0 but the fixes should be ready for the next SIG Tape. The TCP code, which is supposed to work with the Wollongong WIN/TCP, does not work the current version of WIN/TCP, but the corrected code should be on the next SIG Tape.

[.FLOWERS]

This is an area containing miscellaneous management routines. There is a section concerning monitoring system security, a section dealing with version differences in shareable images, and a bunch of miscellaneous unclassified routines. The author notes in his intro that many routines require VMS 4.4 or greater owing to the use of GOSUB's, and that none of the routines are genericized for general use, requiring some minor editing to work on the target system.

Some interesting routines are available. There is a program which will check for security alarms in all of the old operator logs, assuming that they have not been purged. There is also a routine which will save the current VMS shareable images so that later the user can link to the saved images for compatibility with previous version of VMS.

[.GAMES]

[.GAMES.MNVAX.DIAGONAL.CRIB]

CRIBBAGE game written in BASIC using SMG calls to update the screen.

[.GAMES.NFREED.CRIBBAGE]

Another a very good CRIBBAGE game. Screen updates are handled nicely with the righthand portion of the screen scrolling the messages and the lefthand displaying the cards and board.

[.GAMES.NFREED.POKER]

An interesting poker game written in PASCAL using SMG calls to do the screen updating. The game challenges you against four computer opponents: Ian, Rexford, Chuck, and Leo. These players have four distinct playing personalities. Betting is included. The computer tells you what your best hand is and when the hand is won, tells you what hands the remaining players in the game had.

[.GNUSOFTWARE]

The latest from the Free Software Foundation (FSF). This includes GCC (C Complier) and BISON (A Version of the UNIX YACC) for VMS. Compressed TAR files with the source code for GCC, BISON, EMACS, CLUE, AWK, and GAS are included.

The GCC compiler is reported as creating better optimized code. The compiler uses the distributed VAX-C Runtime Library and you have to reference the VAX C manuals for Subroutine calls. A VMS GCC Runtime Library is needed if anyone is looking for something to do. The compiler is ANSI C Draft compliant and some C programs will not compile do to the differences.

The GCC can be used with the debugger, but it doesn't create a symbol table that the debugger can used (it has the /LIST, /DEBUG, etc. qualifiers but they didn't do anything when used). But this is a FREE C Compiler.

You should look at this distribution, if you look at nothing else!

[.JLP]

This submission includes an update to the LIST Utility from the VAX82B tape and an implementation of ZMODEM for VMS.

The LIST Utility is more flexible that VMS SEARCH. It can search for text in specific columns and has a more general search pattern than SEARCH.

ZMODEM file transfer protocol is more efficient that the standard KERMIT. AT 2400 bps, ZMODEM can achieve 220 characters/sec versus 120 characters/sec for standard KERMIT (VMS 3.3.111).

[.JMEADOWS...]

Joe Meadows has provided the latest version's of his extremely useful utilities VERB, FIND (was INDEX), and FILE.

VERB

DCL command definition language extraction utility for VMS V4 and V5. This one is great for finding out about undocumented switches, commands, etc. Generally a "safe" utility, i.e. a mistake won't hurt you.

A MUST HAVE!

FIND

The most flexible file locating utility I am aware of. It searches the INDEXF.SYS file for file headers matching user specified criteria. Another MUST HAVE! This one is also "safe" in the sense that a mistake won't hurt you.

Following is one use of the FIND utility:

FIND is the only utility I know of that will allow you to find all files that contain a specified identifier in an ACL. This is useful for sites that reuse UIC's and general identifiers. This lets you find files that allow or deny access to a specific identifier, so you do not unintentionally grant a "new" user access to files. For example, user JONES with UIC [51,23] leaves the company. User SMITH had permitted user JONES access to one of his directories via an ACL. Two weeks later, you add a new user THOMPSON to UIC group 51 and the UIC [51,23] happens to be the first available UIC and is therefore used for user THOMPSON. Unless SMITH has removed the ACE permitting JONES access to the files, you have now unintentionally given THOMPSON access to the files that JONES originally had access to. To make matters worse, when the JONES account is removed, the alpha identifier JONES is also removed. If SMITH now does a `DIRECTORY/ACL` of the files that JONES had access to, he will see [51,23] in the ACE instead of [JONES].

When we remove accounts we do the following:

1. Delete all files in the removed user's directory, including the top level directory. This should leave zero blocks for the user.

2. If disk quotas are enabled on the volume, use the disk quota utility (SYSMAN in V5) to rebuild the quotas for the disk. This is necessary because DEC still has what appears to be a caching problem in disk quota charging. Once the volume is rebuilt, verify that the blocks allocated to the UIC are zero. If they are zero, remove the disk quota entry.

3. If disk quotas are not enabled, or the disk quota utility reports blocks still allocated to the user, you must find any remaining files that belong to the user. For example, to find the files belonging to JONES use the find utility:

   ```
   FIND disk/OWNER_UIC=[JONES]
   ```

   You must now determine if these files should be deleted, or have their ownership set to another user.

4. If you are using disk quotas, verify that usage is zero and remove the disk quota entry for the UIC.

   This is normally where most system managers stop.

5. Find files that the user had access to via an ACL.

Since VMS V4 introduced ACL access to files, there may be files on the system that give explicit access to a user. If the UIC is reused, a different person will now have access to the file. To prevent this, you can use the FIND utility to find all the files that contain the identifier that is being removed. For example:

```
FIND disk/ACE=([JONES])/DISPLAY=ACL
```

You must now decide if the ACL can be deleted entirely, or just the ACE removed.

Note that to be thorough, you will need to do this for every volume.

FILE

This utility can modify file headers. It can change things like record attributes, file dates, etc. This can be a very useful utility, but it assumes you know what you are doing. It understands wildcards, so BE CAREFUL, a mistake can definitely hurt you. (Keep this in perspective, the DELETE command can hurt too.)

One thing I use it for is to change the record attributes of archive files I upload with the rz program (ZMODEM, see [.JLP.RZSZ]). rz creates stream_lf files, but the VMSSWEEP program cannot digest stream_lf files.

```
FILE TEST.ARC/TYPE=FIXED/RECORD_SIZE=512
```

Modifies the way that VMSSWEEP interprets the contents of the file and everything works fine.

Note: FILE does NOT modify the contents of the file data itself. Use CONVERT/FDL to change from one file type to another while changing the internal structure as well. For mucking with the file contents itself see the DECUS utility VFE ([VAX87E.VFE]).

[.JPRIGOT]

This area contains one submission, a command procedure to scan the UAF file and produce a SYSTEM distribution list for use with the mail facility. Undocumented except for some comments in the command procedure.

[.LILUG]

This submission includes and update to IDXTEX, the Indexing program for LaTeX; a group quota system; an ACL granting program; a REMIND implementation; a Bulletin Board program; a MAIL Check program; and a Windows program.

The Windows program is written in PASCAL, but an executable image is not included. It looks promising, but you will need the compiler to get it running. The MAIL Check program (MAILRRR) is a resubmission of a previous one, but I had problems getting it to work correctly. Also, MAILRRR is supposed to work across DECnet.

[.MAQDEV...]

Software which provides the ability to transfer data between processes using a global section file. The software runs on VMS and RSX11M-plus systems. The software uses DECnet to communicate with other systems.

[.MCM]

This submission is Morrison's Code Management, a programmers software maintenance tools which manages changes to software modules. The PL/I source is included in a text library. Documentation is in RUNOFF form.

[.MNVAX...]

This directory tree comes from the MNvax LUG based primarily in the Minneapolis/St. Paul area. It contains submissions from several members of the MNvax LUG with the emphasis being on utilities. One routine will force a user to change his/her password, while another allows suitably privileged users to become nearly invisible to other users on the system.

The [.MEDTRONIC.Q] sub-directory of this root has a very useful routine for displaying batch and print queue information in a concise manner. The routine utilizes one line for each batch or print job in the system. If no jobs are currently active an appropriate message is displayed. For sites with many queues where data from a SHOW QUEUE/ALL command may scroll off the screen this utility can be very handy. Also, if the utility is installed with SYSPRV it will allow the non-privileged user to see all jobs in the queues, which may assist them in determining when their job may be completed.

The [.SCSU...] sub-directory of the [.MNVAX] directory tree contains some useful extensions to the EVE editor which may be helpful to some users. Included are procedures to trim trailing spaces from a line; go to a previous or subsequent buffer; perform rectangular cut and paste operations; replace tabs with the appropriate number of spaces; and transpose adjacent characters or lines.

[.SCSU...] also contains an excellent utility called the Video Attribute Textual Formatter (VATF). This product allows the user to display text in an easy to read format and include video enhancements such as bolding and reverse video. VATF is a directive interpreting utility that allows the user nearly complete control over the screen. The program uses SMG functions and therefore can be utilized on any video terminal type that can be defined for SMG use on your system. VATF was designed to be used by people with limited experience with computers and comes with extensive documentation and a help facility. In addition to being able to display text, it is also possible to use VATF to capture user responses for a quiz, survey, etc. This utility is definitely worth looking into!

[.MOPAC]

This submission is an LSE Template for Rdb/VMS and the slides from some of the Cincinnati Symposia. The sessions are DT028 (Application Development using Datatrieve, Smartstar, and Rdb) and LT107 (Using VAXset Tools for Application Development). The slides are in SliTeX format.

[.NET_WHAT]

This submission is another Non-Transparent DECnet procedure to implement show commands on remote nodes. It can do more than just SHOW USERS, it also does SHOW SYSTEM like display. The user can specify multiple nodes to SHOW.

[.NEWLON...]

A lot of interesting routines for the system manager. For example, a generic backup command procedure, .COM files for logical name assignments, queue starting, and other useful things.

[.NFREED]

Submitted here are an autologout program (idle terminal killer), a table driven auto-repeat job handler, EDT keypad on top of EVE, and a Utility to allow or disallow access to nodes in a cluster by individual users.

Also submitted are MAKE by Todd Aven and a ASCII to Postscript converter.

[.NIELAND]

This area is a submission of various utility items from varied sources. Some items are electronic submissions from articles printed in the VAX Professional, and others are the latest bug-fix and enhanced resubmissions of "Classic" VMS utilities, as well as some new never-before seen utilities.

Latest resubmissions include BULLETIN, VASSAR SPELL, VMSTPC, ZDEC. All have had some significant work done to correct various erratum. All contain germane revisions to the various documentation sets to reflect any new features. BULLETIN does not run under VMS 5.0, but a new version will be on FALL SIG Tape that will work under 5.0. ZDEC also breaks under 5.0, but can be fixed by substituting SCG$GL_CURPCB with CTL$GL_PCB on line 291.

Some new submissions include a DES encryption scheme which also "hexifies" the file into printable characters so that is can be sent via standard mail packages, an SMG-based menu facility, and a command recall buffer save/flush/restore utility (CMD). CMD works under version 4.X, but does not work under 5.0, but an update for 5.0 is expected on the next SIG Tape.

An interesting submission is an example of a "Local User's Manual" to help users understand what programs, routines, and hardware are available for use at that particular site. The manual is written using LaTeX, authored by the submitter, and is an interesting concept in user services.

[.NSWC]

NWSC RUNOFF is submitted here. NSWC RUNOFF is base3d on BONNER RUNOFF (VAX86C version). The changes are documented in the files CHANGES.NWSC.

[.PCSIG_DECUS...]

[.EDITWP]SEDT219.ARC

Looking for an EDT like editor for your Rainbow or IBM compatible PC? This is version SEDT V3.3 (edit level 219) for MS-DOS. SEDT is compatible with EDT to about the same extent as VMS V5 EVE in EDT Keypad mode.

This version will work on IBM and compatible PC's as well as the DEC Rainbow. The files RBHLP.EDT and RBKEY.EDT files must be renamed to correspond to the keyboard you are using. When you first invoke SEDT, it will attempt to determine the type of computer, the type of display, and the type of keyboard present. It will then look for files based on the type of keyboard you have. If you are not running on a Rainbow, it will complain that it did not find the file xxKEY.TXT where xx can be one of XT for standard PC/XT keyboard, NP for the 101 "enhanced" keyboard, and AT for the original AT keyboard. Note that your ROM BIOS must support the keyboard you are using in order for SEDT to correctly determine which keyboard you are using. There is NO Documentation on the TAPE.

This editor is one of the better editors for the PC that I have seen. It is possible to edit files larger than will fit in available memory, and is the only PC editor I have used that has journaling. It has features that do not exist in standard EDT, for instance a "ruler" for setting tabs, left margin, right margin, and wrap margin; an overstrike mode, multiple windows etc. Worth looking at. There is also a version for VMS, although it is not on the tape.

The complete distribution including documentation and all keyboard type definition and help files can be downloaded from either Kevin Porter's Waystar BBS @ (617)481-7147 in Marlborough, MA or Gary Stebbin's Glacier Peak Rainbow BBS @ (206)644-8431 in Bellevue, WA. I downloaded edit level 224 from Gary Stebbin's board on July 20, 1988. Both of these boards have the VMS version as well. Kevin's board has the Atari ST version too.

[.PTIUTIL]

Submitted here are command procedure that shows available disk space and a FORTRAN program that will generate a hardware BREAK signal on terminals.

[.RAE]

Structured programming in MACRO-32. Is this for real? What we have here are macros to implement structured programming in a non-structured language. The question I ask myself is why not used a structured language in the first place. For those that need the flexibility of MACRO and yet also want it readable, there is very good code here.

[.RCAS88]

This directory tree was submitted by Glenn C. Everhart and, as usual, contains a wealth of interesting files that may be of interest to the DECUS community. Among the contents of this directory tree are programs and utilities for the Amiga as well as the latest versions of various flavors of Analyticalc. If you have never tried Analyticalc, you really should. It is a very powerful spreadsheet program with many nice features and, of course, the price is right!

Also available in this directory tree are the DECUS submissions comprising VAX-212 (PLOT_IT and SPELL:Interactive Dictionary); VAX-240 (SRC: A Functional Programming System); VAX-246 (Menu Utility); VAX-268 (VAXMAC); VAX-308 (Remote Control Wizzard); and VAX-309 (CLOC). Please refer to the latest DECUS Program Library Catalog for full descriptions of these utilities.

In the [.NETS88...] sub-directory of the [.RCAS88] directory tree you will find a treasure of information for those willing to dig and explore! Found in this area are system management hints and warnings, bug fixes to VMS, techniques discovered to solve specific problems, and even a preview of VMS V5.0. There are a whole series of useful and interesting routines in this area as well. Some include ARC (an archive utility), BOSS (an interactive job controller), PFC BULLETIN (a bulletin board utility), and MAKE (similar to the UNIX MAKE utility). There is plenty of material in this area for the interested to spend hours researching. As a well known contributor is fond of saying, "Enjoy!".

[.RPI]

This area contains varied submissions. There is a routine (installed with privilege) to allow a user to CLAIM the files in his directory if they were created by other users. And, yet another flexible idle job killer.

There are a few cluster management tools: a utility to build MODPARAMS files for the cluster, a utility to allow vaxstation screen dumps to be routed to a central print queue, and a centralized login facility.

As a side point, one of the most interesting things about these submissions is that most of the programs are written in PL/I, which is rare on SIG tapes.

[.SDE]

This submission is a Software Development Environment the incorporates DEC/CMS and DEC/MMS. A session was given at Cincinnati (LT151) on this submission.

[.SOFTWARE_TOOLS]

This contains the Software Tools MAILER. This mailer is a store and forward mailer that can interface to multiple networks. It is distributed in a VMSINATLable saveset.

[.WENTZ]

[.MISC]

SHOQUOTA

I attempted to use the SHOQUOTA utility but gave up after trying a few things. First to get it to link, you must link it with SYS$SYSTEM:SYS.STB since it uses the exec routine EXE$NAMPID. After successfully linking it, I tried to run it but got the message "insufficient privilege". I turned on CMKRNL (I know Eric Wentz) and ran it again, with the same results. I gave up there (I don't like `SET PROCESS/PRIV=ALL`).

This looks like a useful utility if you can figure out how to get it to work.

If you are going to recompile this you will need to define a logical name to point to the [.WENTZ.BLISSLIB] directory since it includes the GEMAC.REQ file.

FINDUIC

I didn't try this one out since I use Joe Meadows FIND program for this purpose. It would be interesting to see which is faster. Both of them map the INDEXF.SYS file.

[.WORLTON]

Code written to ease the conversion from CA-DISSPLA to DEC VAX GKS.

[.XEVE]

The eXtended EVE editor adds a spell checker to EVE and includes an EDT type keypad if desired. This entry is an update and bug fix to the eXtended EVE editor found on the Fall 1987 VAX SIG tapes.

Selective Procedure Verification Revisited

Bob De Wolf, Systems Engineer

This article contains a couple of improvements to the Selective Procedure Verification technique described in the June 1987 issue of the *Pageswapper*. To recap, Selective Procedure Verification allows you to choose which of a hierarchy of nested DCL procedures you want to be verified. The other procedures in the hierarchy will not be verified. This is accomplished by inserting entry and exit sequences in the procedure to control verification. The sequences given in this article result in the following improvements:

- The procedure will not produce an error message if the symbol DCL_VFY does not exist when the procedure runs.
- When verification is off for a procedure, no statements will be verified.

To achieve the desired results insert the following DCL statements at the beginning of each procedure:

```
$ SAVE_VER='F$VERIFY(0)

$ IF F$LOCATE("(name)","''DCL_VFY'") .NE. -
```

```
F$LENGTH("''DCL_VFY'") -

THEN SET VERIFY
```

When the global DCL symbol DCL_VFY is defined to be the name of the file containing the procedure, enclosed in parenthesis, as defined in the above statement, verification will be turned on. Several procedures can be listed in the symbol, enclosed in parenthesis. The following exit sequence must be taken by whenever the procedure exits:

```
$ IF SAVE_VER THEN SET VERIFY

$ EXIT
```

The above exit sequence restores verification to the state that existed when the procedure was entered.

Selective procedure verification disables the standard (SET VERIFY/SET NOVERIFY) method of control such that verification in procedures that have the above entry and exit sequences will be controlled by the DCL_VFY symbol.

To make it easier to turn verification on and off, a function key can be defined as follows:

```
$ DEFINE/KEY/NOTERMINATE F17 "DCL_VFY:==("
```

Using this key definition, pressing F17 followed by a carriage return turns verification off. F17 can also be used to turn on verification.

Using the (Undocumented) Checksum Utility

George B. Bone

Recent articles in various publications have shown the undocumented CHECKSUM verb as a means to determine if an image file has been modified (presumably by a "virus"). Unfortunately, none have shown a method of dealing with the large number of executable images found in a typical VAX system.

In fact, there are over 200 executable files in the SYS$SYSTEM area alone. This does not include layered applications directories, site written code, or the SYS$LIBRARY area. It is clear that some automated means of comparing checksums is required.

At our site, we have done a little quick DCL programming to provide a means to establish a data baseline (unfortunately, due to limitations in DCL it's a sequential file) and a means to check that baseline on a regular basis.

What follows, then, are two DCL procedures which 1) create a file of checksums and a complete library of sectional checksums and 2) compare existing checksums against this baseline file. It is hoped that more information will be available in the future so that CHECKSUM may be called as a system service from an executable image, and the file may be changed to an indexed file for better access.

One item that was discovered during the writing of these procedures; If an attempt is made to CHECKSUM/FILE a file that, for some reason, can't be CHECKSUMMED (for example, BOOT58.EXE) the process appears to go into an infinite loop, which not even ON ERROR

GOTO's can help. For this reason, a CHECK/IMAGE is done prior to every CHECK/FILE. The CHECK/IMAGE also fails, but it activates the ON ERROR GOTO, so that the loop never happens.

Please note that, due to errors that may occur in transmission or printing of these procedures, the author cannot assume any responsibility for the following material or it's use.

```
$!  CHKSM_CRE_DB.COM
$!
$!  This procedure creates the database that CHKSM_CHK_DB uses to verify
$!  image CHECKSUMS. Also creates the list files of individual image
$!  sections used to further investigate if a problem is found.
$!
$!  Written by G. Bone 5/11/88
$!
$ PROCEDURE_START:
$!
$ SET ON
$ SET CONTROL=Y
$ ON CONTROL_Y THEN GOTO EXIT_LOOP
$!
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT "Enter the directory name (include colons if logical name)"
$ INQUIRE DIRNAME "Press RETURN to exit, type QUIT to quit"
$ WRITE SYS$OUTPUT " "
$!
$ IF DIRNAME .EQS. "" THEN GOTO EXIT_PROCEDURE
$!
$!  Allow the option to get out -- but it can leave 'loose ends'
$!
$ IF DIRNAME .EQS. "QUIT" THEN EXIT
$!
$!  Create a list of executables in specified directory
$!
$ DIRSPEC="''DIRNAME'*.EXE;0"
$ DIR/COLUMNS=1/OUTPUT=SYS$SCRATCH:DIRECTORY.LIS 'DIRSPEC'
$ OPEN/READ DIR SYS$SCRATCH:DIRECTORY.LIS
$!
$!  First three lines are irrelevant
$!
$ READ/END_OF_FILE=EOF/ERROR=EOF DIR RECORD
$ READ/END_OF_FILE=EOF/ERROR=EOF DIR RECORD
$ READ/END_OF_FILE=EOF/ERROR=EOF DIR RECORD
$!
$!  Check to see if Datafile exists.  If so, allow user to append onto file.
$!
$ MODE="WRITE" !  Default mode is to create a new data file
$ FILE_MESSAGE="Creating new CHECKSUM.DAT."
$!
$ CHECKFILE=F$SEARCH("WHEREVERYOUWANTIT:CHECKSUM.DAT")
$ IF CHECKFILE .EQS. "" THEN GOTO OPEN_FILE
$!
$ WRITE SYS$OUTPUT " "
$ INQUIRE ANSWER "CHECKSUM.DAT already exists.  Continue using?[Y/N]"
$ WRITE SYS$OUTPUT " "
$ IF .NOT. ANSWER THEN GOTO OPEN_FILE
```

```
$!
$ MODE="APPEND"
$ FILE_MESSAGE="Using existing CHECKSUM.DAT."
$!
$ OPEN_FILE:
$!
$ WRITE SYS$OUTPUT FILE_MESSAGE
$ WRITE SYS$OUTPUT " "
$!
$ OPEN/'MODE'/ERROR=NO_OPEN DATAFILE WHEREVERYOUWANTIT:CHECKSUM.DAT
$!
$ WRITE SYS$OUTPUT "Creating CheckSum Database for ''DIRSPEC'"
$ WRITE SYS$OUTPUT " "
$ START_LOOP:
$ ON ERROR THEN GOTO NO_CHECK !Problem doing checksum
$!
$ READ/END_OF_FILE=EOF/ERROR=EOF DIR FILENAME
$ IF FILENAME .EQS. "" THEN GOTO EXIT_LOOP ! Last file in directory was read
$!
$!  Create the real path name
$!
$ PATHNAME = DIRNAME + FILENAME
$!
$!  Create the IMAGE/LIST file name.
$!
$ DOT == F$LOCATE(".",FILENAME)
$ NEWNAME == F$EXTRACT(0,DOT,FILENAME)
$ OUTNAME == "WHERVERYOUWANTIT:CHK_''NEWNAME'.LIS"
$!
$ CHECK/IMAGE/OUT='OUTNAME' 'PATHNAME'
$ CHECK/FILE 'PATHNAME'
$ WRITE DATAFILE PATHNAME, " %%%", CHECKSUM$CHECKSUM
$ GOTO START_LOOP
$!
$ NO_CHECK:
$!
$!  Place path name on file, but 'flag' with a checksum of 0
$!
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT "***** Problem in CheckSum for ''FILENAME'."
$ WRITE SYS$OUTPUT "***** list file may be incorrect."
$ WRITE SYS$OUTPUT " "
$!
$ WRITE DATAFILE PATHNAME, " %%%0000000000"
$!
$ GOTO START_LOOP
$!
$ EOF:
$!
$!  End of file detected on the directory list
$!
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT "%%% Abnormal End-Of-File detected on directory list.  "
$ WRITE SYS$OUTPUT " "
$ GOTO EXIT_LOOP
```

```
$!
$ NO_OPEN:
$!
$!  CHECKSUM.DAT could not be opened
$!
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT "%%% Problem opening CHECKSUM.DAT. "
$ WRITE SYS$OUTPUT " "
$!
$ EXIT_LOOP:
$!
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT "Cleaning up misc details.  "
$ WRITE SYS$OUTPUT " "
$!
$ SET NOON
$ CLOSE DATAFILE
$ CLOSE DIR
$!
$!  Clean up
$!
$ DELETE SYS$SCRATCH:DIRECTORY.LIS;0
$!
$ GOTO PROCEDURE_START
$!
$ EXIT_PROCEDURE:
$!
$ WRITE SYS$OUTPUT " "
$ WRITE SYS$OUTPUT "Sorting CHECKSUM.DAT, purging old copies.(/KEEP=2)"
$ WRITE SYS$OUTPUT " "
$!
$ SORT WHEREVERYOUWANTIT:CHECKSUM.DAT -
WHEREVERYOUWANTIT:CHECKSUM.DAT
$ PURGE WHEREEVERYOUWANTIT:CHECKSUM.DAT/KEEP=2
$!
$ WRITE SYS$OUTPUT "End of CHKSM_CRE_DB."
$ WRITE SYS$OUTPUT " "
$ EXIT
$!  CHKSM_CHK_DB.COM
$!
$!  Performs verification of CheckSums of executables.
$!
$!  Written by G. Bone 5/11/88
$!
$ SET PROCESS/NAME="CheckSummer"
$ SET ON
$ OPEN/WRITE OUTFILE SYS$SCRATCH:CHECKSUM.LIS
$ OPEN/READ DATAFILE WHEREVERYOUWANTIT:CHECKSUM.DAT
$!
$!  Prepare the cover message for the report file.
$!
$ DAYTIME=F$CVTIME()
$ WRITE OUTFILE "The CheckSum verification routine was run at ''DAYTIME'. The"
$ WRITE OUTFILE "following verification/file errors were found:"
$ WRITE OUTFILE " "
```

```
$!
$ CK_ERRORS=0
$!
$ START_LOOP:
$ ON ERROR THEN GOTO NOT_CHECKED !Prevents endless loop by CHECK/FILE
$!
$!  Get Filename/CheckSum record
$!
$ READ/END_OF_FILE=EXIT_LOOP DATAFILE RECORD
$!
$!  Separate the two.  '%%' is used as a delimiter between the two fields.
$!  Since we want to check the current version, we have to remove reference to
$!  specific version number from FILENAME.
$!
$ PERCENT=F$LOCATE("%", RECORD) ! Find location of delimiter
$ SEMICOLON=F$LOCATE(";", RECORD) ! Determine where the end of the name is
$!
$ STARTCHKSM=PERCENT+3
$!
$ FILENAME=F$EXTRACT(0,SEMICOLON,RECORD)
$ CHECKSUM=F$EXTRACT(STARTCHKSM,10,RECORD)
$!
$!  Check for file's existence
$!
$ CHKNAME=F$SEARCH(FILENAME)
$ IF CHKNAME .NES. "" THEN GOTO FILE_XIST
$!
$!  Not there -- Make note in report
$!
$ CK_ERRORS=CK_ERRORS+1
$ WRITE OUTFILE "''CK_ERRORS'. NOFL -- ''FILENAME' could not be found."
$ GOTO START_LOOP
$!
$ NOT_CHECKED:
$!
$!  Make entry in report that the file was not checked for some reason
$!
$ CK_ERRORS=CK_ERRORS+1
$ WRITE OUTFILE "''CK_ERRORS'. NTCK -- ''FILENAME' could not be checked."
$ GOTO START_LOOP
$!
$ FILE_XIST:
$!
$!  The CHECK/IMAGE causes a FATAL ERROR or ERROR if the file is unavailable for
$!  CheckSumming for some reason.  CHECK/FILE causes an endless loop of error
$!  messages under the same conditions, so we have to perform a CHECK/IMAGE
$!  first.  This may be removed if the feature is corrected.
$!
$ CHECK/IMAGE/OUT=NL: 'FILENAME'
$ CHECK/FILE 'FILENAME'
$!
$ IF CHECKSUM .EQS. CHECKSUM$CHECKSUM THEN GOTO START_LOOP
$!
$!  Write the filename of a file that does NOT agree with records onto the
$!  output file.
```

```
$!
$ CK_ERRORS=CK_ERRORS+1
$ WRITE OUTFILE "''CK_ERRORS'. CKSM -- ''FILENAME' has a verification error."
$ GOTO START_LOOP
$!
$ EXIT_LOOP:
$!
$ SET NOON ! We don't care about checking for ERROR's now
$!
$ IF CK_ERRORS .EQ. 0 THEN WRITE OUTFILE "No errors found."
$!
$ CLOSE DATAFILE
$ CLOSE OUTFILE
$!
$!  Send the report to the system manager
$!
$ MAIL/SUBJECT="CheckSum Verification Run" SYS$SCRATCH:CHECKSUM.LIS SYSTEM
$!
$!  Clean up
$!
$ DELETE SYS$SCRATCH:CHECKSUM.LIS;*
$!
$!  Submit for random time of at least 4 hours ahead and no greater than
$!  54 hours ahead.  We don't want to have a set pattern.
$!
$ SET VERIFY
$ HOUR_TODAY=F$CVTIME(,,"HOUR")
$ HUNDREDS=F$CVTIME(,,"HUNDREDTH")
$ HOURS_TOTAL=( HUNDREDS / 2 ) + HOUR_TODAY + 4
$ DAYS=HOURS_TOTAL / 24
$ HOURS=HOURS_TOTAL - ( DAYS * 24 )
$ DTIME="''DAYS'-''HOURS':00:00.00"
$ SUBMIT/AFTER="TODAY+''DTIME'"/NOLOG/NOPRINT -
WHEREVERYOUWANTIT:CHKSM_CHK_DB.COM
$!
$ EXIT
```

What we have now is one procedure for creating a baseline, and one for checking files against that baseline. If a file shows up on the report as having been changed, the system manager can perform a CHECK/IMAGE on the file and compare those sectional checksums against the ones kept in WHEREVERYOUWANTIT:CHK_<filename>.LIS. This can provide a little more insight to what is happening.

Some notes on these procedures:

1) The checksum of the file SYS.EXE will change with every re-boot. Relax. This appears to be due to the boot procedure, and not to any "outside agency".

2) Once written into a baseline file, to change the record simply edit the file (using EDT or whatever), inserting what is now the correct checksum. The new sectional files may be copied in, then purged to obtain new data.

Hopefully, any files that show up on this report will be the result of authorized modifications. If they aren't authorized, well... they will still show up on the report, and you (as system manager) will be warned that SOMETHING has happened, and can take the appropriate precautions.

Note that simply maintaining this baseline cannot guarantee that you will find every "virus" that comes by. The intruder may be clever enough to modify this database during the intrusion. Or he (or she) may be clever enough to modify code without changing the checksum. This is meant to be only part of a solution that should include physical security, password policing, monitoring of external modem connections, and budgeting of available privileges on an "as needed" basis.

Remember – If you don't think you have a security problem, you've just found your first one.

Bug Fix for JUICER_2 Version 2-016

Mike LeVine, Naval Weapons Center, China Lake, CA

JUICER_2 Version 2-016 as distributed in the Fall 1987 VAX SIG tape has a bug that causes the program to die with out any damage to the target disk. To fix the problem, apply the following SLP patch to the source. The corrected program will be submitted to DECUS.

```
- 5, 5
      ACTION /V02-017/
- 125
;   18-Feb-1988 MNL                            V02-017
;     Bug reported by Cliff Bicker of NWC-JUICER dies due
;     to access violation. Problem found in routine
;     FORCE_FILE_MOVE-used high LBN for pointer to
;     queue entry
- 3559
    MOVL   R9,R8                                ;V02-017
- 3569
    MOVL   R9,R8                                ;V02-017
- 3578,3578
5$: MOVL   R8,R7                                ;V02-017
/
```

RT-11 DUCM/DYC GRAF11 DEL DIR PLOT-10 IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB-
DATMAN/VAX EDTPlus SPICE2 TREEDUPL LISPEX MCLS TYPE_PLUS MINC DISK_USE FRAG EDTEX
PORT LOCATOR TECO CHPLOT NANNY DIR11-W WATCHDOG INACTIVE ACCOUNTS IMGSPL ICE TEX
EDITOR VAX-LIB-4 GRAPHIC UTILITIES SETAUX.ARC STATPK FIGure KERMIT Distribution TENBACKU
JUICER VTEDIT 2022 VAX-LIB-3 VISTA EDITOR MTU TDE RSTSOPEN DRAWTREE WATCHDOG PRM-1
SMARTMAILER TEN SPELL DECPoint of Sale JUICER PARALLEL Library V2 RTMULTI and Addo
SMARTMAILER for RSTS/E CU FILTRA Spring 86 RT-11 SIG CP/M KERMIT S Invasion for PRO Bonner La
SPLICE RUNOFF VAX-LIB-3 VAX-LIB-2 IMAGE SPELL TURBOCOM FNDFIL PC-8088 Collection #10 VT20
TOOLKIT PLATOOLS SMARTMAILER DEPROC LaTex KERMIT-11 FANCY FONTS XMIT CU ReGis to HPG
CED International RUNITOFF JP5-JP6 FODT PASCAL-OS/8 ANISMT WPSIM PARALLEL LIBRARY
DECSYSTEM-20 SIG Spring 85 CAMERA DELPHIN HACK BIBENTRY APFELN DIGITIZING Acid Docume
Generator VAX-LIB-2 AMAR-10 AMAR-20 DATMAN/VAX IMAGE RT-11 DUCM/DYC GRAF11 DEL DIR PLOT-1

DECUS

IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB-5 DATMAN/VAX EDTPlus SPICE2 TREEDUP
LISPEX MCLS TYPE_PLUS AMAR-20 DISK_USE FRAG EDTEXT PORT LOCATOR TECO CHPLOT NANN
DIR11-W WATCHDOG_INACTIVE ACCOUNTS IMGSPL ICE TEXT EDITOR VAX-LIB-4 GRAPHIC UTILITIE
SETAUX.ARC STATPK FIGure KERMIT Distribution TENBACKUP JUICER VTEDIT 2022 VAX-LIB-3 VIST
EDITOR MTU TDE RSTSOPEN DRAWTREE WATCHDOG PRM-11 SMARTMAILER TEN SPELL DECPoint of Sa
JUICER PARALLEL Library V2 RTMULTI and Addons SMARTMAILER for RSTS/E CU FILTRA Spring 86 RT-1
SIG CP/M KERMIT S Invasion for PRO Bonner Labs APFELN RUNOFF VAX-LIB-3 VAX-LIB-2 IMAGE SPEL
TURBOCOM FNDFIL PC-8088 Collection #10 VT200 TOOLKIT PLATOOLS SMARTMAILER DEPROC LaTe
KERMIT-11 FANCY FONTS XMIT CU ReGis to HPGL CED International RUNITOFF JP5-JP6 FODT PASCAL-OS/
ANISMT TECO WPSIM DECSYSTEM-20 SIG Spring 85 CAMERA DELPHIN HACK BIBENTRY APFELN KERMI

Program Library

S DIGITIZING Acid Document Generator VAX-LIB-2 AMAR-10 AMAR-20 DATMAN/VAX IMAGE VT200 TOOLKI
COMPRO EVENTS PC8088 Collection #9 TECO Family Tree Workstation Bookings System EXPORT Data Inputt
Generator CMSBROWSE PERSONNEL INVENTORY MS/DOS COMMS Selection Electronic Grade Book CP/
KERMIT LaTex JUICER SPELL PORTACALC DPRINT DUNGEON MINC BUDGET BUG CALC C Langua
System DPROC "DEP" DECENC DECmate II OS/278 DIAL DTC GAMMA-11 GDADL LISP for RSX-11 MEM
KERMIT S VAX-LIB-6 SPICE 3A6 VT200 TOOLKIT RUNNOFF SPLICE SPY:RSX TCOPY SPELL VT-200 COMPR
EVENTS CMSBROWSE UNDELETE DIAL BLOCKER SCAN CODER BITMAP DTC/PC ADDRESS BOO
LaserWriter PORTACALC SPICE 3A6 PRO/Smart Mailer CBASIC2 Accts JP5-JP6 Payable/Receivable McGraw-Hi
Payroll SEDT: EDT/WPS Screen CLNDRS:A Calendar Program INDEX AKCOUNT CORPHONE E-Systems Grab Ba
RGT RDG PLTXSMB ICON DEVICS DATATRIEVE Library Collection CMSBROWSE EXPERT FPaint IMAG

SOFTWARE ABSTRACTS

DBMS/Spreadsheet for MS/DOS AMAR-10 AMAR-20 RDIR/SQMAP PC-8088 Collection #11 UP TIME REPORTE
RT-11 DUCM/DYC GRAF11 DEL DIR PLOT-10 IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB
DATMAN/VAX EDTPlus SPICE2 TREEDUPL LISPEX MCLS TYPE_PLUS EXPORT DISK_USE FRAG EDTEX
PORT LOCATOR TECO CHPLOT NANNY DIR11-W WATCHDOG_INACTIVE ACCOUNTS IMGSPL ICE TEX
EDITOR VAX-LIB-4 GRAPHIC UTILITIES SETAUX.ARC STATPK FIGure KERMIT Distribution TENBACKU
JUICER VTEDIT 2022 VAX-LIB-3 VISTA EDITOR MTU TDE RSTSOPEN DRAWTREE WATCHDOG PRM-1
SMARTMAILER TEN SPELL DECPoint of Sale JUICER PARALLEL Library V2 RTMULTI and Addo
SMARTMAILER for RSTS/E CU GRAPHKIT FILTRA Spring 86 RT-11 SIG CP/M KERMIT S Invasion for PR
Bonner Labs RUNOFF VAX-LIB-3 VAX-LIB-2 IMAGE SPELL TURBOCOM FNDFIL PC-8088 Collection #10 VT2
TOOLKIT PLATOOLS SMARTMAILER DEPROC LaTex KERMIT-11 FANCY FONTS XMIT MEMO ReGis to H
CED International RUNITOFF JP5-JP6 FODT PASCAL-OS/8 ANISMT CODER WPSIM DECSYSTEM-20 SIG S
85 CAMERA DELPHIN HACK BIBENTRY APFELN REPORTER DIGITIZING Acid Document Generator VAX-L
AMAR-10 AMAR-20 DATMAN/VAX IMAGE VT200 TOOLKIT COMPRO EVENTS PC8088 Collection #9 TECO C
Tree Workstation Bookings System EXPORT Data Inputter Generator CMSBROWSE PERSONNEL INVENT
MS/DOS COMMS Selection Electronic Grade Book CP/M KERMIT LaTex JUICER SPELL PORTACALC DP
DUNGEON MINC BUDGET BUG CALC C Language System DPROC "DEP" DECENC DECmate II OS/278 DIA
DTC GAMMA-11 GDADL LISP for RSX-11 MEMO PORTACALC VAX-LIB-6 SPICE 3A6 VT200 TOOLKI
RUNNOFF SPLICE SPY:RSX TCOPY SPELL VT-200 COMPRO EVENTS CMSBROWSE UNDELETE DIA
BLOCKER SCAN CODER BITMAP DTC/PC ADDRESS BOOK LaserWriter PORTACALC SPICE 3A6 PRO/Sma

"A Solution with a Future"

Mailer CBASIC2 Accts TREEDUPL Payable/Receivable McGraw-Hill Payroll SEDT: EDT/WPS Screen CLNDRS:
Calendar Program INDEX AKCOUNT CORPHONE E-Systems Grab BaRGT RDG PLTXSMB ICON DEVIC
DATATRIEVE Library Collection CMSBROWSE EXPERT FPaint IMAGE DBMS/Spreadsheet for MS/DOS AMAR-1
AMAR-20 RDIR/SQMAP PC-8088 Collection #11 UP TIME REPORTER RT-11 DUCM/DYC GRAF11 DEL DIR PLO
10 IMAGE LIBED FSTATS MS/DOS TIC-TAC-TOE QIX VAX-LIB-5 DATMAN/VAX SPICE2 RT-11 DUCM/DYC G

DECUS PROGRAM LIBRARY

NEW LIBRARY PROGRAMS AVAILABLE FOR THE VAX/VMS FAMILY OF COMPUTERS

DECUS No: V-SP-77 **Title:** Languages and Tools SIG Tape **Version:** May 1987

Author: Various

Operating System: VAX/VMS, Various **Source Language:** Various **Keywords:** Symposia Tapes - L & T

Abstract: This is the Language and Tools SIG tape, 1987. The programs have been placed in two major directory structures named LT87A and LT86A. The programs on this tape are mostly VAX/VMS specific. The following is a brief summary of highlights:

[LT87A]

[.ADAM] Adam, an extended Eve. Submitted by Art Ragosta, Ames Research Ctr.

[.AVEN] Submitted by Todd Aven, The Software Sweatshop.

[.ARC] Archive utility. Submitted by Thom Henderson, ported by Mike Stump, from System V, from BSD, from IBM PC ARC 5.12.

[.MAKE] VMS Make.

[.SCHEME] A Lisp-like system. See "Structure and Interpretationof Computer Programs". Submitted by Abelson, Sussman, and Sussman.

[.CORY. F87] Submitted by Earl Cory, EATON Corporation.

[.CRIB] Cribbage game.

[.USER] Enhancements to LSE.

[.DOC] Documentation for LSE.

[.TXT] Help file for LSE.

[.GCE] Submitted by Glenn Everhart, GE Government Electronics Systems Div.

[.BULLETIN0131] Bulletin V1.31 from MIT.

[.CCPUBVX] Descendant of Dr. Dobbs' Small C for VAX.

[.ISPELL] Spelling checker. Submitted by Jon L. Sherling.

[.MSGINC] Build C, FORTRAN, and PASCAL include files from VMS message files. Submitted by Donald R. Gummow.

[.VMSDS] VMS disassembler. Submitted by Andrew Pavlin. MFLOP.FOR-Megaflop benchmark. DESNOTES.TXTNotes on Data Encryption Standard.

[.ICON.V7] Icon Version 7.0, see "The Icon Programming Language" and "The Implementation of the Icon Programming Language". Submitted by Griswold and Griswold, Prentice-Hall.

[.MG1B] MicroGNUEmacs 1b, a small Emacs-like editor.

[.SEWELL. TUG] Submitted by Wayne Sewell, E Systems.

[.MWEB] Modula-2 version of Web.

[.SCAN] Scan a TeX file for Changes.

[.WEBM] Merge Web Change files.

[.SRLJAS. EDTSHELL. WAIT] Run EDT at a higher priority, then return to normal. Submitted by Jack Schwartz.

[.TECOC] TECO in C. Submitted by Pete Siemsen.

[.WATSON] Extensions to Eve and EVEPlus. Submitted by Allen Watson.

[LT86A]

[.APCE] Extended-EVEPlus. Submitted by Rick Stacks, Arkansas Department of Pollution Control & Ecology.

[.DEPROC] TeX and LaTeX support for DECUS proceedings. Submitted by Barbara N. Beeton, American Mathematical Society.

[.EVEPlus] EVEPlus -- extensions to EVE. Submitted by Digital Equipment Corporation as unsupported.

[.FERMILAB..] Submitted by Frank J. Nagy, Fermi National Accelerator Laboratory.

[.EDTX] EDT eXtended.

[.EXTRACTOR] Extract source comments for RUNOFF or TeX.

[.FERMILIB] Definitions and subroutines you wish came with VMS.

[.GETUAI] $GETUAI extensions.

[.MAKEINCS] Translate Bliss .REQ files to FORTRAN and C.

[.PROCEDURES] General utility command procedures.

[.TABS] Spaces {--} tabs, understands FORTRAN.

[.TRMPRINT] Print on terminal's printer port.

[.XDFO] Extract Definitions From Object code, reads an object file, creates definitions for several languages.

[.FLOWERS] Subset of Emacs written in TPU. Submitted by Jeff Flowers, Pilot Executive Software, Inc.

[.INFOC] Multi-network bulletin board comments on C language submitted by Joel M. Snyder, University of Arizona.

[.LSL..] Submitted by Kent McPherson, Lear Siegler, Inc.

[.LATEX] Additional styles, help, and LSE templates.

[.LSE] LSE templates for TPU and LSE.

[.TPU] TPU extensions.

[.UTIL] Misc. utilities.

[.MERRITT] How to use callable CMS, CMS library editor. Submitted by Del Merritt, Computer Sciences Corporation.

[.NIELAND] EDT-Plus -- extensions to TPU EDT emulator. Submitted by M. Edward (Ted) Nieland, Ames Laboratories & Systems Research Laboratories, Inc.

[.RIGS] C library extensions: system, string functions. Submitted by Wayne E. Baisley, Rockwell International/Graphic Systems Division.

[.RLAURB] IdxTeX -- index generator for LaTeX documents. GloTeX -- glossary generator for LaTeX documents. Submitted by R.L Aurbach, Monsanto.

[.SWTOOLS] Software Tools Users Group latest version for VMS. Submitted by David Martin, Hughes Air-craft.

[.VANTASSLE] C utilities: GREP, CBU (C beautifier) LST (lister), XRF (cross referencer). Submitted by Ray Van Tassle, Motorola.

No guarantees are made as to the completeness, usability, or quality of the programs on the tape and the material has not been checked or reviewed.

Media (Service Charge Code): 2400' Magnetic Tapes (PB) **Format:** VMS/BACKUP, 2400' Magnetic Tape (SB) **Format:** VMS/BACKUP, TK50 Tape Cartridge (TB) **Format:** VMS/BACKUP

DECUS No: V-SP-78 **Title:** AMIGA Utilities Collection 4 **Version:** 1, June 1988

Author: Various

Submitted by: Glenn C. Everhart

Operating System: AMIGA **Source Language:** BASIC, C, FORTRAN, PASCAL **Keywords:** Business Applications, Data Communications, Editors, Games, Mathematical, Utilities - VMS

Abstract: This tape contains a large collection of utilities and programs for the AMIGA 32 bit computer. The AMIGA is an inexpensive machine well suited to be used as a powerful graphics workstation in a Digital Equipment Corporation host environment, with multitasking, large address space, windows, graphics, color, and more.

The tape contains some new and improved VT220 emulators, editors, CAD programs, database software, games, picture processors, editors, spell correctors, raytracers, drivers, music players and scores, multiwindowing remote host packages, hard disk backup utilities, new fonts, appointment keepers, a BBS, CLI frontends, AMIGA hypertext system (a bit like the Apple Hypercard(tm)), animations, plotters, disk cataloguers, calculators, LISP interpreter, and much more.

This package contains items introduced for AMIGA PD consumption since AMIGA Utilities Collection 1 through AMIGA Utilities Collection 3 tapes became available. (See DECUS Part No.'s V-SP-68, V-SP-69, and V-SP-72). Numerous source programs make these programs valuable even on non-AMIGA computer configurations.

Because many of the files are in .ARC form, the VMSSWEEP utility is provided to allow for examination of these archives online on a VAX under VMS. An executable version of the ARC utility for VMS is also provided. However, since this is an alpha version of VMS ARC, it has several limitations which make it less able to read archives under VMS than VMSSWEEP. This is why both are provided.

Complete sources not included.

Media (Service Charge Code): 2400' Magnetic Tape (PC) **Format:** VMS/BACKUP, TK50 Tape Cartridge (TC) **Format:** VMS/BACKUP

DECUS No: VAX-348 **Title:** VTOUTPUT **Version:** 1 May 1988

Submitted by: Ronald William Burke, Westinghouse Electric Corporation, Baltimore, MD

Operating System: MicroVMS V4.X, VAX/VMS V4.X **Source Language:** DCL, VAX FORTRAN **Keywords:** Utilities - VMS

Abstract: VTOUTPUT - permits users to easily compose ASCII character strings and VT-1XX escape sequences for output to terminals, to files, and to DCL symbols.

CHARACTER - allows users to compose crude drawings using characters (or parts thereof) of the various character sets in a VT-1XX terminal.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS No: VAX-349 **Title:** SUM - Show Users More **Version:** 1.0, May 1988

Submitted by: Michael Kimura, Hughes Aircraft, Los Angeles, CA

Operating System: VAX/VMS V4.6, V4.7, V5.0 **Source Language:** VAX FORTRAN **Keywords:** Terminal Management

Abstract: SUM displays the terminal name (physical device name if terminal is a virtual terminal, remote node name and port name if terminal is a LAT device), username, and process identification code (PID) of either specific interactive users or all interactive users on the system.

Information about a particular user can be specified. If you specify a string, the system lists all interactive users whose usernames begin with the specified character string. For example, if the string MAR is specified as a username, a list of all interactive users whose usernames begin with MAR is displayed. If no user exists whose username matches the specified username string, an error message is displayed. If you omit this parameter, a list of all interactive users is displayed.

The command_qualifier /FULL (/NOFULL is default) requests that the full physical device name of a terminal and the full remote node name of a terminal server be displayed instead of the default which is to display only the first eight characters of the physical device name and the first six characters of the remote node name.

The command_qualifier /OUTPUT (/NOOUTPUT is default) controls where the output of the command is sent. If you do not enter the qualifier, or if you enter / OUTPUT without a file specification, the output is sent to the current process default output stream or device, identified by the logical name SYS$OUTPUT. If you enter /OUTPUT with a partial file specification (for example, specifying only a directory), SUM is the default file name and LIS the default file type. If you enter a file specification, it may not include any wildcard characters. If you enter /NOOUTPUT, output is suppressed.

Notes: Operating system VAX/VMS V4.6 or later is required.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS No: VAX-351 **Title:** QUOTE **Version:** April 1988

Submitted by: Tom Brink, Arizona Department of Transportation **Operating System:** VAX/VMS V4.6 **Source Language:** VAX FORTRAN **Memory Required:** 3KB **Keywords:** Utilities - VMS

Abstract: This program displays a random quote upon LOGON. Small efficient executable displays one of one hundred quotes available in ASCII data file. Extra one hundred quote data file included. User can create own data files.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

NEW LIBRARY PROGRAMS AVAILABLE FOR THE PDP-11 COMPUTER FAMILY

DECUS No: 11-SP-100 **Title:** Symposium Collection from the RSTS SIG, Spring 1987, Nashville and Fall 1987, Anaheim **Version:** May 1988

Author: RSTS SIG Tape Copy 1987

Operating System: RSTS/E V8 through V9 **Source Language:** BASIC-PLUS, BASIC-PLUS-2, C, MACRO-11, PASCAL **Memory Required:** 1KW - 32KW **Keywords:** Symposia Tapes - RSTS/E

Abstract: This is the RSTS SIG Tape from the Spring 1987 DECUS Symposium in Nashville and the Fall 1987 DECUS Symposium in Anaheim. Following are brief descriptions of the contents of the directories on the tape. The tape includes, but is not limited to these following items:

[87,0] A README.1ST + MT.LST directory, submitted by the RSTS SIG.

[87,1] Example .COM files, programs, etc. from 1987 Symposia sessions, submitted by Digital Equipment Corporation RSTS Development Team.

[87,2] PBS.SYS file reorganizer, idle user logoff program, simple MAIL system, program to make a mixed ANSI/DOS tape, and more, submitted by Erskine College.

[87,3] Utilities to improve system performance, copy tapes, transfer files via XMODEM, and more, submitted by Mike Mayfield, Northwest Digital Software.

[87,4] A (VAX-like) command line editor (CLE), submitted by Brian Nelson, University of Toledo.

[87,5] A new CLE, submitted by Brian Nelson, University of Toledo.

[87,6] KERMIT-11, submitted by Brian Nelson, University of Toledo.

[87,7] KERMIT for various Personal Computers, submitted by Brian Nelson, University of Toledo.

[87,8] PASCAL Library routines for RSTS, submitted by Ed Moran, Horace Mann School.

[87,9] TYPE - a 1K program plus a 1K runtime system that prints text files on one's terminal, submitted by James Wilkinson, The College of Charleston.

[87,10] DECUS C (with many bug fixes), submitted by Terry Kennedy, St. Peter's College.

[87,11 - A Fortune Cookie program submitted by Terry
87,13] Kennedy, St. Peter's College.

[87,14] Back issues of the RSTS newsletters, submitted by Terry Kennedy, St. Peter's College.

[87,15] System utilities, submitted by Terry Kennedy, St. Peter's College.

[87,16] DEC MAIL Utilities, submitted by Terry Kennedy, St. Peter's College.

[87,17] MS DOS KERMIT V2.30, submitted by Terry Kennedy, St. Peter's College.

[87,18] Games, submitted by Terry Kennedy, St. Peter's College.

[87,19] Directory of all past RSTS SIG tapes, submitted by Gene Alpern, Saber Computer Services.

No guarantees are made as to the completeness, usability, reliability, or quality of the programs on the tape. The programs have not been checked or reviewed.

Notes: Operating system RSTS/E V9 is required for some of the programs.

Media (Service Charge Code): 2400' Magnetic Tape (PS) **Format:** DOS-11, TK50 Tape Cartridge (TC) **Format:** DOS-11

DECUS No: 11-907 **Title:** CMD - IND Control Files **Version:** May 1988

Submitted by: R.W. Barnard, BIO/Comp Applications, Albuquerque, NM

Operating System: RT-11 V5.0 **Source Language:** IND **Keywords:** File Management

Abstract: CMD.IND provides the capability of issuing RT-11 commands such as COPY/ALPHABETICAL, or using the BACKUP command on a wildcarded file selection.

CSICMD is a similar concept, but it is primarily designed for doing wildcard file transfers with the TRANSF communications server. It permits issuing commands such as TRANSF/SINCE:9:MAY DU3:*.FOR/T VD3:/S.

Included are UCL-PLUS symbols for doing the transfers.

Notes: Other required software is either provided or identified in the package.

Media (Service Charge Code): One RX50 Diskette (JA) **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11

DECUS No: 11-908 **Title:** RTCLOK/TSXCLK **Version:** 1.16, 1.18, June 1988

Submitted by: James D. Whitlock, Jr., Woodcraft Components, San Antonio, TX

Operating System: RT-11 V5.4C **Source Language:** MACRO-11 **Memory Required:** 1KB **Hardware Required:** DL(V)11 type Serial line, GCW1000 Heath/Zenith. Most accurate clock with RS232 interface. **Keywords:** Clock

Abstract: RTCLOK automatically sets the date and time at boot-up and can reset the date and time as required. Since time is lost during the TSX boot process, the TSXCLK variant is also included. These programs allow the Heath/Zenith GCW1000 Most Accurate Clock to properly set the system date and time via DL(V)11 type serial interface lines.

Media (Service Charge Code): One RX01 Diskette (KA) **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11

DECUS No: 11-909 **Title:** DATUM: Setting System Date and Time **Version:** December 1987

Submitted by: Peter Lueders, D-4670 Lunen, West Germany

Operating System: RT-11 V5.01 **Source Language:** FORTRAN IV **Memory Required:** 15KB **Software Required:** DECUS No. 11-495 **Hardware Required:** VT52 or VT100 terminal **Keywords:** Utilities -RT-11

Abstract: DATUM is a date and time routine specifically for regular users of RT-11 operating systems. It shows after system start the next calendar day and the stored time of the last system start. After correcting this value the program sets the system's date and time. The package includes two versions for VT52 and VT100 terminals.

Notes: Operating system RT-11 V4 or V5 is required. DECUS No. 11-495, "TTLIB: VT100 Library Routines" is required.

Media (Service Charge Code): One RX01 Diskette (KA) **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11

NEW LIBRARY PROGRAMS AVAILABLE FOR THE RAINBOW FAMILY OF COMPUTERS

DECUS No: RB-131 **Title:** JOBSDUMP/JOBSDISP **Version:** 4.1, April 1988

Submitted by: James A. O'Brien, Ph.D.

Operating System: CP/M V2.00, MS/DOS V2.11 **Source Language:** PASCAL **Memory Required:** 192KB **Hardware Required:** Rainbow 100 A/B/PLUS; Rainbow Graphics option; Digital Equipment Corporation LA series or EPSON compatible printer. (Optional). **Keywords:** Graphics

Abstract: JOBSDUMP is a utility which dumps the contents of graphics memory on a Digital Equipment Corporation Rainbow computer in either Digital Equipment Corporation sixel format or Epson graphics format directly to a printer or to a disk file. (The difference between the CP/M and MS/DOS versions is that the latter allows the setting of an environment variable to select which colors (0-3 in high resolution, 0-F in medium resolution) should be printed). Both command-line and menu-driven operation modes are provided. Graphic images can be in either of two sizes, one a dot-for-dot image of the screen and the other designed to fill most of a printer page. Images can be printed as negatives, to save printer ribbons. JOBSDUMP can be run from within GW-BASIC, providing a simple graphics printing capability for the latter. See the documentation file on the disk for further information.

JOBSDISP is a utility which displays, until a key is pressed, the last graphic image contained in the graphics option memory of a Digital Equipment Corporation Rainbow computer. It is useful for previewing images to be printed by JOBSDUMP (although the latter also has a preview option).

Restrictions: Not for commercial use.

Sources not included.

Media (Service Charge Code): One RX50 Diskette (JA) **Format:** MS/DOS

NEW ULTRIX PROGRAMS

DECUS No: UX-113 **Title:** Creep **Version:** 1.5, May 1988

Submitted by: University of California at Berkeley, through Digital Equipment Corp.

Operating System: ULTRIX V4.3BSD, UNIX V4.2 **Source Language:** C **Memory Required:** 1.5MB **Keywords:** Circuit Simulation

Abstract: CREEP is a two-dimensional (2D) process simulator designed to solve certain creep-flow problems encountered in integrated circuit fabrication technology. Its most important capability is the prediction of general 2D silicon oxidation, using stress-dependent oxidation models. It also performs glass-reflow or film-shrinkage simulation as subset-problems of silicon oxidation.

CREEP is equipped with a flexible and robust data structure for handling geometric information. All geometric structures are represented using nodes and segments; hence, fairly general geometric structures can be handled by the CREEP program. A finite-element mesh generator has also been built into CREEP. Mesh generation is done automatically at every time-step of the computation, with the user supplying only a mesh-density parameter.

Ordering information for the documentation is as follows:

- Order DECUS No. UX-113 (EB) for the "CREEP - A 2D Creep-Flow Process Simulator" User's Guide.
- Order DECUS No. UX-113 (ED) for the "Finite-Eement Methods for Process Simulation Application to Silicon Oxidation" Manual.

Notes: This program was developed by the Computer-Aided Design Group, Department of Electrical Engineering and Computer Sciences, University of California-Berkeley.

Restrictions: U.S. Government export regulations prohibit the distribution of this program outside the United States without the appropriate export licenses. Documentation available in hardcopy only.

Media (Service Charge Code): User's Manual (EB), User's Manual (ED), 600' Magnetic Tape (MA) **Format:** TAR

REVISIONS TO LIBRARY PROGRAMS

DECUS No: V-SP-53 **Title:** KERMIT Distribution **Version:** June 1988

Author: Various

Submitted by: Glenn C. Everhart, Ph.D.

Operating System: CP/M, IAS, MICRO/RSX, MS/DOS, Micro/VMS, OS/278, P/OS, RSTS/E, RSX-11M, RSX-11M-PLUS, RT-11, TOPS-10, TOPS-20, VAX/VMS, VENIX **Source Language:** ALGOL, BASIC, BLISS-16, BLISS-32, BLISS-36, C, FORTRAN 77, FORTRAN IV, FORTRAN IV-PLUS, LISP, MACRO-10, MACRO-11, MACRO-32, PAL-8, PASCAL, VAX-11 FORTRAN **Keywords:** Data Communications, KERMIT

Abstract: This tape collection contains a VMS Backup distribution made from a KERMIT distribution from Columbia University dated June 24, 1988. The tape collection contains all KERMITS known to Columbia as of that date plus a large amount of documentation.

The Columbia distribution is on five (5) reels of tape. To reduce costs, the distribution has been placed on TWO (2) reels of tape for DECUS, in VMS/BACKUP format at 1600 BPI, one tape at 6250 BPI. All KERMITS are here as distributed by Columbia University. The new MS/DOS KERMIT (V2.30), a new universal IBM Mainframe KERMIT, and an update to C KERMIT are present on this collection as recent additions. Complete KERMIT documentation and booting instructions are on the tape. No paper documentation is needed. Files beginning with AA*.* should be looked at first for an overview of what's here.

Changes and Improvements: Later versions of several Kermits.

MEDIA (Service Charge Code): 2400' Magnetic Tapes (PB) **Format:** VMS/BACKUP, 2400' Magnetic Tape (SB) **Format:** VMS/BACKUP, TK50 Tape Cartridge (TB) **Format:** VMS/BACKUP

DECUS No: VAX-129Title: FORTRAN Programming ToolsVersion: III.4, June 1988 Submitted by: A. Ragosta & L. Jurgeleit, US Army ARTA, MS219-3, Moffett Field, CA

Operating System: VAX/VMS V4.7 **Source Language:** DCL, FORTRAN 77, MACRO-32 **Memory Required:** Varies **Keywords:** Debugging, System Management - VMS, Tools - Software Development

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, a general purpose filter, a user-friendly executive subsystem, 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.

Notes: Operating system VMS V4.0 or later is required.

Changes and Improvements: Miscellaneous bug fixes and enhancements, new program to send messages, new executive subsystem.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP, or order VAX-LIB-4

DECUS No: VAX-300 **Title:** JMU Bulletin Board **Version:** 2.15, June 1988

Submitted by: Michael S. O'Neill, James Madison University, Harrisonburg, VA

Operating System: VAX/VMS V4.7 **Source Language:** VAX FORTRAN **Software Required:** FMS **Hardware Required:** VT100 compatible terminal or PC with VT emulation. Keywords: Bulletin Board

Abstract: The JMU Bulletin Board/Conferencing System is a FMS based menu driven system that utilizes the return and cursor keys for command selection. It is designed to allow novice users to easily use it for viewing notices without forcing them to become familiar with its advanced features.

Among its features are:

- Tracking of last notice read in each category.
- A menu driven user interface.
- Integral access to the EDT text editor.
- Context sensitive HELP system.
- Selective category omission on a per user basis.
- Automatic insertion of notice owner's userid.
- Direct access to the VMS mail utility while viewing a notice.
- A reply option for posting a response to a notice while it is being viewed.
- A backup option that allows the viewing of previously viewed notices.
- An output option that allows you to output a copy of a notice to a file, line printer (SYS$PRINT), or a printer connected to your terminal or PC.
- Support for multiple bulletin boards.
- Chaining of notice replies.
- Multi-level conferencing support.
- File upload and download support.

Notes: Operating System VAX/VMS V4.4 or higher is required.

Changes and Improvements: Menu enhancements, conference enhancements, bug fixes.

Media (Service Charge Code): 600' Magnetic Tape (MA) **Format:** VMS/BACKUP, or order VAX-LIB-8

DECUS No: VAX-318 **Title:** Micro-FLX **Version:** 1.3, June 1988

Submitted by: Trevor Taylor, Microcomputer Technology, Aspley, QLD, Australia 4034

Operating System: MicroVMS V4.4 - 4.6, VAX/VMS V4.6 **Source Language:** C, VAX FORTRANHardware Required: RX33 or RX50 Floppy Diskette Drive **Keywords:** Data Communications, Utilities - VMS

Abstract: Micro-FLX is a file transfer program designed to allow VAX users to read and write CP/M and MS/DOS floppies.

It handles RX50 diskettes in either CP/M or MS/DOS format from a Rainbow or RX33 diskettes in MS/DOS format from a VAXmate. Floppies can be mounted in an appropriate disk drive on either a VAX or a MicroVAX, and files can then be copied to and from them using commands similar to DCL. There is also built-in help.

Release notes are distributed with each order.

Changes and Improvements: Bug fixes and new "initialize" command.

Sources not included.

Media (Service Charge Code): One RX50 Diskette (JA) **Format:** VMS/BACKUP, 600' Magnetic Tape (MA) **Format:** VMS/BACKUP

DECUS No: 11-490 **Title:** TSXLIB: A FORTRAN Callable Library Implementation of EMTs for TSX-PLUS **Version:** 6.31/88e30a

Submitted by: N. A. Bourgeois, Jr., NAB Software Services, Inc., Albuquerque, NM

Operating System: RT-11 V5.4, TSX-PLUS V6.31 **Source Language:** MACRO-11 **Software Required:** FORTRAN for application programs **Hardware Required:** MMU to support TSX-PLUSKeywords: FORTRAN, Libraries - RT-11 Abstract: TSXLIB is a library of FORTRAN callable routines that implement the TSX-PLUS system services which are unique to TSX-PLUS. The library has been updated to include all TSX-PLUS unique services through TSX-PLUS V6.31.

Like RT-11, TSX-PLUS offers the MACRO-11 programmer a number of system services. These services are implemented via both the RT-11 programmed requests (for those services common to both RT-11 and TSX-PLUS) and raw EMT instructions (for those unique to

TSX-PLUS). RT-11 makes its system services available to the FORTRAN programmer through the system subroutine library, SYSLIB. TSX-PLUS also honors the bulk of the service requests in the SYSLIB routines. TSXLIB, however, makes the TSX-PLUS unique EMTs available to the FORTRAN programmer.

These TSX-PLUS library routines provide facilities to support communication lines, detached jobs, device allocating and deallocating, file structured device mounting and dismounting, communication between running programs, job privileges control, job status monitoring, program performance analysis, real time program execution, shared runtime systems, shared files, special files information, spooler control, subprocess control, system status information, communication between running programs and a terminal, program control of the terminal, ODT activation mode, user name control, windowing, and several miscellaneous EMTs.

The TSXLIB distribution kit includes the MACRO-11 source modules for all the routines, a user's manual in machine readable form, an indirect command file to build the library, and the implemented library. The FORTRAN sources for some test programs are also included. These test programs have been executed with both FORTRAN 66 and FORTRAN 77.

Changes and Improvements: New routines for TSX-PLUS V6.31.

Media (Service Charge Code): Two RX50 Diskettes (JB) **Format:** RT-11, One RX02 Diskette (LA) **Format:** RT-11, 600' Magnetic Tape (MA) **Format:** RT-11, TK50 Tape Cartridge (TB) **Format:** RT-11

STEERING COMMITTEE LISTS

ARTIFICIAL INTELLIGENCE SIG

CHAIR
Cheryl Jalbert
JCC
128 West Broadway
Granville, OH 43023
(614) 587-0157

VICE-CHAIR
OPS5 WORKING GROUP CHAIR
Don Rosenthal
Space Telescope Science Inst.
Homewood Campus
Baltimore, MD 21218
(301) 338-4844

NEWSLETTER TASK FORCE CHAIR
ADMINISTRATIVE ASSISTANCE
Becky Wise
Amdalh CSD
2200 North Greenville Ave.
Richardson, TX 75081
(214) 699-9500 x 272

NEWSLETTER EDITOR
Terry Shannon
Computer Info. Sys., Inc.
Technical Consultant
165 Bay State Drive
Braintree, MA 02184
(617) 848-7515

SYMPOSIA COORDINATOR
Pam Vavra
Hughes Aircraft EDSG
P.O. Box 902 E52/D220
El Segundo, CA 90245-0902
(213) 616-7071

MEMBERSHIP COORDINATOR
SUITE COORDINATOR
Chris Goddard
Simpact Associates
9210 Skypark Court
San Diego, CA 92123
(619) 565-1865

SESSION NOTE EDITOR
George Humfeld
Naval Sea Systems Command
PMS 350 ED Dept of the Navy
Washington, DC 20362-5101
(202) 692-0137

ASS'T SESSION NOTES EDITOR
David Frydenlund

STORE REPRESENTATIVE
Sally Townsend
Inst. Defense Analysis
1801 N. Beauregard St.
Alexandria, VA 22311
(703) 845-2122

PUBLIC DOMAIN SOFTWARE TF CHAIR
LIBRARY REPRESENTATIVE
Jim Sims
Space Telescope Science Ins.
3700 San Martin Drive
Baltimore, MD 21218
(301) 338-4949

AI LUG COORDINATOR
ASSISTANT STORE REP.
Dennis Clark
RT2 Box 264
Kingston, TN 37763
(615) 576-7384

REPORTER TO THE UPDATE.DAILY
Bill Lennon

SEMINAR UNIT REP.
CAMPGROUND COORDINATOR
Leona Fluck
Educational Testing Service
Rosedale Road
Princeton, NJ 08540
(609) 734-1243

DEC COUNTERPART
Art Beane
Hudson, MA

MEMBERS-AT-LARGE
David Slater
George Winkler
Jeff Fox
John Williamson
Wayne Graves
Matt Mathews
Dave Campbell
Shirley Bockstahler-Brandt
Barry Breen
Tom Viana

BUSINESS APPLICATIONS SIG
SOLUTIONS
DECUS

BUSINESS APPLICATIONS SIG

CHAIRMAN
George Dyer
Gallaudet University
800 Florida Ave, NE-EMG Bldg
Washington, DC 20002
(202) 651-5300

COMMUNICATIONS COORDINATOR
Steve Lacativa
Price Waterhouse
153 East 53rd Street
New York, NY 10022
(212) 371-2000 x 3107

SYMPOSIA COORDINATOR
Mark Hults
USSA Administrative Systems
USSA Bldg. B01E
San Antonio, TX 78288
(512) 498-8725

LUG COORDINATOR
Patrick LeSesne
U.S. Coast Guard
Room 1416E 2100 2nd St. SW
Washington, DC 20593
(202) 267-0354

MARKETING COORDINATOR
Tom Byrne
L. Karp & Sons
1301 Estes
Elk Grove Village, IL 60007
(312) 593-5706

PROGRAM PLANNING COORDINATOR
Stuart Lewis
Douglas Furniture Corp.
P.O. Box 97
Bedford Park, IL 60499
(312) 458-1505

SEMINARS COORDINATOR
Daniel Esbensen
Touch Technologies, Inc.
9990 Mesa Rm , Rd. #220
San Diego, CA 92121
(619) 455-7404

LRP COORDINATOR
Arnold I. Epstein
D-M Computer Consultants
Rolling Meadows, IL 60008
(312) 394-8889

NEWSLETTER EDITOR
Dave Levenberg
Credit Suisse
Dept OA1 15th floor
100 Wall Street
New York, NY 10005
(212) 612-8372

SESSION NOTE EDITOR
Marty Schmitt
Harris Publishing
3 Barker Avenue
White Plains, NY 10601
(914) 946-7500 x 287

LIBRARY REPRESENTATIVE
David Hittner
Projects Unlimited
3680 Wyse Road
Dayton, OH 45414
(513) 890-1800

CL SIG LIAISON
Becky Burkes-Ham

DMS SIG LIAISON Joe Sciuto **MEMBERS-AT-LARGE**
Robert D. Lazenby
Dixie Beer Dist., Inc.
Louisville, KY

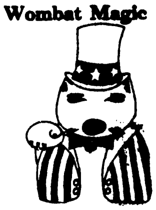
Robert Kayne
Gallaudet College
Washington, DC

Ray Evanson
Paragon Data Systems
Winona, MN

DEC COUNTERPARTS
Sue Yarger
Digital Equipment Corporation
Merrimack, NH 03054-0430

Paula Daley
Digital Equipment Corporation
Merrimack, NH 03054-0430

Pam Kukla
Digital Equipment Corporation
Maynard, MA 01754



DATATRIEVE/4GL SIG

CHAIRMAN
Donald E. Stern Jr.
Warner Lambert Company
10 Webster Road
Milford, CT 06460
(203) 783-0238

SYMPOSIA COORDINATOR
Lisa M. Pratt
Vitro Corporation
Nuwes Code 3144
Keyport, WA 98345
(206) 396-2501

ASS'T SYMPOSIA REPRESENTATIVES
T.C. Wool
E.I. duPont DeNemours & Co.
Engineering Dept.
P.O. BOX 6090.
Newark, DE 19714-6090
(302) 366-4610

Janet G. Banks
Weyerhaeuser Info. Sys.
Mail Stop CCB-2E
Tacoma, WA 98477
(206) 924-4082

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR

Joe H. Gallagher
Research Medical Center
2316 East Meyer Blvd.
Kansas City, MO 64132
(816) 276-4235

COMMUNICATION REPRESENTATIVE
PRODUCTION EDITOR

Steve Cordiviola
Kentucky Geological Survey
228 Mining & Mineral Resources Bldg.
Lexington, KY 40506-0107
(606) 257-5863

ASSOCIATE NEWSLETTER EDITOR

Pasquale (Pat) F. Scopelliti
Corning Glass Works
Mail Stop MP-RO-01-1
Corning, New York 14831
(607) 974-4496

Lorey B. Kimmel
Digital Equipment Corp.
6707 Whitestone Road
Baltimore, MD 21207
(301) 298-1500

Herbert G. Reines
Reznick Feddder & Silverman
4520 East West Highway
Suite 300
Bethesda, MD 20814
(301) 652-9100

Alan Winston
Stanford Synchrotron Radiation Lab.
SLAL BIN 69
P.O. Box 4349
Stanford, CA 94305
(415) 854-3300 x2874

VOLUNTEER COORDINATOR

Susan Krentz
NKF Engineering
12200 Sunrise Valey Dr.
Reston, VA 22091
(703) 620-0900

ASSISTANT VOLUNTEER COORD.

Harry Miller
City of Ontario Police
200 N. Cherry Avenue
Ontario, CA 91764
(714) 988-6481

SEMINARS COMMITTEE REP.

Dana Schwartz
15719 Millbrook Lane
Laurel, MD 20707
(301) 859-6277

SESSION NOTES EDITOR

Bernadette Reynolds
City of Ontario Police
200 N. Cherry Avenue
Ontario, CA 91764
(714) 988-6481

SUITE COORDINATOR

Bert Roseberry
Commandant (G-APA-1)
2100 2nd Street, S.W.
Washington, DC 20593-0001
(202) 267-2629

FEATURE EDITOR

Philip A. Naecker
Consulting Engineer
3011 N. Mount Curve Ave.
Altadena, CA 91001
(818) 791-0945

DEC COUNTERPARTS

John Carpenter
110 Spit Brook Road
2K02-2/Q21
Nashua, NH 03062
(605) 881-0043

ARTIST & LIBRARY REP.

Bart Z. Lederman
WU World Communications
67 Broad Street (28th Floor)
New York, NY 10004
(212) 607-2657

RALLY WORKING GROUP CHAIR

Steven G. Fredrickson
Fredrickson Consulting Service
107 1st Avenue N.
Seattle, WA 98109
(206)283-0273

POWERHOUSE W/G CHAIR

David Nagurney
TSO Financial Corp.
Five TSO Financial Center
Three Hundred Welsh Road
Horsham, PA 19044-2009
(215) 657-4000

DMS SIG LIAISON

William Tabor
W.I. Tabor, Inc.
12018 Royal Palm Blvd.
Coral Springs, FL 33065
(305) 755-7895

SMARTSTAR WORKING GROUP CHAIR

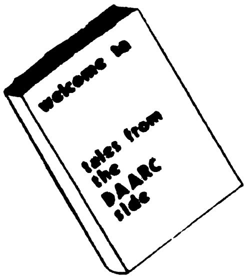
Thomas Colati
Time Enterprises
301 North Harrison
Suite 101
Princeton, NJ 08540
(800) 548-6865

ACCENT-R USER GROUP LIAISON

Winston Tellis
Fairfield University
North Benson Road
Fairfield, CT 06430
(203) 255-5411

FOCUS WORKING GROUP CHAIR

Les Hulse
The Gillette Company
Prudential Tower Bldg.
Boston, MA 02199
(617) 421-7910



DAARC SIG

CHAIRMAN

James Deck
Inland Steel Research Lab.
3001 East Columbus Drive
East Chicago, IL 46312
(219) 392-5613

SYMPOSIA COORDINATOR

Mack Overton
FDA
Chicago, IL

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR

Dale Hutchison
Cummins Engine Co.
4720 Baker St., Ext.
Lakewood, NY 14750
(716) 456-2191

DEC COUNTERPART

Bill Forbes
Marlboro, MA

HARDWARE & INTERFACING

Peter Clout
Los Alamos National Lab
Los Alamos, NM

MATH STATISTICS & ANALYSIS

Herbert J. Gould
C.C.F.A. Univ. of Ill. Medical Ctr.
Chicago, IL

PROCESS CONTROL-INDUSTRIAL AUTOMATION

Bill Tippie
Kinetic Systems Corp.
Lockport, IL

RS-1

George Winkler
CPC International
Argo, IL

DATA MANAGEMENT SYSTEMS SIG

CHAIRMAN

Doug Dickey
GTE Government Systems
1700 Research Blvd.
Rockville, MD 20850
(301) 294-8462

MEMBER AT LARGE

Alan Schultz
Southwestern Bell Yellow Pages
12800 Publication Dr., Suite 108
St. Louis, MO 63131
(314) 957-2029

SYMPOSIA COORDINATOR
SQL Standards Rep.

Keith Hare
JCC
P.O. Box 463
Granville, OH 43023
(614) 587-0157

COMMUNICATIONS REP.

Debbie Kennedy Coleman
Shane Co.
2 W Washington St., Suite 600
Indianapolis, IN 46204
(317) 635-9100

NEWSLETTER EDITOR

William Packard
Mass Mutual Life Ins.
1296 State Street B-391
Springfield, MA)1111
(413) 788-8411

SESSION NOTE EDITOR

Mark Morgan
Farm Credit Banks
P.O. Box 141
Springfield, MA 01102
(413) 786-7600

MEMBERSHIP COORDINATOR
MEMBER AT LARGE

Rocky Hayden
Userware International
2235 Meyer Avenue
Escondido, CA
(619) 745-6006

MEMBER AT LARGE
PAST SIG CHAIRMAN

Steve Pacheco
Ship Analytics
North Stonington, CT 06359
(203) 535-3092

PAST SIG CHAIR
MEMBER AT LARGE

Joe Sciuto
U.S. Army
5910 Westchester Street
Alexandria, VA 22310
(202) 692-6903

SEMINAR REP.

Stephen Gomez
Signal Technology
1700 Montgomery St.
San Francisco, CA 94111
(415) 954-8532

WORKING GROUP COORDINATOR/

Jim Perkins
PSC, Inc.
20 Kimball Ave., Suite 305
Shelburne, VT 05401
(802) 863-8825

CAMPGROUND COORDINATOR

Rosemary O'Mahony
Arthur Anderson & co.
33 West Monroe Street
Chicago, IL 60603
(312) 507-6510

SESSION CHAIR COORDINATOR

Andy Menezes
AD & E
29-B Montvale Avenue
Woburn, MA 01801
(617) 938-1979

Rdb WORKING GROUP Coordinator

Howard Cheng
Bechtel Western Power Corp.
12440 East Imperial Highway
Norwalk, CA 90650
(213) 807-4077

ROADMAP COORDINATOR

Elizabeth Irving
Dupont
P.O. Box 1089
Orange, Texas 77630-1089
(409) 886-9427

DBMS COORDINATORS

Bryan Holland
1006 Pleasant St., #20
Weymouth, MA 02189
(617) 335-7573

Paul E. Reese
Aetna, Systems Dept
Financial Division
CityPlace
Hartford, Connecticut 06156
(203) 275-2012

MEMBER AT LARGE

Jim Redfield
BDM
9020 N. Cap. of Texas Highway
Austin, TX 78759
(512) 346-6760

STORE REPRESENTATIVE
FIMS STANDARDS REP.

Paul W. Plum, Jr
Lukens Steel Company
Coatesville, PA 19320
(215) 383-2024

RMS WORKING GROUP COORDINATOR

Allen Jay Bennett
Logisticon, Inc.
5035 Whitneyville Road
Ada, MI 49301
(616) 452-1823

DEC COUNTERPART

Wendy Herman
John Wood
Nashua, NH

EDUSIG

CHAIRMAN

Robert A.Shive, Jr.
Associate Dean of the College
Millsaps College
Jackson, MS 39210-0001
(601) 354-5201

VICE CHAIR AND UNIV. COORD.

Ardoth A. Hassler
Assistant Director
for Academic Services
Computer Center
Catholic University of America
Washington, DC 20064
(202) 635-5373
HASSLER@CUA.BITNET

SYMPOSIUM COORDINATOR

Mary Jac Reed
VAX Project Manager
Office of Instructional Tech.
University of Delaware
Newark, DE 19716
(302) 451-8161

COMMUNICATIONS COORDINATOR

Paula Barnes
Computer Center Operations Manager
North Carolina School of Science & Math.
1219 Broad Street
P.O. Box 2418
Durham NC 27705
(919) 286-3366
plb@ecsvax.bitnet

SEMINARS COORDINATOR

Donald C. Fuhr
Director of Computer Services
Tuskegee University
Tuskegee, AL 36088
(205) 727-8242

NEWSLETTER EDITOR

Fred Bell
Professor of Computer Science and
Coordinator of Computer Based Education
Taft College
29 Emmons Park Drive
P.O. Box 1437
Taft, CA 93268
(805) 763-4282

SESSION NOTES EDITOR

Jim Gerland
SUNY/Buffalo
Academic Computing, Univ. Computing Serv.
Computer Center
Buffalo, NY 14260
(716) 636-3557
gerland@ubvms.bitnet
gerland@ubvmsc.cc.buffalo.edu

ADMINISTRATIVE APPLICATIONS COORD.

David Cothrun
President
Taft College
29 Emmons Park Drive
P.O. Box 1437
Taft, CA 93268
(805) 763-4282

DIGITAL COUNTERPART

C. Michael Greene, Jr.
Networking Consultant
Education Industry Marketing Group
Digital Equipment Corporation
Three Results Way, MR03-2/E7, Box 1003
Marlboro, MA 01752-9103
(508) 467-2149
GREENE%SANTEE.DEC@DECWRL
GREENE%SANTEE.DEC@DECWRL.DEC.COM
GREENE@SANTEE.DEC.COM

GRAPHICS APPLICATIONS SIG

CHAIRMAN

Bijoy Misra
Smithsonian Institution
60 Gordon St., MS39
Cambridge, MA 02138
(617) 495-7392

SESSION NOTE EDITOR

Carol Schwob
Florida Atlantic Univ.
Bldg. 22, Room 106
Boca Raton, FL 33431
(305) 393-2640

NEWSLETTER EDITOR (acting)

OPEN

ASSOCIATE NEWSLETTER EDITOR

Charles D. Carter
Huntington Alloys, Inc.
Technology Dept.
P.O. Box 1958
Huntington, WV 25720
(304) 526-5721

WORKSTATION WORKING GROUP COORD.

Bob McCormick
Video Communications, Inc.
1325 Springfield Street
Feeding Hills, MA 01030
(413) 786-7955

ENGINEERING GRAPHICS WORKING GROUP COORD.

Eric Rehm
Gonzaga University
SPOCAD
E 502 Boone
Spokane, WA 99258
(509) 484-6814

COMMUNICATIONS REP
NEWSLETTER EDITOR

Robert Hays
KMS Fusion
3621 So. State Road
Box 1567
Ann Arbor, MI 48106

LIBRARY COORDINATOR

Mike McPherson
Michigan University
269 Engineering Bldg.
East Lansing, MI 48824
(517) 353-9769

STANDARDS COORDINATOR

OPEN

VOLUNTEER COORDINATOR

Dick McCurdy
University of Florida
Room 216, Larsen Hall
Gainsville, FL 32611
(904) 392-4915

LIBRARY COMMITTEE

James M. Turner
Saber Technology
San Jose, CA

DEC COUNTERPART

Jim Flatten
Spit Brook, NH

Rick Landau
Marlboro, MA

INFORMATION OFFICER

Mike York
Boeing Computer Services
P.O. Box 24346 M/S 03-73
Seattle, WA 98124
(206) 342-1442

SYMPOSIUM COORDINATOR

Dottie Elliott
Northrop Services Inc.
P.O. Box 12313
Research Triangle PK, NC 27709
(919) 541-1300

DATA DISPLAY WORKING GROUP COORD.

Joy Williams
Eaton Corp.
P.O. Box 766
Southfield, MI 48037

HARD NEWS

HARDWARE MICRO SIG

CHAIRMAN
Willian K. Walker
Monsanto Research Corp.
Miamisburg, OH

PRODUCT PLANNING COORDINATOR
George Hamma
Synergistic Technology
Cupertino, CA

PRE-SYMPOSIUM SEMINAR COORDINATOR
James R. Lindesmith
Monsanto Research Corp.
Miamisburg, OH

COMMUNICATIONS COORDINATOR
John G. Hayes
Information Systems
South Central Bell
Birmingham, AL

NEWSLETTER EDITOR
Carmen D. Wiseman
Digital Review
Boston, MA

SESSION NOTE EDITOR
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

TOEM (CHIPS & BOARDS)
Jack J. Peterson
Horizon Data Systems
Richmond, VA

HHK (HARDWARE HINTS & KINKS)
Wayne Kesling
Monsanto Research Cor.
Miamisburg. OH

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

PERFORMANCE MEASUREMENT COORD.
William Wallace
600 W. Washington Street
Peoria, IL

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

UNISIG LIAISON
Jim Livingston
1 Results Way
Cupertino, CA

SITE SIG LIAISON
Emily Kitchen
A.H. Robins 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
16533 Evergreen
Southfield, MI

DEC COUNTERPARTS TERMINALS
Nina Abramson
Maynard, MA

TOEM (Chips & Boards)
Art Bigler
Marlboro, MA

DIAGNOSTIC
George D. Cooke
Maynard, MA

STORAGE
Marilyn Fedele
Maynard, MA

MSD (Micro Systems Develp.)
Roy Rodgers
Maynard, MA

PRINTER PRODUCTS
Frank Orlando
Maynard, MA

DECUS EUROPE LIAISON
Hans Zoller

IAS SIG

Alan Frisbie
Flying Disk Systems, Inc.
4759 Round Top Drive
Los Angeles, CA 90065
(213) 256-2575

NEWSLETTER EDITOR
Frank R. Borger
Radiation Therapy
Michael Reese Medical Center
Lake Shore Drive @ 31st St.
Chicago, IL 60616
(312) 791-2515

WHIMS COORDINATOR
Kathleen Anderson
Eaton Information Management
System Division
Hampton, VA
(804) 326-1941

RSX LIAISON
Ray French
Boeing Computer Services
Seattle, WA
(206) 655-6228

MEMBER-AT-LARGE
Doug Reno
Abbot Laboratories
North Chicago, IL
(312) 937-7504

PAST CHAIRMAN
Mike Robitaille
Grumman - CTEC, Inc.
6862 Elm Street
McLean VA 22101
(703) 556-7400 x431

CHAIRMAN EMERITUS
Bob Curley
Division of Medical Physics
University of Pennsylvania
Philadelphia, PA
(215) 662-3083

SYMPOSIA COORDINATOR
Lynda L. Roenicke
Special Systems Branch
Naval Ocean Systems Center
271 Cataline Blvd Code 742
San Diego, CA
(619) 225-7569

LIBRARY COORDINATOR
Ted Smith
The University of PA
Philadelphia, PA 19101
(215) 662-3500

MEMBER-AT-LARGE
Kerry Wyckoff
Salt Lake City, UT

DEC COUNTERPART
Nancyfaye Autenzio
Stow, MA
(617) 496-9606

Leverage
Languages & Tools

LANGUAGES AND TOOLS SIG

CHAIRMAN
Sam Whidden
American Mathematical Society
201 Charles St.
P.O. Box 6248
Providence, RI 02940
(401) 272-9500

VICE CHAIR
SYMPOSIA COORDINATOR
Earl Cory
Eaton Corporation
31717 La Tienda Dr.
Westlake Village, CA 91359
(818) 706-5385

STORE REPRESENTATIVE
Chair, TECH. PROD OF DOC. W/G
Howard Holcombe
RCA
Front & Cooper Sts.
Camden, NJ 08055
(609) 338-4946

NEWSLETTER EDITOR
ALT. CommComm REP.
Al Folsom, Jr.
Fischer & Porter Co.
E. County Line Rd.
Warminster, PA 18974
(215) 674-7154

SESSION NOTES EDITOR
Mark Katz
GTE Government Systems
100 First Avenue
Waltham, MA 02154
(617) 466-3437

AUSTRALIAN L&T INTERFACE
Gordon Brimble
Bldg. 180 Labs Area
Defence Research Centre
Box 2151 GPO
Adelaide, S.A. Australia 5001
(61)(8)259-6119

INTERSIG COORDINATOR
Dorothy Geiger
Wollongong Logistics Group
49 Showers Drive #451
Mountain View, CA 94040
(415) 948-1003
(415) 962-7160

EUROPEAN METHODS
L&T INTERFACE
Bernd Gliss
Max-Planck-Institute
Heisenbergstra Be 1
7000 Stuttgart 80, W. Germany
(711) 686-0251

AST CHAIR
RODUCTIVITY TOOLS COORDINATOR
Kathy Hornbach
Digital Equipment Corporation
110 Spit Brook Rd., ZK03-3/Y25
Nashua, NH 03062

HAIR TECO WORKING GROUP
Mark J. Hyde
Advanced Computing Services
209 Ardsley Drive
DeWitt, NY 13214
(315) 446-7223

EMBER, ANSI BASIC X3J2 STDS, COMM.
TANDARDS COORD.
DP-11 REP.
HAIR, PDP-11 LAYERED PRODUCT W/G
CTING SYMPOSIUM COORDINATOR
Stephen C. Jackson
SCJ Consulting, Inc.
7260 University Avenue N.E.
Suite 105
Minneapolis, MN 55432
(612) 571-8430

HAIR, CONFIG. MGMT. WORKING GROUP
IG SECRETARY
Mark Alan Kidwell
Texas Instruments Inc.
P.O. Box 801 M/S 8006
McKinney, TX 75069
(214) 952-2058

EVEL. COUNTERPART, PDP-11 SOFTWARE
Joe Mulvey
Digital Equipment Corp. ,ZK01-3/J10
110 Spit Brook Road
Nashua, NH 03062-2642
(603) 881-1218

ISP/AI COORDINATOR
Don Rosenthal
Space Telescope Science Institute
Homewood Campus
Baltimore, MD 21218
(301) 338-4844

IBRARY REPRESENTATIVE
IG TAPE LIBRARIAN
HAIR, PUBLIC DOMAIN SOFTWARE W/G
Tony Scandora
Argonne National Laboratory
CMT 205
Argonne, IL 60439
(312) 972-7541

HAIR, C WORKING GROUP
James Maves
Eaton Corp, IMSD
31717 La Tienda Drive
Box 5009
Westlake Village, CA 91359
(818) 706-5395

TEERING COMMITTEE MEMBER-AT-LARGE
Jay Wiley
Bechtel Power Corp
12400 East Imperial Highway
Norwalk, CA 90650
(213) 807-4016

ESSION CHAIRS COORDINATOR
OF CHAIRS COORDINATOR
ESSIONS QUALITY COORD.
LT. SYMPOSIUM COORD.
Gary C. Lelvis
IMSL
2500 Park West Tower One
2500 City West Blvd.
Houston, TX 77042-3020
(713) 782-6060

CHAIR, FORTRAN WORKING GROUP
Scott Krusemark
Systemation, Inc.
8473 Daisywood Ave N.W.
North Canton, OH 44720
(216) 499-6251

CHAIR, LOW LEVEL LANGUAGES W/G
Gerald Lester
Computerized Processes Unlim.
4200 South 1-10 Service Road
Suite #205
Metairie, LA 70001
(504) 889-2784

DEVEL. COUNTERPART, COMM. LANG.
Jim Totton
Digital Equipment Corp.
ZK02-3/K06
110 Spit Brook Road
Nashua, NH 03062

ALT. ANSI X3J9 PASCAL STDS. COMM.
Phil Wirth
E-Systems, Garland Division
Box 660023, MS 53730
Dallas, TX 75266-0023
(214) 272-0515 x4319

ALT. ANSI X3J4 COBOL STDS. COMM.
Dale Marriott
El Paso County Office Bldg.
27 E. Vermijo Street
Colorado Springs, CO 80903
(303) 520-6435

ACTING CHAIR, DIBOL WORKING GROUP
ASSOC, W/G COORD. UNSCHEDULED TOPICS
CHAIR, COBOL WORKING GROUP
Bruce Mebust
Burlington Northern Railroad
176 East Fifth Street
P.O. Box 64962
St. Paul, MN 55164
(612) 298-2382

CHAIR, SECURITY WORKING GROUP
Rich Harris
General Research Corp.
5383 Hollister Avenue
P.O. Box 6770
Santa Barbara, CA 93160-6770
(805) 964-7724

ASSISTANT CAMPGROUND COORD.
Tom J. Jeffrey
Rockwell International Corp.
1225 N. Alma Road
Richardson, Texas 75081
(214) 996-7818

CHAIR, ADA WORKING GROUP
Lisa Kerby-Rodgers
GTE Government Systems
100 Ferguson Drive
P.O. Box 7188
Mountain View, CA 94039
(415) 966-2720

CHAIR, PROJECT MANAGEMENT W/G
Lynn C. Lewis
Lawrence Livermore National Lab.
University of California
P.O. Box 808
Livermore, CA 94550
(415) 422-8949

CHAIR, OBJ. ORIENTED DES. W/G
Frank B. Modruson
Arthur Andersen & Co.
33 West Monroe Street
Chicago, IL 60603
(312) 580-0033

CHAIR, TeX/LaTEX WEB W/G
John Osudar
Argonne National Lab.
9700 South Cass Avenue
Argonne, IL 60439
(312) 972-7505

CHAIR, VAXset W/G
David J. Powell
The Upjohn Company
7294-25-7
301 Henrietta St.
Kalamazoo, MI 49007
(616) 385-7214

MEM., ANSI DIBOL X3J12 Stds. Comm.
Kenneth Schilling
2314 Mira Vista Avenue
Montrose, CA 91020
(818) 249-0795

SUITE & RECEPTION COORD.
Matt Variot
Eaton Corporation
Box 5009
31717 La Tienda Drive
Westlake Village, CA 91359
(818) 706-5388

CHAIR, TPU/EVE/LSE W/G
John Wilson
Kellogg Company
724 Oak Brook Blvd
Battle Creek, MI 49015
(616) 961-3515

VICE CHAIR
SYMPOSIUM LOGISTICS COORD.
Terry Medlin
Survey Sampling, Inc.
1 Post Road
Fairfield CT 06432
(203) 255-4200

ASST. MASTERS COORDINATOR
Dena Shelton
Cullinet Software Inc.
2860 Zanker Rd, Suite 206
San Jose, CA 95134
(408) 434-6636

CHAIR, BASIC W/G
WISHLIST COORDINATOR
Bob Van Keuren
UserWare International, Inc.
4087 Cahmoune Ave.
San Diego, CA 92105
(619) 283-5285

INCOMING SIG CHAIR
Joseph Pollizzi, III
Space Telescope Science Institute
3700 San Martin Drive
Homewood Campus
Baltimore, MD 21218
(301) 338-4901

CHAIR, SCAN WORKING GROUP
WORKING GROUPS COORD.
VOLUNTEERS COORD.
David Ream
Lexi-Comp
26173 Tallwood Drive
N. Olmsted, OH 44070
(216) 777-0095
(216) 468-0744

CHAIR, PASCAL WORKING GROUP
MEMBER, ANSI PASCAL X3J9 STDS. COMM.
CHAIR, MODULA-2 W/G
STANDARDS COORD.
E. Wayne Sewell
E-Systems, Garland Div.
Box 660023, MS 53700
Dallas, TX 75266-0023
(214) 272-0515 x3553

CHAIR, SOFTWARE METRICS W/G
CAMPGROUND COORD.
MEMBER, ANSI C X3J11 STDS. COMM.
Michael S. Terrazas
LDS Church
50 E. North Temple, 27th Floor
Salt Lake City, UT 84150
(801) 531-3246

MEMBER, ANSI COBOL X3J4, STDS, COMM.
Bruce Gaarder
Donahue Enterprises, Inc.
2441 26th Ave., S.
Minneapolis, MN 55406
(612) 721-2418

ALT. SEMINAR COMM REP.
Janet E. Bressan
Boeing Aerospace Co.
P.O. Box 3999, MS3C-24
Seattle, WA 98124
(206) 773-9404

CHAIR, RPG WORKING GROUP
Charles Williamson
Hargray Telephone Co.
P.O. Box 5519
Hilton Head Is., SC 29938
(803) 686-1204

SEMINAR COMMITTEE REP.
Barry C. Breen
Sundstrand Data Control, Inc.
15001 N.E. 36th Street
P.O. Box 97001
Redmond, WA 98073-9701
(206) 885-8436

MASTERS COORD.
CLINIC DIRECTOR
CHAIR, CASE & TOOLS INTEGRATION W/G
CHAIR, CROSS-DEV. SYSTEMS W/G
George Scott
Computer Sciences Corp.
304 West Route #38, P.O. Box N
Moorestown, NJ 08057
(609) 234-1100

DEVEL. COUNTERPART TECH. LANG.
Leslie J. Klein
Digital Equipment Corp.
ZK02-3/N30
110 Spit Brook Road
Nashua, NH 03062

DIGITAL COUNTERPART
Celeste La Rock
Digital Equipment Corp.
ZK02-3/Q08
110 Spit Brook Road
Nashua, NH 03062

FOLDER EDITOR
Donald E. Amby
Delco Electronics Corporation
P.O. Box 471, MS1A21
Milwaukee, WI 53201
(414) 768-2682

MEMBER ANSI PL/I X3J1 STDS COMM.
Arthur Coston
Applied Information Systems, Inc.
500 Eastowne Dr.
Chapel Hill, NC 27514
(800) 334-5510

ASST. SESSION CHAIRS COORD.
Antonino N. Mione
Rutgers University
Center for Computer & Info. Serv.
Hill Center
P.O. Box 879
Piscataway, NJ 08855-0879
(201) 932-4784

UPDATE.DAILY REPORTER
Terry Shannon
Computer Information Systems
165 Bay State Drive
Braintree, MA 02184
(508) 848-7515

CHAIR, APL WORKING GROUP
Chet Small
MIT Lincoln Laboratory
244 Wood Street
Lexington, MA 02173
(508) 981-4172
(508) 863-5500 x-4172

CHAIR, PL/I WORKING GROUP
Jack Straub
13102 Borgman
Huntington Woods, MI 48070
(313) 358-6338
(313) 541-1941

ASST. NEWSLETTER EDITOR
Jim Wilson
Pfizer Inc.
QC Division
P.O. Box 88
Terre Haute, IN 47808
(812) 299-2121 x271

COMMCOMM REPRESENTATIVE
Kerry Wycoff
1117 E. 1000 Street
Spanish Fork, UT 84660
(801) 240-5948

LARGE SYSTEMS

CHAIR
E.F. Berkley Shands
Washington University
Department of Computer Science
P.O. Box 1045
St. Louis, MO 63136
(314) 889-6636
UUCP:BERKLEY@WUCS.UUCP
BITNET: Berkley@Uunet

COMMUNICATIONS REPRESENTATIVE
NEWSLETTER EDITOR
Clyde T. Poole
The University of Texas at Austin
Department of Computer Sciences
Taylor Hall 2.124
Austin, TX 78712-1188
(512) 471-9551
ARPANET/CSNET:ctp@sally.utexax.edu

36 BIT SYSTEMS
Clive Dawson
Microelectronics & Computer Technology Corp.
9430 Research Blvd.
Echelon Bldg. #1, Suite 200
Austin, TX 78759
(512) 343-0860
ARPANET/CSNET:CLIVE @ MCC. COM

SYMPOSIUM REPRESENTATIVE
Vacant

DISTRIBUTED SYSTEMS
Don Kassebaum
Computation Center
University of Texas at Austin
Austin, TX 78712
(512) 471-3241
ARPANET:CC.KASSEBAUM@A20.CC.UTEXAS.EDU

SEMINARS REPRESENTATIVE
Robert C. McQueen
(201) 428-4242
ARPANET:SIT.MCQUEEN@cu2OB.COLUMBIA.EDU

SUPERCOMPUTING
Vacant

SIG VICE-CHAIRMAN
Ralph M. Bradshaw
Johnson & Johnson
Research &Scientific Services
Management Information Center
Raritan, NJ 08869-1489
(201) 685-3434

LIBRARY REPRESENTATIVE
SIR/MENU BALLOT
Jack Stevens
The Gillette Company
Technical Services, 4U-3
1 Gillette Park
Boston, MA 02106-2131
(617) 463-2089

SIG MARKETING
Steve Attaya
Weiner Enterprises
P.O. Box 23607
Harahan, LA 70183
(504) 733-7055
ARPANET:G.ATTAYA@R20.UTEXAS.EDU

CORPORATE ISSUES
Ralph Bradshaw
Johnson & Johnson
Research and Scientific Services
Management Information Center
Raritan, NJ 08869-1489
(201) 685-3434

DEC COUNTERPARTS
Dave Braithwaite
Digital Equipment Corporation
Marlboro, MA

Rich Whitman
Digital Equipment Corporation
Marlboro, MA

Reed Powell
Digital Equipment Corporation
Marlboro, MA

MUMPS SIG

CHAIRMAN
Chris Richardson
Richardson Computer Research
P.O. Box 8744
La Jolla, CA 92038
(619) 488-6193

NEWSLETTER EDITOR
VICE-CHAIR
COMMCOMM REP.
Mark J. Hyde
Advanced Computing Services
209 Ardsley Drive
DeWitt, NY 13214
BITNET: MJHYDE@SUNRISE
INTERNET: MJHYDE@SUNRISE.ACS.SYR.EDU
(315) 446-7223

SYMPOSIUM SCHEDULER
Brad Hanson
Group Health, Inc.
2829 University Ave., S.E.
Minneapolis, MN 55414
(612) 623-8427

LIBRARY REPRESENTATIVE
PDP-11 WORKING GROUP REP.
Michael McIntyre
PRx, Inc.
43 Bradford Street
Concord, MA 01742
(617) 369-3566

SEMINARS REPRESENTATIVE
Edward Woodward
Science Applications Intl. Corp.
10260 Campus Point Drive MS42
San Diego, CA 92121
(619) 535-7210

CAMPGROUND COORDINATOR
ASSIST. SYMPOSIUM SCHEDULER
Jerry Hsu
Rubicon Corp.
1200 E. Campbell
Richardson, TX 75083
(214) 231-6591

SESSION NOTES EDITOR
Paul A. Price
SciCor, Inc.
2643 Rand Road
Indianapolis, IN 46241
(317)244-8811

PAST CHAIR
MUMPS DEV. COMMITTEE REP.
Mark Berryman
Digital Equipment Corp.
3 Results Way (MR03-2/H7)
Marlborough, MA 01752
(617) 467-4875
BITNET: BERRYMAN@DSM.DEC.COM

DEC COUNTERPART
Dave Smith
Digital Equipment Corp.
2 Iron Way (MR03-2/H7)
Marlborough, MA 01752
(617) 467-2397

ALTERNATE DEC COUNTERPART
Denise Simon
Digital Equipment Corp.
129 Parker Street (PK02-1/M23)
Maynard, MA 01754
(617) 493-9077

NETWORKS SIG

CHAIRMAN
Stuart Lewis
Douglas Furn. of California, Inc.
5020 W. 73rd St., Box 97
South Suburban, IL 60499-0097
(312) 458-1505

SYMPOSIUM COMMITTEE REP
L. Stuart Vance
University of Texas System
Office to Telecomm. Services
Balcones Research Center
10100 Burnet Road
Austin, TX 78758-4497
(512) 471-2416

SEMINARS COMMITTEE REP
Jeffrey Snover
47 Walden Pond Dr.
Nashua, NH 03060
(508) 256-6600

STANDARDS REP
Jim Ebright
Software Results Corp.
2887 Silver Drive
Columbus, OH 43211
(614) 267-2203

COMMCOMM REP
Allen Jay Bennett
Logisticon, Inc.
1551 Alexander SE
Grand Rapids, MI
(616) 452-1823

NEWSLETTER EDITOR
Judi Mandl
University of Conn. Health Center
263 Farmington Ave.
Farmington, CT 06032
(203) 674-3912

SESSION NOTES EDITOR
Mary Marvel-Nelson
General Motors Research Lab.
Warren, MI 48090
(313) 986-1382

WISH LIST COORDINATOR
Stuart L. Labovitz
USAF AFWAL/AADM-2
Bldg 620 Area B
WPAFB, OH 45433-6543
(513) 255-7680

PAST CHAIRMAN
Bill Brindley
HDQ Naval Security Group Cmd.
(202) 282-0527

TECHNOLOGY COORDINATOR
Bill Hancock
ERI Training
P.O. Box 13557
Arlington, TX 76094-0557

MEMBER-AT-LARGE
Sandy Traylor
Target Systems
21063 Carlos Rd.
Yorba Linda, CA 92686

DEC COUNTERPART
Monica Bradlee
Digital Equipment Corporation
550 King St. LKG2-1/U2
Littleton, MA 01460
(508) 486-7341

OFFICE AUTOMATION SIG

(*) CHAIR
Joseph W. Whatley
Nielsen Media Research
375 Patricia Ave.
Dunedin, FL 34698
(813) 734-5473 x 2438

(*) VICE CHAIR
Ralph Bradshaw
Johnson and Johnson
Route 202
Raritan, NJ 08869-1489
(201) 685-3434

SIR PROGRAM ADMINISTRATOR
Edward L. Bowen
VAX Systems Manager
BellSouth Services
1876 Data Drive, Room N204
Birmingham, AL 35244
(205) 988-6800

LUG COORDINATOR & OASIG FOUNDER
Tom Orlowski
American Council on Education
One Dupont Circle (Suite 110)
Washington, DC 20036
(202) 939-9371

OASIS ADMINISTRATOR
Dale E. Coy
E-8, MS/J957
Los Alamos National Laboratory
P.O. Box 1663
Los Alamos, NM 87545
(505) 667-3270

OASIS ASSISTANT ADMINISTRATOR
LIBRARY REPRESENTATIVE
Bob Hassinger
Liberty Mutual Research Ctr.
71 Frankland Road
Hopkinton, MA 01748
(508) 435-9061

SUITE COORDINATOR
Jeanne Ward
State University of New York at Stony Brook
Stony Brook, NY 11794-3499
(516) 632-7795

ASSISTANT SUITE COORDINATOR
Mary L. Keenan
State University of New York at Stony Brook
Stony Brook, NY 11794-2400
(516) 632-8053

(*) SYMPOSIA COORDINATOR
Mitch Brown
Bank of Boston
Central Systems, 99-02-07
Box 2016
Boston, MA 02106-2016
(508) 434-1587 / 6318

ASSISTANT SYMPOSIA COORDINATOR
Lynda L.Peach
Mustang Energy Corp.
1100 First National Center East
Oklahoma City, OK 73102
(405) 272-9471

ROADMAP/PUBLICATIONS COORDINATOR
Scott McClure
Office Systems Senior Analyst
Industrial Tape Division/3M
220-8E-01 3M Center
St. Paul, MN 55144-1000
(612) 736-4297

SESSION CHAIR COORDINATOR
Dee Wade
Office Automation Specialist
Silicon Systems
14351 Myford Road
Tustin, CA 92680
(714) 731-7110

SPEAKER COORDINATOR
Monica Falk
MIF Enterprises
3260 S. Irving Street, Suite C202
Englewood, CO 80110
(303) 781-5726

CAMPGROUND COORDINATOR
LT. Scott Cline
AFWAL/AATC
Wright-Patterson AFB, OH 45433-6543
(513) 255-5623

(*) COMMUNICATIONS COMMITTEE REP
Mary-Jane Bolling
Foreign Mission Board
3806 Monument Avenue
Richmond, VA 23230
(804) 353-0151

NEWSLETTER EDITOR
Therese LeBlanc
T.M. LeBlanc & Assoc.
275 London
Wheeling, IL 60090
(312) 459-1784

TAPE COORDINATOR
Roger Bruner
Foreign Mission Board
3806 Monument Avenue
Richmond, VA 23230
(804) 353-0151

STORE COORDINATOR
Ellen Shields
E-Systems
P.O. Box 12248
St. Petersburg, FL 33733
(813) 381-2000 x2010

ASSISTANT STORE COORDINATOR
Linda A. Peters
E-Systems
P.O. Box 12248
St. Petersburg, FL 33733
(813) 381-2000 x3517

SIG 'EFFECTS' COORDINATOR
Robert O. Wilkinson
Naval Underwater Systems Center
Code 301
New London, CT 06320
(203) 440-4654

SESSION NOTES EDITOR
George Bone
Code 2301.2 mailstop T-11
Mare Island Naval Shipyard
Vallejo, CA 94592-5100
(707) 646-2531

SIG PERQS COORDINATOR
Enid Woods
Word Processing Coordinator
Crowe, Chisek and Company
330 E. Jefferson Blvd.
P.O. Box 7
South Bend, IN 46624
(219) 232-3992

(*) SPECIAL PROJECTS COORDINATOR
Katherine Trimm
PIVOTAL, Inc.
6892 Dorado Ct.
Tucson, AZ 85715
(602) 886-5563

SECURITY WORKING GROUP CHAIR
Ray Kaplan
PIVOTAL, Inc.
6892 E. Dorado, CT
Tucson, AZ 85715
(602) 886-5563

(*) Executive Committee Members

OASIS (OA SIG NOTES CONFERENCES)

(603) 884-1738	1200 baud	8 bit, no parity
(603) 88401742	2400 baud	8 bit, no parity

PERSONAL COMPUTER SIG

CHAIR
Lynn Jarrett
San Diego Union-Tribune Pub. Co.
350 Camino de la Reina
San Diego, CA 92108
(619) 293-1130

WORKSTATIONS/MACS/PRO
WORKING GROUP CHAIRMAN
Thomas R. Hintz
Univ. of Florida
IFAS Computer Network,
Bldg. 120
Gainesville, FL 32611
(904) 392-5180

VICE, CHAIR RAINBOW W/G CHAIRMAN
Lynn Jarrett
Union Tribune Publishing Co.
P.O. Box 191
San Diego, CA 92108
(619) 299-3131 x1130

VAXMATE WORKING GROUP CHAIRMAN
Frederick G. Howard
Eastman Kodak Company
901 Elmgrove Road, D345-LP
Rochester, NY 14650
(716) 253-2363

VOLUNTEER COORDINATOR
Pierre M. Hahn
SUNY HSC-T10-028-8101
Stony Brook, NY 11794

LIBRARIAN Rep.
Ron S. Hafner
Hafner and Associates
P.O. Box 2924
2499 Wellingham Dr.
Livermore, CA 94550
(415) 422-2149

COMMUNICATIONS REPRESENTATIVE
Kenneth LeFebvre
Sytek, Inc.
19 Church St.
P.O. Box 128
Berea, OH 44017
(216) 243-1613

NEWSLETTER EDITOR
Gary Rice
McDonnell Douglas
5555 Garden Grove Blvd.
MS: K200 77/200
Westminster, CA 92683
(714) 952-6582

RAINBOW/DECmate W.G. CHAIR
Vince Perriello
Crosfield Composition Systems
One Crosfield Ave.
West Nyack, NY 10994
(914) 353-4000

SYMPOSIA COORDINATOR
Jimbo Wilson
Natl Tech. Inst. for Deaf
Rochester Inst. of Tech.
P.O. Box 9887
Rochester, NY 14623
(716) 475-6241

SESSION NOTES EDITOR
Dr. Tom. Warren
Oklahoma State Univ.
Dept. of English
Dir. Tech. Writing Program
Stillwater, OK 74078
(405) 624-6138

PCSA WORKING GROUP CHAIRMAN
To be announced

MEMBERS-AT-LARGE

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

Theodore Needleman
Odea Tech.
67 W. Burda Pl.
Spring Valley, NY 10977
(914) 250-100

DEC COUNTERPARTS
PERSONAL COMPUTING SYSTEMS GROUP

Anita Uhler
Digital Equipment Corporation
LJ02/13
30 Porter Road
Littleton, MA 01460
(617) 486-2397

PRO

Jeff Slayback
Digital Equipment Corp.
ML021-2/U12
146 Main Street
Maynard, MA 01754
(617) 493-9340

BUTTON COORDINATOR

Ken Stricker
Martin Marietta Aerospace
P.O. Box 5837 MP 320
Orlando, FL 32855
(305) 356-6589

CAMPGROUND COORDINATOR

Jim Hobbs
Adolf Coors Co.
Golden, CO 80401-1295
(303) 277-2855

SEMINARS COORDINATOR

Tim Bundrick
3480 TCHTW/TTVC
Goodfellow AFB,TX 76908-5000
(915) 657-5424

RSTS SIG

CHAIRMAN

Charles Mustain
Stark County School system
Data Services Division
7800 Columbus Rd. N.E.
Louisville, OH 44641
(216) 875-1431

COMMUNICATIONS REPRESENTATIVE
STORE REPRESENTATIVE

Ed Beadel
Instructional Computer Center
S.U.N.Y. College at Oswego
Oswego, N.Y. 13126
(315) 341-3055

SYMPOSIA COORDINATOR

Glenn Dollar
Digital Computer Consultants Inc.
21363 Lassen St., Suite 205
Chatsworth, CA 91311
(818) 341-9171

ASS'T SYMPOSIA COORDINATOR

Dan Stoller
Natural Country Farms
P.O. Box 758
58 West Road
Rockville, CT 06066
(203) 872-8346

NEWSLETTER EDITOR

Terence M. Kennedy
St. Peter's College
Department of Computer Science
2641 Kennedy Blvd.
Jersey City, NJ 07306
(201) 435-1890

LIBRARY REPRESENTATIVE

R.R. Patel (PAT)
Krupp/TaylorUSA
12800 Culver Blvd
Los Angeles, CA 90066
(213) 306-3646

PRE-SYMPOSIA SEMINAR COORDINATOR

Scott Castleberry
1750 North Collins
Suite 108
Richardson, TX 75080
(214) 437-3477

VICE CHAIRMAN
WISH LISTS COORDINATOR

Lynnell Koehler
Campus America
POISE Prod. Ctr.
201 North Nevada Avenue
Roswell, NM 88201
(505)625-5500

EDUSIG LIAISON

George Wyncott
Purdue University Computer Center
W. Lafayette, IN

RSTS PRODUCT PLANNING COORDINATOR

Errol E. Ethier
Information Design and Management, Inc.
23 Hunting Avenue
(617) 842-4220
Shrewsbury, MA 01545

DEC COUNTERPART

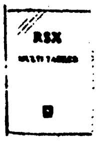
Kathy Waldron
Digital Equipment Corporation
Continental Blvd.
Merrimack, NH 03054

MEMBERS-AT-LARGE

Edward F. Beadel, Jr.
Manager
Instructional Computing Center
S.U.N.Y. College at Oswego
Oswego, NY 13126
(315) 341-3055

Mark Hartman
Jadtec Computer Group
546 W. Katella Avenue
Orange, CA 92667
(714) 997-8928

Jeff J. Killeen
Information Design & Management, Inc.
31 Hopedale Street
Hopedale, MA 01747



RSX SIG

CHAIRMAN

Dan Eisner
Perkin-Elmer Corp.
Garden Grove, CA

SYMPOSIA COORDINATOR

Rick Sharpe
Toledo Edison
Toledo, OH

PRE-SYMPOSIUM SEMINAR COORDINATOR

Hans Jung
Associated Press
New York, NY

COMMUNICATIONS REPRESENTATIVE

Jay Allen Bennett
Logisticon, Inc.
5035 Whitneyville Road
Ada, MI 49301
(616) 452-1823

NEWSLETTER EDITOR
MULTI-PROCESSORS WORKING GROUP COORDINATOR

Bruce Mitchell
Machine Intelligence & Industry Magin
Byron, MIN

STORE COORDINATOR

Jim Hopp
Carlton Financial Computation
South Bend, IN

SESSION NOTE EDITOR

Burt Janz
BHJ Associates
Nashua, NH

LIBRARIAN

Glenn Everhart
Mt. Holly, NJ

CAMPGROUND COORDINATOR

Jerry Ethington
Prolifix Inc.
Frankfort, KY

DEC COUNTERPARTS

Lin Olsen
Nashua, NH

Dick Day
Nashua, NH

WORKING GROUP COORDINATOR

Sharon Johnson
Epidemiology
Minneapolis, MN

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
RCS
Moorestown, NJ

VOLUNTEER COORDINATOR

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

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

RT-11 SIG

CHAIRMAN

John T. Rasted
JTR Associates
58 Rasted Lane
Meriden, CT 06450
(203) 634-1632

COM. COM VOTING REP.
COBOL CONTACT

Bill Leroy
The Software House, Inc.
P.O. Box 52661
Atlanta, GA 30355-0661
(404) 231-1484

STANDARDS COORDINATOR

Robert Roddy
Naval Ship Research Ctr.
Bethesda, MD 20084
(301) 227-1724

MACRO CONTACT

Nick Bourgeois
NAB Software Services Inc.
P.O. Box 20009
Albuquerque, NM 87154
(505) 298-2346

NEWSLETTER EDITOR
TECO CONTACT
PRODUCT PLANNING CONTACT

John M. Crowell
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616
(916) 756-3291

NETWORKING CONTACT

Jim Crapuchettes
Omnex Corp.
2483 Old Middlefield Way
Mountain View, CA 94043
(415) 966-8400

WISH LIST CONTACT
UNIX/ULTRIX CONTACT

Bradford Lubell
L.A. Heart Lab, UCLA
10833 Le Conte Avenue
Los Angeles, CA 90024-1760
(213) 206-6119

TSX & C CONTACT

Jack Peterson
Horizon Data Systems
P.O. Box 29028
Richmond, VA 23229
(804) 740-9244

RUNOFF CONTACT

John Davis
Naval Ship Research Center
Code 2950
Bethesda, MD 20084
(301) 227-1592

LUG CONTACT

Ned Rhodes
Software Systems Group
2001 North Kenilworth St.
Arlington, VA 22205
(703) 534-2297

PERSONAL COMPUTERS

Dennis V. Jensen
AMES Labs. ISU/USDOE
310 Metallurgy
Ames, Iowa 50011
(515) 294-4823

SYMPOSIA COORDINATOR

Milton Campbell

Talisman Systems
Drawer CP-255
Manhattan Beach, CA 90266
(213) 318-2206

TAPE COPY GENERATION
TAPE COPY DISTRIBUTION
RT DECUS LIBRARY CONTACT

Tom Shinal
Syntropic Technology
P.O. Box 198
Waterford, VA 22190
(703) 882-3000

PRE-SYMPOSIUM SEMINAR
RT-11 SUITE MANAGER

Bruce Sidlinger
Sidlinger Computer Corp.
4335 N.W. Loop 410, #209
San Antonio, TX 78229
(512) DIG-ITAL

BASIC CONTACT

Ralston Barnard
Div 7523
Sandia Labs
Alburquerque, NM 87185
(505) 844-5115

PRO RT-11 & HARDWARE

Bill Walker
Monsanto Research Corp.
P.O. Box 32, A-152
Miamisburg, OH 45342
(513) 865-3557

FORTRAN CONTACT

Robert Walraven
Multiware, Inc.
2121-B 2nd St. Suite 107
Davis, CA 95616
(916) 756-3291

OTHER LANGUAGES

Gary Sallee
19912 Fernglen Drive
Yorba Linda, CA 92686
(714) 970-2864

SITE SIG

CHAIRMAN

Timothy Fraser
Specialized Bicycle Components
15130 Concord Circle #77
Morgan Hill, CA 95037
(408) 779-6229

SYMPOSIA COORDINATOR

Sue Abercrombie
48 Malilly Rd.
Portland, ME 04103
(207) 772-2837

SESSION NOTE EDITOR
LARGE SYSTEMS SIG LIAISON

Gary Bremer
Emerson Electric Co.
8100 W. Florisant
St. Louis, MO. 63136
(314) 553-4448

NEWSLETTER EDITOR
NETWORKS SIG LIAISON
OA SIG LIAISON

Gregory N. Brooks
Washington University
Behavior Research Labs
1420 Grattan St.
St. Louis, MO. 63104
(314) 241-7600 ext. 257

HARDWARE COORDINATOR

HMS SIG Liason
Emily Kitchen
A.H. Robins Co.
1211 Sherwood Ave. RT-2
Richmond, VA. 23220
(804) 257-2925

COMMUNICATIONS COMMITTEE REPRESENTATIVE
AI SIG Liason

Terry C. Shannon
Digital Review
160 State St.
6th Floor
Boston, MA. 02109
(617) 367-7190

PRE-SYMPOSIA SEMINAR COORDINATOR

Phillip Ventura

STAFF MANAGEMENT

Adam Zavitski
Simmonds Precision ICD
3100 Highland Blvd.
Raleigh, NC. 27625
(919) 872-9500

MEMBERS-AT-LARGE

Ann Goergen
Texas Instruments
13510 N. Central
M/S 437
Dallas, TX. 75266
(214) 995-4629

HMS SIG Liason
RT SIG Liason

David Hunt
Lawrence Livermore National Lab
MS L-230
P.O. Box 808
Livermore CA. 94550
(802) 656-3190

Gary Siftar
Digital Equipment Corporation
Tulsa, OK.

DEC COUNTERPARTS

Joe Allen
Stow MA.

Lil Holloway
Bedford MA.

Susan Porada
Marlboro, MA.

UNISIG

CHAIRMAN

Kurt L. Reisler
Hadron Incorporated
9990 Lee Highway
Fairfax, VA 22030
(703) 359-6100
decvax!seismo!hadron!klr

SYMPOSIA COORDINATOR

Sam Kimery
Purdue University Math Dept.
Math Science Bldg.
West LaFayette, IN 47907
(317) 494-6055
ucbvax!helicon.math.purdue.edu!kimery

SESSION NOTES EDITOR

William Cheswick
New Jersey Institute of Tech.
Computing Services
323 Martin Luther King Blvd.
Newark, NJ 07102
(201) 596-2900
bellcore!njitcccc!bc

NEWSLETTER EDITOR

Sharon Gates-Fishman
NDC Corporation
730 E Cypress Avenue
Monrovia, CA 91016
(818) 358-1871
!amdahl!cit-vax!ndc!sgf

COMMCOMM REPRESENTATIVE
James W. Livingston, Jr.
Measurex Automation Systems
10411 Bubb Rd
Cupertino, CA 95014-4150
(408) 973-1800 x 766
ihnp4!masl!jwl

ADMINISTRATIVE DAEMON
Dorothy A. Geiger
The Wollongong Group
49 Showers Drive, #451
Mountain View, CA 94040
(415) 948-1003
ihnp4!decwrl!dgeiger

TAPE LIBRARIAN
Carl Lowenstein
Marine Physical Laboratory
Scripps Institute of Oc'graphy, P-004
LaJolla, CA 92093
(619) 294-3678
[ihnp4,decvax,akgua,dcdwest,ucbvax
!sdcsvax!mplvax!cdl

USENET LIAISON
STANDARDS COORDINATOR
Ed Gould
Mt. Xinu
2910 7th St.
Suite 120
Berkley, CA 94710
(415) 644-0146
ucbvax!mtxinu!ed

MINISTER WITHOUT PORTFOLIO
Norman Wilson
Bell Laboratories, 2C-529
600 Mountain Avenue
Murray Hill, NJ 07974
(201) 582-2842
[decvax,ihnp4]!research!norman

SEMINARS COORDINATOR
Steven Stepanek
Computer Science Dept.
School of Eng. & Computer Science
18111 Nordhoff St.
Northridge, CA 91330
(818) 885-2799 or 3398
ihnp4!csun!sgs

DEC COUNTERPART
Gary Oden
Digital Equipment Corporation
Continental Blvd, MK02
Merrimack, NH 03054
(603) 884-5111
decvax!oden

VAX SYSTEMS SIG

CHAIRMAN
Susan T. Rehse
Lockheed Missiles & Space Co.
0/19-50, B/531,
P.O. Box 3504
Sunnyvale, CA 94088-3504

KECUTIVE COMMITTEE
David Cossey
Computer Center
Union College
Schenectady, NY 12308

Jeffrey Jalbert
J C C
P.O. Box 381
Granville, OH 43023

Betsy Ramsey
American Mathematical Society
P.O. Box 6248
Providence, RI 02940

David Schmidt
Management Science Associates
6565 Penn Avenue
Pittsburgh PA 15206-4490

Al Siegel
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201-2693

David Wyse
Projects Unlimited
3680 Wyse Road
Dayton, OH 45414-2539

SYMPOSIA COORDINATOR
Betsy Ramsey
American Mathematical Society
P.O. Box 6248
Providence, RI 02940

SYMPOSIA COORDINATOR, ASST.
Michael Carullo
Westinghouse Electric Corp.
10120 Kings Bench Court
Ellicot City, MD 21043

SESSION CHAIRMAN COORDINATOR
Elaine Hall
Westinghouse
P.O. Box 746
M/S 432
Baltimore, MD 21203

SEMINAR COORDINATOR
Robert McQueen
Knoll Pharmaceuticals
MIS Department
30 North Jefferson Road
Whippany, NJ 07981

LIBRARY COORDINATOR
Glen Everhart
General Electric
25 Sleigh Ride Road
Glen Mills, PA 19342

COMMUNICATIONS COORDINATOR
David Wyse
Projects Unlimited
3680 Wyse Road
Dayton, OH 45414-2539

NEWSLETTER EDITOR
Clyde T. Poole
The University of Texas at Austin
Department of Computer Sciences
Taylor Hall 2.124
Austin, TX 78712-1188
ARPANET/CSNET: ctp@sally.utexas.edu

PAGESWAPPER VAX MANAGER
Lawrence J. Kilgallen
Box 81, MIT Station
Cambridge, MA 02139-0901

SESSION NOTES EDITOR
Ken Johnson
Meridien Technology
P.O. Box 2006
St. Louis, MO 63011

SYMPOSIUM HANDOUT
BOOTBLOCK EDITOR
John L. Prather
Technicon Instruments Corp.
511 Benedict Avenue
Tarrytown, NY 10591

STORE COORDINATOR
G. Beau Williamson
Rockwell International
1200 N. Alma Road
MIS 406-280
Richardson, TX 75081

MASTER'S LIST COORDINATOR
Carl Friedberg
Seaport Systems
165 William Street, 9th Floor
New York, NY 10038-2605

SYSTEM IMPROVEMENT REQUEST (SIR)
David Schmidt
Management Science Associates
6565 Penn Avenue
Pittsburgh, PA 15206-4490

VOLUNTEER COORDINATOR
Elizabeth Bailey
222 CEB
Tennessee Valley Authority
Muscle Shoals, AL 35661

WEST COAST VOLUNTEER COORDINATOR
Ron Tencati
Jet Propulsion Lab
4800 Oak Grove Drive
MS:602-145
Pasadena, CA 91109

CAMPGROUND COORDINATOR
David Paul
Shell Oil Company
MS:D2145
P.O. Box 1380
Houston, TX 77251-1380

SUITE COORDINATOR
Open

WORKING GROUP COORDINATOR
Jeffrey Jalbert
J C C
P.O. Box 381
Granville, OH 43023

COMMERCIAL WORKING GROUP CHAIR
E.F. Berkley Shands
Washington University
Department of Computer Science
P.O. Box 1045
St. Louis, MO 63136

DECNET SECURITY WORKING GROUP CHAIR
Ron Tencati
Jet Propulsion Lab
4800 Oak Grove Drive
MS:602-145
Pasadena, CA 91109

HIGH-END VAX SYSTEMS WORKING GROUP CHAIR
Al Siegel
Battelle Memorial Institute
505 King Avenue
Columbus, OH 43201-2693

INTERNALS WORKING GROUP CHAIR
Carl Friedberg
Seaport Systems
165 William Street, 9th Floor
New York, NY 10038-2605

MICROVAX WORKING GROUP CHAIR
David Mehren
Pivotal, Inc.
6892 East Dorado Court
Tucson, AZ 85715-3264

MIGRATION AND HOST DEVELOPMENT
VAXINTOSH WORKING GROUP CHAIR
Jim Downward
KMS Fusion Inc.
P.O. Box 156D
Ann Arbor, MI 48106

MULTIPROCESSOR WORKING GROUP CHAIR
Eugene Pal
U.S. Army
CAORA (ATORCATC)
Fort Leavenworth, KA

PERFORMANCE WORKING GROUP CO-CHAIR
John T. Peterson
Data Metrics Systems Inc.
5270 Lyngate Court
Burke, VA 22015

PERFORMANCE WORKING GROUP CO-CHAIR
Louise Wholey
Measurex Corporation
One Results Way
Cupertino, CA 95014

REAL TIME/FOREIGN DEVICES WORKING GROUP CH
Larry Robertson
Bear Computer Systems
56512 Case Avenue
North Hollywood, CA

SECURITY WORKING GROUP CHAIR
C. Douglas Brown
Sandia National Labs
Division 2644
P.O. Box 5800
Alburquerque, NM 87185-5800

SYSTEM MANAGEMENT WORKING GROUP CHAIR
Steve Tihor
251 Mercer Street
New York, NY 10012

VAXCLUSTER WORKING GROUP CHAIR
Thomas Linscomb
Computation Center
University of Texas
Austin, TX 78712

VMS USER'S NETWORK WORKING GROUP CHAIR
Jamie Hanrahan
Simpact Associates
9210 Sky Park Court
San Diego, CA 92123

ADVISORS
Joseph Angelico
U.S. Coast Guard Detachment
National Data Buoy Center
NSTL Station, MS 39529-6000

Jack Cundiff
Horry-Georgetown Tech. College
P.O. Box 1966
Conway, SC 29526

Don Golden
Shell Oil Company
MS:D2145
P.O. Box 1380
Houston, TX 77251-1380

Marg Knox
Computation Center
University of Texas
Austin, TX 78712

Art McClinton
Mitre
1820 Dolley Madison Blvd.
McLean, VA 22102

Ross Miller
Online Data Processing
N 637 Hamilton
Spokane, WA 99202

SUBMITTING ARTICLES TO HARD NEWS

The purpose of HARD NEWS, 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 would like to publish it in the newsletter. We hope that HARD NEWS will be published at least six times a year.

You can submit material to the editor, Carmen Wiseman, or to the HMS SIG chair, Bill Walker. We can accept submissions in a wide variety of formats:

- o Items can be sent to the editor on VMS-format RX50s, TK50 cartridges, or IBM PC format 5 1/4" floppies. The SIG chair prefers RT-11 floppies but can handle any reasonable media.
- o Hard copy, like cash, is always acceptable. Camera-ready copy will save us a lot of typing, but we don't insist on it. You can also use the Hardware Submission Form in the "Questionnaires" section of the combined SIGs Newsletters.
- o Those of you with access to DCS can send things to WALKER or WISEMAN. DCS is usually checked on a daily basis.
- o You can reach the SIG chair on CompuServe as "Bill Walker 71066,24" or via EasyLink mailbox 62752448 or MCI Mail account 333-1675. You can reach the editor via EasyLink mailbox 62960090 (be sure to say ATTN: or TO: Carmen Wiseman somewhere in the body of the message).

If you have anything to submit, send it! If it is a mess, but we can read it, we will get it into the newsletter somehow. Finally, if you have any questions about submitting material, call one of us. The telephone numbers are listed below.

Contributions can be sent to:

William K. Walker
Monsanto Research Corp.
P.O. Box 32 A-152
Miamisburg, OH 45342
(513) 865-3557 (work)
(513) 426-7094/0344 (home)

OR
==

Carmen D. Wiseman
Digital Review
Prudential Tower, Suite 1390
800 Boylston Street
Boston, MA 02199
(617) 375-4361 (work)

Ask the WOMBAT WIZARD
Submission Form

To submit a problem to the WIZARD, please fill out the form below and send it to:

WW Editor, Philip A. Naecker
Consulting Software Engineer
3011 North Mount Curve Avenue
Altadena, CA 91001
USA

Name:________________________________ DECUS Membership No. __________

Affiliation:____

Address:____

____

____

Telephone Number:____

Statement of Problem:____

____

____

____

____

____

Please following the following guidelines when submitting support material:

1. If you are trying to demonstrate a method or a concept, please simplify the procedures, records, and other information to the shortest form possible.

2. Annotate your attachments. Simple comments or hand-written notes ("Everything worked until I added this statement.") go a long way toward identifying the problem.

3. Keep an exact copy of what you send. And number the pages on both copies. But send everything that is related to your question, even remotely.

4. If you would like a direct response or would like your materials returned, please don't forget to include a stamped, self-addressed envelope large enough to hold the materials you send.

DATATRIEVE/4GL SIG
1989 Product Improvement Request Submission Form
Submission Deadline - January 13, 1989

Submittor:	DECUS Membership Number:
Address:	Firm:
Phone:	Product or Products:

How to write a PIR.

A PIR should be directed at a specific product or group of products. Be sure to give the full name of the product(s) and version numbers if applicable. Describe the functionality you would like to see in as complete terms as possible. Don't assume that the PIR editors or software developers know how it is done in some other software product - state specifically how you want the software to function. Provide justification of your request and give an example of its use. If you can, suggest a possible implementation of your request.

Abstract: (Please limit to one or two short sentences.)

Description and Examples: (Use additional pages as necessary.)

Mail submission to:

T. Chris Wool, PIR Editor
E. I. duPont DeNemours & Co.
Engineering Department
P. O. Box 6090
Newark, DE 19714-6090

HMS SIG

HARDWARE SUBMISSION FORM -- A SIG INFORMATION INTERCHANGE

Message

Contact
Name

Address

Telephone

Type of equipment

SUBMIT ANY TYPE OF HARDWARE PROBLEMS AND/OR FIXES.

SEND TO:

```
William K. Walker              Carmen D. Wiseman
Monsanto Research Corp.   OR   Digital Review
P.O. Box 32  A-152        ==   Prudential Tower, Suite 1390
Miamisburg, OH  45342          800 Boylston Street
                               Boston, MA  02199
```

MASTERS APPLICATION

Name: ______________________ **Title** ______________________

Company: ______________________

Address: ______________________

____

______________________ **Phone:** (   ) ______________________

Network Address: ______________________ **Date:** ______________________

The Languages & Tools SIG has established the designation "LANGUAGES AND TOOLS MASTER", to be applied to selected, qualified people willing to share their expertise in various subjects with others. Masters are people who are knowledgeable enough in one or more languages or tools to be comfortable answering questions about them. The qualifications of an L&T Master are: expertise in a specific area, a willingness to have his/her name published as a Master, and a willingness to volunteer services in different ways. Each product may have several Masters, and there is an overall Masters Coordinator who is a member of the L&T Steering Committee.

Masters are asked to serve other users (and, under some circumstances, DEC), as a resource on products within their competence. In addition to being listed in the L&T Masters Directory (published in the newsletter) as available for occasional telephone consultation, Masters may act as 'Doctors' at Symposium Clinics, present Symposium sessions on the products of interest to them, field test products, interact with DEC product managers when appropriate, or act as a reference for a product for Digital salespeople. Especially on mature products, the SIG is anxious for knowledgeable users to offer product tutorial sessions at Symposia, and Masters can be of great help here. At Symposia, Masters will wear an identifying button bearing the legend "Ask Me About....." and the name of the language or tool in which he/she specializes.

If you'd like to serve as an L&T Master, please mark the products on which you are willing to answer questions with an "M" (for Master). Please mark any other products running at your site with an "A" (for "also running") to provide users with a broader picture of your facilities. (Although not an L&T product, Mumps is included here at the request of the Mumps SIG as a service to Mumps users). You may request removal of your name from the Masters Directory at any time, although you may continue to be listed for a month or two, because of publication lead times.

I am qualified to act as an L&T Master for the following products: □ Mumps

□ Debug
□ Pascal
□ Fortran
□ Document
□ VAX Notes

□ Bliss
□ Basic
□ Cobol
□ Dibol
□ Emacs

□ CMS
□ MMS
□ LSE
□ SCA
□ PCA

□ TPU
□ EVE
□ EDT
□ TECO
□ PL/I

□ C
□ Ada[1]
□ APL
□ RPG
□ Scan

□ Test Manager
□ Runoff & DSR
□ TeX & LaTeX
□ Cobol Generator
□ Software Project Mgr

Briefly describe your experience with those you checked. ______________________

____

____

____

How long have you held your present position? ______________________

Are you able to attend at least one symposium each year? ______________________

Users are encouraged to seek assistance with products by calling appropriate Masters listed in the Directory. As a Master, your name and telephone number will be published in the Masters Directory, and users will call on you for limited help from time to time. Please check, below, any additional activities you might do:

□ Field-test new versions of your product at your work site.

□ Provide feedback on the product when needed by its DEC product manager.

□ Act as a reference for the product at the request of Digital Sales or Marketing people.

Mail to: **George Scott, L&T SIG Masters Coordinator, Computer Sciences Corporation, 304 West Route #38, P.O. Box N, Moorestown, NJ 08057.**

[1] Ada is a trademark of the DoD

Languages & Tools SIG

WISHLIST QUESTIONNAIRE

Name: ______________________________ **Title** ______________________________

Company: ____

Address: ____

____

______________________________ **Phone:** (   ) ______________________________

Network Address: ____________________________ **Date:** _________________________

The Languages & Tools SIG is principally concerned with the DEC and public domain software products listed below. If your request directly involves one of these products, please check which one (if you have more than one request, please use a separate form for each):

□ Debug
□ Pascal
□ Fortran
□ Document
□ VAX Notes

□ Bliss
□ Basic
□ Cobol
□ Dibol
□ Emacs

□ CMS
□ MMS
□ LSE
□ SCA
□ PCA

□ TPU
□ EVE
□ EDT
□ TECO
□ PL/I

□ C
□ Ada[1]
□ APL
□ RPG
□ Scan

□ Test Manager
□ Runoff & DSR
□ TeX & LaTeX
□ Cobol Generator
□ Software Project Mgr

If your request or suggestion doesn't relate to one of the products listed above, check which one of the following Language & Tools SIG topics it concerns:

□ Newsletter
□ Masters Program
□ Information Folder
□ Other L&T SIG topic: ______________________________________

□ Symposium Sessions
□ Working Group Activities
□ SIG Tape

□ Pre-Symposium Seminars
□ Session Notes
□ DECUS Store Item

Wish List Request—brief description: ______________________________________

____

____

____

Complete description—please explain your request thoroughly; don't assume we know details of other products or services; give examples. ______________________________________

____

____

____

____

____

____

____

____

____

____

____

____

____

Mail to: **Shava Nerad, L&T Wishlist Coordinator, MIT, 77 Mass Ave. W91-219A, Cambridge, MA 02139**; (617)253-7438

[1] **Ada is a trademark of the DoD**

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:

JUDI MANDL
UCONN HEALTH CENTER
263 FARMINGTON AVENUE, BLDG. #19
FARMINGTON, CT 06032

Title:

Message:

Your Name:
Address:

Telephone:

If this is a reply to a previous DATAGRAM, what #?

Signature: Date:

Place
Stamp
Here

JUDI MANDL
UCONN HEALTH CENTER
263 FARMINGTON AVENUE, BLDG. #19
FARMINGTON, CT 06032

Fold Here

OFFICE
AUTOMATION
SIG

SYSTEM IMPROVEMENT REQUEST BALLOT

DECUS
membership
number

INSTRUCTIONS: System Improvement Request (SIR) ballots allow you, the user to assist in the priortization of the submitted SIR's before they are forwarded to DEC. The total number of points you may allocate on this ballot may not exceed 100 points (absolute value). No more then 10 points may be given to any single SIR. Your ballot must be received by Sept. 3 to be counted.

SIR NUMBER POINTS SIR NUMBER POINTS

Send To:

Edward L. Bowen
1876 Data Dr. Rm N-204
Birmingham, Al.
35244

BellSouth Services

RT-11 WISH LIST SURVEY

Name (optional) ________________________________

Address (optional) ________________________________
____
____

DECUS Number (optional) --

1.1 ___
1.2 ___
1.3 ___
1.4 ___
1.5 ___
1.6 ___
1.7a ___
1.7b ___
1.8 ___
1.9a ___
1.9b ___
1.9c ___
1.9d ___
1.10 ___
1.11 ___
1.12 ___
1.13 ___
1.14 ___
2.1 ___
2.2 ___
2.3 ___
2.4 ___
2.5 ___
2.6 ___
2.7 ___
2.8 ___
2.9 ___
2.10 ___
2.11 ___
2.12 ___
2.13 ___
2.14 ___
2.15 ___
2.16 ___
2.17 ___
2.18 ___
2.19 ___
2.20 ___
2.21 ___
2.22 ___
2.23 ___
2.24 ___

3.1 ___
3.2a ___
3.2b ___
3.2c ___
3.2d ___
3.2e ___
3.3a ___
3.3b ___
3.3c ___
3.3d ___
3.4a ___
3.4b ___
3.4c ___
3.5a ___
3.5b ___
3.6a ___
3.6b ___
3.6c ___
3.6d ___
3.6e ___
3.6f ___
3.6g ___
3.7a ___
3.7b ___
3.7c ___
3.7d ___
3.7e ___
3.7f ___
3.7g ___
3.7h ___
3.7i ___
3.7j ___
3.7k ___
3.7l ___
3.7m ___
3.7n ___
3.7o ___
3.7p ___
3.7q ___
3.7r ___
3.7s ___
3.7t ___

3.7u ___
3.7v ___
3.7w ___
3.7x ___
3.7y ___
3.7z ___
3.7aa ___
3.7bb ___
3.7cc ___
3.7dd ___
3.7ee ___
3.8a ___
3.8b ___
3.8c ___
3.9a ___
3.9b ___
3.9c ___
3.9d ___
3.9e ___
3.9f ___
3.9g ___
3.9h ___
3.9i ___
3.9j ___
3.9k ___
3.10a ___
3.10b ___
3.10c ___
3.10d ___
3.10e ___
3.10f ___
3.10g ___
3.10h ___
3.10i ___
3.10j ___
3.10k ___
3.10l ___
3.10m ___
3.10n ___
3.11a ___
3.11b ___
3.12 ___

3.13a ___
3.13b ___
3.13c ___
3.13d ___
3.14 ___
3.15 ___
3.16 ___
3.17a ___
3.17b ___
3.17c ___
3.17d ___
3.17e ___
3.17f ___
3.18 ___
3.19a ___
3.19b ___
3.19c ___
4.1 ___
4.2a ___
4.2b ___
4.3 ___
4.4a ___
4.4b ___
4.5a ___
4.5b ___
4.6 ___
4.7a ___
4.7b ___
4.7c ___
4.7d ___
4.7e ___
4.7f ___
4.7g ___
4.7h ___
4.7i ___
4.7j ___
4.7k ___
4.7l ___
4.7m ___
4.7n ___
4.7o ___
5.1a ___

5.1b ___
5.2a ___
5.2b ___
6.1 ___
6.2a ___
6.2b ___
6.2c ___
6.2d ___
6.3 ___
6.4a ___
6.4b ___
6.4c ___
6.4d ___
6.5 ___
6.6a ___
6.6b ___
6.6c ___
6.6d ___
6.7 ___
6.8a ___
6.8b ___
6.8c ___
6.8d ___
6.8e ___
7. ___
8. ___
9.1 ___
9.2a ___
9.2b ___
9.3a ___
9.3b ___
10.1 ___
10.2 ___
10.3 ___

Send Responses to: RT-11 Wish List Survey
Multiware, Inc.
2121-B Second St. Suite 107
Davis, CA 95616

VAX Systems SIG Fall 1988 SIR Ballot

DECUS membership number____(six digits)

Our site uses the following VAX cpus (check all that apply)

11/730,11/725	□	11/750	□	11/780,785	□	6200 series	□
8500 series	□	8600 series	□	8700/8800	□	8200/8300 series	□
MicroVAX I,II	□	MicroVAX 2000, VAXstation 2000	□	MicroVAX 3000 series	□	Other:	

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:

Dave Schmidt
Management Science Associates
6565 Penn Avenue
Pittsburgh, PA 15206-4490

Reminder:
To be counted, your ballot must be received by August 31.

VAX Systems SIG

System Imporvement 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 (uese additional pages if required):

Tear out or photocopy reverse to submit an SIR.

Dave Schmidt
Management Science Associates
6565 Penn Avenue
Pittsburgh, PA 15206-4490
USA

DECUS

DECUS U.S. CHAPTER
SUBSCRIPTION SERVICE SIGS NEWSLETTERS ORDER FORM
(U.S. 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.
- Minimum of **$25.00** for orders placed via a credit card.
- 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 # ____________

Company ____

Address ____

____

City ______________________________ State ____________ Zip ________

Telephone # __(____)________________________

Subscription Service Offering	Unit Price	Quantity	Total
SIGs Newsletters	$35.00	________	____________
Spring '87 Proceedings (SP7)	15.00	________	____________
Fall '87 Proceedings (FA7)	15.00	________	____________
Spring '88 Proceedings (SP8)	15.00	________	____________
Fall '88 Proceedings (FA8)	15.00	________	____________
		Total Amount	$ ____________

□ MASTERCARD □ VISA □ DINERS CLUB/CARTE BLANCHE® □ AMERICAN EXPRESS

Credit Card # __ __ __ __ __ __ __ __ __ __ __ __ __ __ __ __ Expiration Date ____________

I understand that there will be no refunds even if I decide to cancel my subscription.

Signature ____

FOR DIGITAL EMPLOYEES ONLY

Badge # ______________________ Cost Center ______________________

Cost Center Mgr. Name ______________________ Cost Center Mgr. Signature ______________________

MAIL TO: Subscription Service, DECUS (BP02), 219 Boston Post Road, Marlboro, MA 01752-1850, (508) 480-3659. As of July 1, 1988 the phone number will be (508) 480-3659

FOR DECUS OFFICE ONLY

Check Number ______________________ Bank Number ______________________

Amount $ ______________________

DECUS

DECUS U.S. CHAPTER APPLICATION FOR MEMBERSHIP

□ New Membership □ Update to current membership profile Current DECUS Member.# ___ ___ ___ ___ ___ ___

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**

NOTE: Please print clearly or type!

Name: ____
(First) (Middle Initial) (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
2 □ SYMPOSIA
8 □ DECUS CHAPTER OFFICE
10 □ DIGITAL STORE

4 □ DIGITAL SALES
5 □ HARDWARE PACKAGE
6 □ SOFTWARE PACKAGE
12 □ ADVERTISING

13 □ LOCAL USERS GROUP
14 □ SPECIAL INTEREST GROUP
7 □ SOFTWARE DISPATCH (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
82 □ DECSYSTEM-10
83 □ DECSYSTEM-20

52 □ LSI-11
3 □ PDP-8 FAMILY
50 □ PDP-11 FAMILY

21 □ PROFESSIONAL
22 □ RAINBOW
54 □ VAX FAMILY

5 □ WPS-8
51 □ WPS-11

Major Operating Systems? Languages Used: Please Check Those Applicable To You.

1 □ ADA
2 □ ALGOL
5 □ APL
7 □ BASIC
17 □ BLISS
19 □ C
22 □ COBOL

26 □ CORAL-66
28 □ COS
34 □ DATATRIEVE
35 □ DBMS
38 □ DECNET
43 □ DIBOL
45 □ DOS-11

47 □ FOCAL
48 □ FORTRAN
51 □ GAMMA
110 □ IAS
53 □ IQL
58 □ MACRO
65 □ MUMPS

67 □ OS/8
68 □ PASCAL
72 □ PL-11
92 □ RPG
81 □ RSTS/E
83 □ RSX
91 □ RMS

109 □ RT-11
97 □ TECO
70 □ TOPS-10
71 □ TOPS-20
111 □ ULTRIX/UNIX
104 □ VMS
107 □ WPS-8

Type Of Business (Environment)/Computer Applications
Please Check That Which Best Describes Your Business/Application.

21 □ ACCOUNTANCY
7 □ BANK
64 □ BUSINESS/COMMERCIAL
74 □ BUSINESS/INFORMATION SYSTEMS
57 □ CHEMISTRY
54 □ CLINICAL LABORATORY
63 □ COMPUTATION
11 □ CONSUMER ELECTRONICS
18 □ CONSULTANT
72 □ DATA ACQUISITION
52 □ DATA COMMUNICATIONS
13 □ DATA PROCESSING SERVICES
71 □ DATA REDUCTION
17 □ DIGITAL EMPLOYEE-ENGINEERING
15 □ DIGITAL EMPLOYEE-MARKETING
16 □ DIGITAL EMPLOYEE-SERVICE GROUP
60 □ EDUCATIONAL ADMINISTRATION
1 □ EDUCATION/PRIMARY
2 □ EDUCATION/SECONDARY
61 □ EDUCATION-TECHNOLOGY
3 □ EDUCATION/UNIVERSITY
67 □ ENGINEERING
65 □ FINANCE/ACCOUNTING
77 □ GOVERNMENT
75 □ GRAPHICS
4 □ HOSPITAL
62 □ INDUSTRIAL
55 □ LABORATORY/SCIENTIFIC
14 □ LIBRARY
58 □ LIFE SCIENCES
70 □ MANUFACTURING
79 □ MARKETING
59 □ MEDICAL RESEARCH
6 □ MILITARY INSTALLATION
23 □ NUMERICAL CONTROL
68 □ OEM-COMMERCIAL
78 □ OEM-TECHNICAL
56 □ PHYSICAL SCIENCES
20 □ RESEARCH/DEVELOPMENT
10 □ RETAIL
73 □ SOFTWARE DEVELOPMENT
53 □ TELECOMMUNICATIONS
19 □ TELEPHONE/UTILITIES
51 □ TIMESHARING
80 □ TRAINING/INSTRUCTION
66 □ TYPESETTING/PUBLICATION

Special Interest Groups (SIGs) Enrollment
I Wish To Participate In The Following DECUS U. S. Chapter Special Interest Groups.

3 □ ARTIFICIAL INTELLIGENCE
7 □ BUSINESS APPLICATIONS
2 □ COMMERCIAL LANGUAGES
6 □ DATA MGMT. SYSTEMS
31 □ DAARC (LABS)
5 □ DATATRIEVE/4GL
8 □ EDUSIG
10 □ GRAPHICS APPLICATIONS
11 □ HARDWARE AND MICRO
35 □ IAS
27 □ LARGE SYSTEMS
16 □ L & T
14 □ MUMPS
15 □ NETWORKS
34 □ OFFICE AUTOMATION
36 □ PERSONAL COMPUTER
18 □ RSTS/E
17 □ RSX
19 □ RT-11
32 □ SITE MGMT. & TRNG
21 □ UNISIG
26 □ VAX

Job Title/Position - Please Check:

1 □ CORPORATE STAFF
2 □ DIVISION OR DEPARTMENT STAFF
3 □ SYSTEMS ANALYSIS
4 □ APPLICATIONS PROGRAMMING
5 □ SYSTEMS ANALYSIS/PROGRAMMING
6 □ OPERATING SYSTEM PROGRAMMING
7 □ DATABASE ADMINISTRATION
8 □ DATA COMMUNICATIONS/TELECOMMUNICATIONS
9 □ COMPUTER OPERATIONS
10 □ PRODUCTION CONTROL
101 □ CORPORATE DIRECTOR OF DP/MIS
102 □ ADMINISTRATIVE ASSISTANT
103 □ TECHNICAL ASSISTANT
104 □ SERVICES COORDINATOR
105 □ MANAGER
106 □ ANALYST
107 □ PROGRAMMER
108 □ DATABASE MANAGER
109 □ DATABASE ADMINISTRATOR
110 □ MANAGER OF DP OPERATIONS

Citizen of The United States of America? □ YES □ NO Country: ____________________

Signature: ____ Date: ____________

Forward To: DECUS U. S. Chapter
Digital Equipment Computer Users Society
Membership Processing Group
219 Boston Post Road, BP02
Marlboro, MA 01752-1850
Phone: (617)480-3418
DTN: 8-296-3418

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 01752-1850
USA

DECUS

DECUS SUBSCRIPTION SERVICE
DIGITAL EQUIPMENT COMPUTER SOCIETY
219 BOSTON POST ROAD, (BP02)
MARLBORO, MA 01752-1850