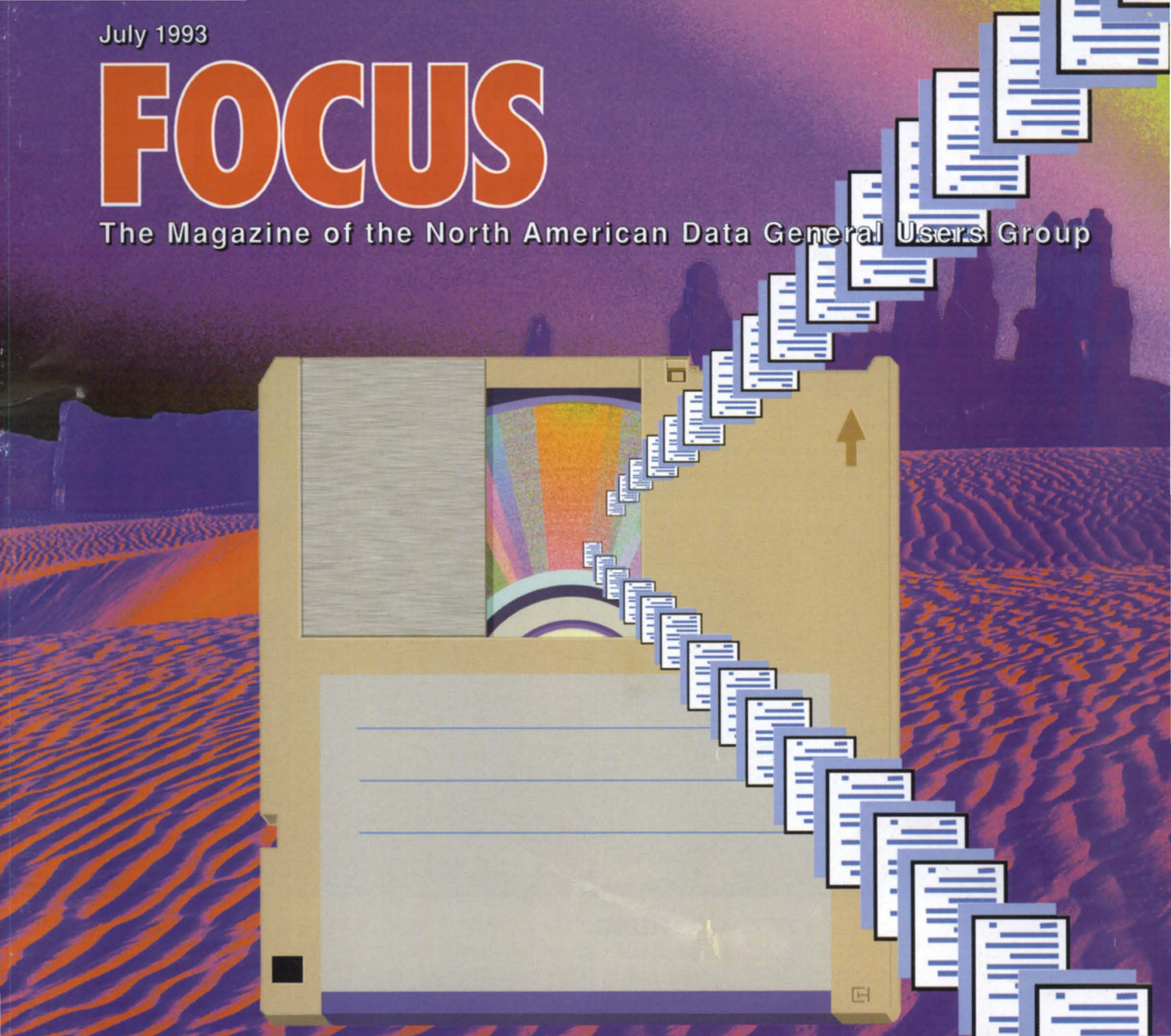
July 1993

FOCUS

The Magazine of the North American Data General Users Group



In Focus

The wide, wide world of document imaging
How DG solved imaging network bottlenecks
DG's "imaging" niche

Plus

A quick tour of the SQL language
Peaceful optimization
NADGUG news: A UK welcome

Bulk Rate
U.S. Postage
Paid
Permit No. 38
Fulton, MO

THE FASTEST WAY TO PUT 25 GB ON A SINGLE TAPE.

The CY-8500 can store up to 25 GB on a single tape, at speeds of up to 90 MB per minute, completely unattended.

How? With hardware data compression that can boost capacity by up to five times.

No other 8mm tape drive can match the capacity, speed, and price-performance of this best-selling drive.

ADVANCED. You'll never be left in the dark. A bright, backlit status display gives you the command under execution, transfer rate, compression ratio, amount of tape remaining and more.

Our *data compression* option is the fastest available. And it's switch selectable, so you can still read and write standard tapes. Locate and restore files quickly and easily with our *accelerated file access* option. Add *data encryption* to control backup and restore operations through the use of encoded card keys.

FLEXIBLE. The CY-8500 is plug compatible with virtually every computer system. Rack mounting options, dual drive configurations and a variety of cable lengths ensure a seamless fit into your computing environment. And as storage needs grow, you can upgrade to our ten tape library, capable of managing between 25 GB and 250 GB–all without manual intervention.

RELIABLE. A sophisticated Error Correction Code yields a bit error rate of less than one in 10^{17}–the best in the industry. A MTBF rate of 60,000 hours ensures reliability.

Each turnkey subsystem features a 12-month warranty that includes technical support from our experienced in-house engineering group.

Everything you want in a backup subsystem is available right now.

Call today for information at

(804) 873-9000

Circle 7 on reader service card.

TRUE PLUG COMPATIBILITY

Alliant	Motorola
Alpha Micro	NCR
Altos	NeXT
Apollo	Novell
Arix	OS/2
AT&T	PS/2
Basic-4	Parallel Port
Concurrent	PC 386/ix
Convergent	PC MS-DOS
DataGeneral	PC Xenix/ Unix
DEC SCSI	Pertec
DEC BI-Bus	PICK
DEC DSSI	Plexus
DEC HSC	Prime
DEC Q-Bus	Pyramid
DEC TU/TA81	Sequent
DEC Unibus	Silicon Graphics
Gould/Encore	STC
HP	Stratus
IBM AS/400	Sun
IBM Mainframe	Texas Instruments
IBM RISC/6000	Unisys
IBM RT	Ultimate
IBM S/38	Wang
ICL	*and more*
Intergraph	
Macintosh	
McDonnell Douglas	

CONTEMPORARY
CYBERNETICS

Rock Landing Corporate Center • 11846 Rock Landing
Newport News, Virginia 23606 • Fax (804)873-8836

VS COBOL to UNIX

VS COBOL Applications to UNIX in a Single Bound.

Running VS COBOL applications under UNIX is a problem. And losing your investment in those applications seems unavoidable. Until now.

Wild Hare now gives you a way to migrate your VS COBOL and INFOS applications to open systems in a single hop. You benefit from UNIX and other industry standards while saving time, money and effort.

Let us migrate your code and data to open systems. Your applications will use industry standards such as UNIX, DOS, and networks, and run on virtually any platform.

It's a simple leap to open systems. It protects your investment. And we can do it in a fraction of the time, for a fraction of the cost. So give us a call at 303-530-2221.

VS COBOL applications to UNIX. Yet another great reason to run with the Hare.

wild hare™
COMPUTER SYSTEMS INC.

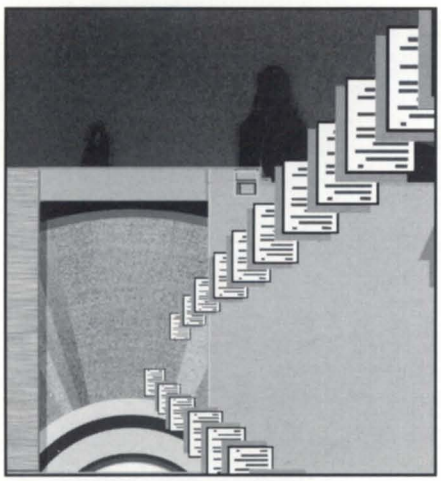
6595 O'Dell Place Boulder, Colorado 80301 USA Tel: 303-530-2221 Fax: 303-530-9637

WILD HARE CONSULTING

Circle 37 on reader service card.

FOCUS

The Magazine of the North American Data General Users Group



FOCUS ON: DOCUMENT IMAGING

The wide, wide world of document imaging
Data General's new AV Image product will attempt to lead the way in this burgeoning market
by Doug Johnson **10**

How Data General solved imaging network bottlenecks
by Jules H. Gilder **16**

DG's "imaging" niche
Our columnist perceives that Data General's new AV Image system achieves remarkable sophistication, but wonders anyway where this emergent document-imaging market will go. One thing seems certain, though: competitive pressures will force companies to move away from paper
by David Novy **20**

PRESIDENT'S PAGE
A UK welcome
by Jan Grossman **4**

EDITOR'S NOTE
Document imaging: You will
by Doug Johnson **6**

IN GENERAL
News and notes from the greater Data General community **8**

SYSTEM MANAGER'S LOG
:WFMOVE wrap-up
With endurance that awes us lesser folks, BJ pushes inexorably onward toward the light at the end of the :WFMOVE tunnel
by Brian Johnson **26**

AHEAD WITH RAD
A quick tour of the SQL language
Though you've probably heard of SQL (structured query language), perhaps you've never used it. It's easy to learn and difficult to master, but it's a giant step forward for productivity, reliability, and portability
by Kim Medlin **31**

SCREEN TEST
Peaceful optimization
Imagine the complexity of a disk optimizer written for a multiuser machine—files constantly created and deleted, multiple users accessing multiple records in the same file, Infos volumes. An optimizer peacefully coexisting with all this chaos? Not bloody likely, say you? Well, check out DISK_PAK
by Tim Boyer **22**

ON-LINE HELP
Who to call for information about NADGUG and Focus **32**

PRODUCTS AND SERVICES
The latest products for Data General systems **38**

BULLETIN BOARDS
DASH items from the Data General Customer Support Center; and Bits and Bytes from the Rational Data Systems NADGUG bulletin board **40**

Cover design by Casey Hunter

FOCUS, the Magazine of the North American Data General Users Group (ISSN 0883-8194) is the official publication of the North American Data General Users Group (NADGUG) in cooperation with Turnkey Publishing.

Editorial and business offices are at P.O. Box 200549, Austin, TX 78720, phone 512/335-2286. NADGUG headquarters are located at NADGUG, c/o Danieli & O'Keefe Associates, Inc., Chiswick Park, 490 Boston Post Rd., Sudbury, MA 01776, phone 508/443-3330.

FOCUS, the Magazine of the North American Data General Users Group is distributed to members of the North American Data General Users Group. Membership fees are $60 per person ($110 outside North America). A one-year (12 issues) subscription to *FOCUS, the Magazine of the North American Data General Users Group*, costs $48. For memberships and subscriptions outside the U.S., add $50 to defray the cost of mailing.

The cost of single copies is $4. Requests to replace missing issues free of charge are honored only up to six months after date of issue. Send request to *FOCUS, the Magazine of the North American Data General Users Group*, c/o Turnkey Publishing.

NADGUG is an independent association of computer users; it is not affiliated with Data General Corporation, nor does it represent the policies or opinions of Data General Corporation. The views expressed herein are the opinions of the authors, and do not necessarily represent the policies or opinions of NADGUG or of Turnkey Publishing, Inc.

Advertisements in *FOCUS, the Magazine of the North American Data General Users Group* do not constitute an endorsement of the advertised products by NADGUG or Turnkey Publishing, Inc.

Volume 9, Issue 7.

NORTH AMERICAN DATA GENERAL USERS GROUP


Cut the glaring high price of imaging with...
AV Image™
Imaging software starting at $695 per user.
View and zoom hundreds of images in rapid succession without decompression.
Superior functionality for scanning, document searching, viewing and annotation.
Works within your environment– MS-Windows and Motif clients with links to Oracle, Informix, Sybase, Ingres and Progress databases over any LAN– and best of all, works with your applications!
Image-enabling toolkit for existing applications.
"Other vendors are trying to figure out how to deliver open systems imaging. All they have to do is look at what DG is doing." Michael Goulde, Patricia Seybold Group April 23, 1993
To learn more about how AV Image can reflect your Image call 1-800-DATA GEN
Data General
Where the World is going™
© 1993 Data General Corporation
The aforementioned products are the trademarks of their respective vendors.

Jan Grossman

A UK welcome

It's about three months until the Atlanta NADGUG 93 conference, and everyone involved in its planning has been assigned their respective responsibilities. The members of the NADGUG Board of Directors are excited about the event and upcoming anniversary celebrations (DG's 25th, NADGUG's 20th). While musing about these activities I reflected upon springtime in London and the UK User Group's annual conference last March.

I was met at Gatwick Airport by Data General people who escorted me to their headquarters in London. I hadn't slept well on the flight, so my first reaction to British drivers and traffic was a sense of panic—they drive on the wrong side of the road!

Upon arriving at Data General UK I immediately noticed DG's logo prominently displayed on top of a clock tower in front of the building. Data General security is also tight in London, so I ended up waiting a few minutes for a badge (this gave me time to wipe the sand out of my eyes and regain my composure).

I was hosted on a tour of the offices by Michael Skehin, who pointed out that this headquarters contains sales, customer services, field engineering, marketing, and DG Direct. It's equivalent to Westboro and Atlanta.

Michael and I then journeyed to Birmingham, about one and a half hours from London. The conference site was quite busy; I wondered if it was because of all those British Data General users, but I was wrong—there was a conference going on for Avon representatives! I retreated to my room where I enjoyed a leisurely cup of hot chocolate—I could have had tea; there was an abundant supply in all hotel rooms in England—and managed to stay awake until Dennis Doyle arrived from London about 8 p.m.

Day one of the UK Conference was exciting. For once I could be a typical conference attendee and be free to attend all the sessions (but then again, how could I be typical when everyone around spoke with foreign accents . . . or was *I* the foreigner?). Tea (coffee for me) was served in the registration and exhibit areas, which meant that we became well acquainted with the exhibitors. The energetic Wordperfect people were there giving away hats, shirts, and other goodies, just like at NADGUG 92 in Kansas City, only they were represented by Marco van Bemmelen from the Netherlands. Also present were representatives from Cyberscience, CSS Software Solutions, Data General Limited, Data General Direct, Matrix Logic, Transoft Ltd., and Unique UK. I managed to fill two DG Direct bags with product literature and trinkets.

Paul Hewton, Data General User Group chairman, led the day and introduced Brian Johnson (*Focus* magazine's "BJ"), who gave his personal overview of Data General's 25 years. Peat Marwick presented its predictions for the next quarter-century of the IT industry, and a there was a lively lecture about computer viruses from Dr. Alan Solomon.

Terry Radford, Data General Limited general manager, gave a status report of DG activities, followed by John Coon's Data General Marketing

Continued on page 33

FOCUS
The Magazine of the North American Data General Users Group

At last – INFOS® on UNIX®!

Transoft is proud to announce U/FOS™ – The UNIX implementation of INFOS you've been waiting for

- smooth, rapid migration of your COBOL, Fortran and PL/I applications
- full INFOS functionality
- automatic data conversion
- choice of major UNIXs
- improved functionality and better utilities
- API for third party software products

Huge savings on redevelopment and retraining costs.

With seven years experience, Transoft are experts in software portability and are a leading international supplier of Open Systems tools to the Data General community. Our UBB Universal Business Basic product – nearly 5,000 copies sold to date – has enabled DG users worldwide to migrate to UNIX. And our powerful U/SQL (Universal Structured Query Language) operates on a variety of COBOL, BASIC and RDBMS file types.

With U/FOS you can achieve not only a smooth, rapid migration of your INFOS applications to UNIX, but also powerful additional features including better data compression, better checkpointing and RDBMS-level data integrity and recovery. SQL reporting is also planned, using our proven U/SQL product.

Act now – get the financial and operational benefits of Open Systems – U/FOS improved functionality – RDBMS-level transaction security – and make massive financial savings on redevelopment and training.

To obtain further details and arrange a personal demonstration, please contact:
TRANSOFT INC, 1100 Circle 75 Parkway, Suite 800, Atlanta, GA 30339, USA.
Tel: (404) 933 1661 Fax: (404) 933 1662
TRANSOFT LTD, Nash House, Datchet Road, Slough, SL3 7LR, England.
Tel: 0753 692332 (Int + 44 753 692332)
Fax: 0753 694251 (Int + 44 753 694251)

TRANSOFT®
Portability and Productivity

Software Tools for Your Trade

Document imaging: you will

by Doug Johnson • Focus staff

On my spare Saturdays I frequent the noted LBJ Library on the campus of the University of Texas at Austin. Tourists enter the big, picturesque limestone building to see exhibits about former president Lyndon Johnson, but I go to the nearby Barker Center for American History to pursue my secret passion: digging up interesting historical tidbits in old newspapers. There's quite an extensive microfilm collection, all kinds of great stuff.

I load a reel on a microfilm reader, fire up my computer to take notes, and get started cranking backward in time. It's a fascinating way to spend an afternoon, but there's no automation to the process. It's difficult to be systematic. Certainly there's no helpful index. I may know what year I'm searching, and sometimes I even have a specific date, but most of the time the search requires manual cranking—frame after frame, page after page. My ability to find what I'm looking for is limited, therefore, by the endurance of my nearsighted eyes and my fallibly human attention span.

I'd like to be able to do this:

Call up all articles published in the *San Antonio Daily Express* during calendar year 1911 that mention Mexican dictator Porfirio Díaz and/or rebel leader Francisco I. Madero. I want to page through those articles on screen, print a summary report of what's there, and transfer copies of selected files to my computer for future reference and further processing.

Or at the very least, I wish I could tell the #!@*&% machine to go instantly to Sunday, June 4, 1911, page 4, top of column 2. Even those relatively simple capabilities would save me mountains of time.

But alas, that microfilmed newspaper frame I need to find today is cut off from any kind of relational data base or other organizational system. It's not connected to any computer at all. That tidbit I'm searching for is as lonely and remote as a mislaid memo in a musty storeroom filing cabinet.

So what does all this have to do with document imaging, our "Focus on:" topic for July? Well, I like to think that someday in the whizbang Utopian future, those quick and efficient searches I want (and much more) will be available through incredibly sophisticated but easy-to-use electronic store-and-retrieve systems. Such a system for archiving and indexing newspapers would begin with today's *New York Times* and *Wall Street Journal*, and then maybe the Clinton Administration's economic-stimulus program can hire an army of clerical staff to scan and organize all those millions of microfilm frames.

Actually, document imaging is already happening, and on an impressively wide scale. The ability to connect a data base system with scanned images of related documents will find plenty of opportunities to help out in legal firms, healthcare recordkeeping, banking transactions—anywhere you find proliferating documents, and that's just about everywhere, really. And the truly paperless office? Well, maybe someday, maybe.

I'm reminded of some slick AT&T television commercials airing recently, in which Tom Selleck's voice invites you to imagine such futuristic wonders as sending a fax from the beach (pen-based, wireless file transfers).

Me? I'll be doing *that*?

"You will," Selleck's voice promises. And I believe him. Sign me up.

The same goes for document imaging. You will. Δ

Complete Automated Backup

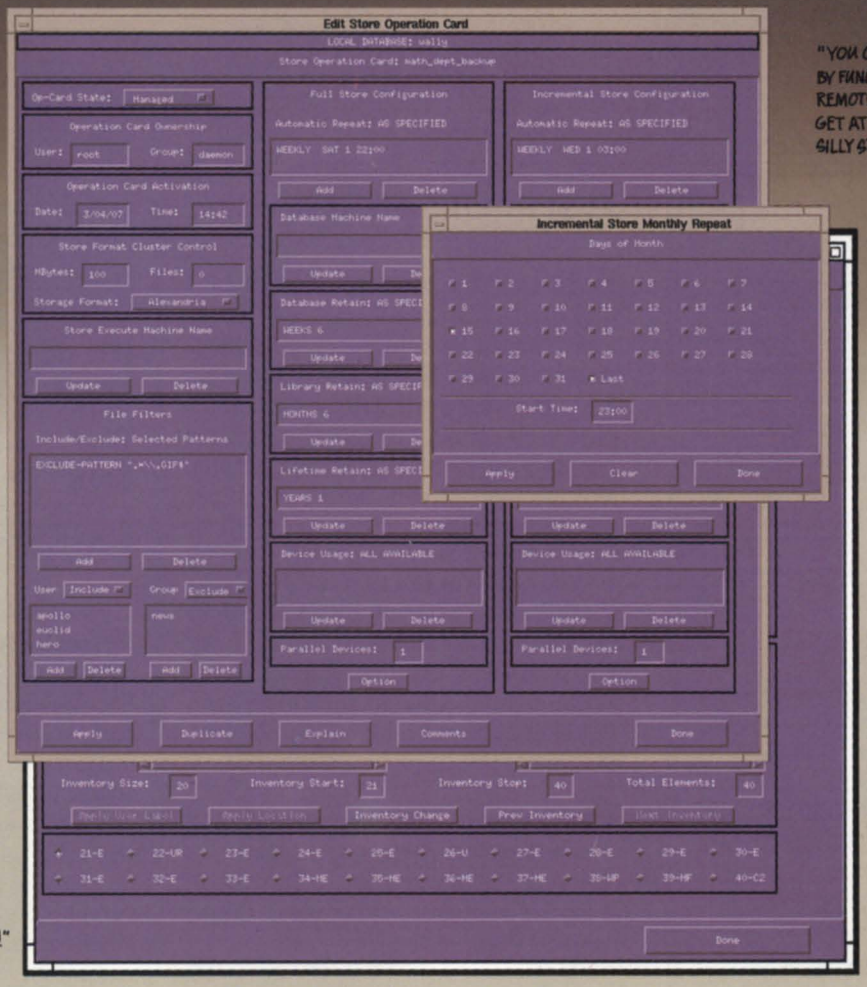
THE POWERFUL SCHEDULER COORDINATES FULL AND INCREMENTAL STORES FOR EVERY SERVER AND WORKSTATION.

ALEXANDRIA INTEGRATES WITH OTHER UTILITIES TO BACK UP COMMERCIAL DATABASES OFF RAW DISK PARTITIONS.

"YOU CAN INCLUDE OR EXCLUDE FILES BY REGULAR EXPRESSION. LET'S TRY 'exclude .*\.GIF$' -THAT'LL SAVE TAPE!"

"YEAH, I KNOW YOU'RE BACKING UP HP's TO RS/6000s AND SUNS TO AN SGI. RELAX, THIS IS THE '90s!"

"C.P. STAR IS TAKING ME OUT TO LUNCH FOR RESTORING HER FACEPLATE DESIGN. DON'T TELL HER SHE CAN RESTORE HER OWN FILES RIGHT FROM HER DESK, I'M STARVING!"



"YOU CAN DEFINE PERMISSIONS BY FUNCTION, USER, GROUP OR REMOTE MACHINE. NO, HE CAN'T GET AT YOUR BACKUPS... WELL, YES, SILLY STRING IS HARD TO GET OUT."

PRIVILEGES PERMITTING, ANY ALEXANDRIA CLIENT CAN BACK UP TO ANY ALEXANDRIA LIBRARY OR SERVER ON THE NETWORK.

"NO PROBLEM! I CAN CONFIGURE YOUR BACKUPS RIGHT FROM MY DESK. SOME OTHER TIME WE'LL SEE YOUR GRAND CANYON PICTURES..."

ALEXANDRIA'S BUILT-IN DATABASE MANAGES DATA, MEDIA AND DEVICES IN LIBRARIES. IT WAS DEVELOPED SPECIFICALLY FOR USE WITH AUTOMATED LIBRARIES.

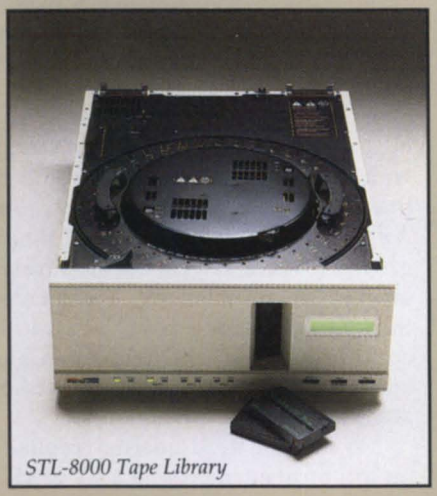
"LAST YEAR'S NCAA POOL? YES, THAT'S IN THE VAULT. ASK KATE FOR THIS TAPE."

"TWO MORE PASSES LEFT ON THAT CLEANING CARTRIDGE."

Alexandria Backup Librarian Software

SPECTRA STORAGE SOLUTION

Hardware and software, from one vendor, to provide complete, client/server backup for heterogeneous UNIX networks.



STL-8000 Tape Library

THE SOFTWARE

Alexandria Backup Librarian was developed specifically for use with automated libraries. Its built-in, relational database manages data, media and libraries across the network.

Alexandria lets you define backup strategies, security needs, media and database retention schedules — even where Alexandria's database information is stored.

Alexandria executes backups according to your schedules. You will be alerted by e-mail if Alexandria needs more media, when tapes can be moved to the vault or if any error conditions are encountered.

Alexandria is ideal for heterogeneous networks, supporting SunSPARC, SGI, RS/6000, HP9000, SCO UNIX, 88Open, Sequent, DG AViiON, Motorola, Auspex and Unisys U6000 clients and servers.

THE HARDWARE

Spectra Logic's STL-8000 Automated Tape Library provides random access to 200 Gigabytes of uncompressed data on forty 8mm tape cartridges. Centralize your data in a 7" height, rack-mount or desktop enclosure.

Alexandria also supports libraries from Exabyte, Hewlett-Packard and ADIC as well as Spectra Logic's STL-6000: a 20 cartridge, 8mm carousel tape library.

THE COMPLETE STORY

Call Eagle Software at **(800) 477-5432** for complete information on the Spectra Storage Solution.

EAGLE SOFTWARE, INC.

EAGLE Software, Inc.
123 Indiana Ave/PO Box 16
Salina, KS 67402-0016
Phone (913) 823-7257
Fax (913) 823-6185
Toll Free: (800) 477-5432

RS/6000 Sun SGI SCO DG AViiON Sequent HP9000 Motorola

Circle 23 on reader service card.

News Briefs

▶ **Cyborg Systems, Inc.**, and Data General will jointly market The Solution Series human resource management system on Aviions.

▶ **Groupe Bull** is bullish on Data General Clariion storage systems. Bull will offer DG's state-of-the-art disk arrays in connection with Bull's line of Unix systems and servers. In addition, the two companies will investigate areas of cooperation on future-generation storage products. Groupe Bull is one of the world's largest computer companies, and has a strong presence in Europe.

▶ Data General's Special Systems Division was recommended for ISO 9000 quality certification by Underwriter's Laboratories. ISO certification is an international standard for product design, manufacturing, and service. DG's Special Systems Division, based in the company's Westboro headquarters, is responsible for designing custom hardware.

▶ Data General now offers a PICK migration/porting service to help customers move existing PICK applications to the Aviion open systems platform. If you want to move your PICK applications to uniVerse/VMARK, DG/Advanced PICK, UNIDATA, or Prime Information/OPEN (PI/open) PICK-based implementations, call your DG sales rep or systems engineer, or your nearest Professional Services group representative at 404-705-2500 in Atlanta; 714-724-3900 in Irvine, California; or 201-587-8700 in Saddlebrook, New Jersey.

In General is compiled by Robin Perry. If you have an item for In General, please send it to Robin, c/o Focus magazine, P.O. Box 200549, Austin, TX 78720; 512/335-2286; Fax: 512/335-3083.

NADGUG 93

NADGUG 93 will be an historic event! The annual conference of NADGUG members, to be held October 25-28 in Atlanta, marks NADGUG's 20th year and Data General's 25th. Once again, NADGUG's meeting will be held in conjunction with Data General's Sales and Systems Kickoff and DG's annual reseller meeting.

Technical seminars, open systems training, and more than 55 breakout sessions will offer many educational opportunites for attendees.

NADGUG and Data General Educational Services are sponsoring nine half-day technical seminars throughout the conference. Topics include

- Networking technologies and how they apply to customer applications
- A technical introduction to DG/UX
- Simple Network Management Protocol (SNMP)
- Netware vs. TCP/IP for PC connectivity—which is the right choice?
- Information technology acquisition support methodology
- Domain name system implementation
- Data base performance tuning on Aviion and DG/UX
- The DG/UX filing system.

In addition, two full-day executive seminars conducted by the Open Systems Training company will be available to conference attendees on Tuesday and Wednesday. Those who register for the full conference may participate in two half-day technical seminars or one full-day executive seminar at no additional cost.

"From the seminars and breakouts to the exhibits, NADGUG 93 has so much to offer attendees. The training value alone is worth the price of admission" said Calvin Durden, conference committee chairman and vice president of Tractor & Equipment Co. in Birmingham, Alabama.

To make sure you receive a conference registration kit, call or fax your name, company, address, and telephone number to NADGUG: c/o Danieli & O'Keefe, at telephone: 800-932-6663 (U.S. and Canada only) or 508-443-3330; fax: 508-443-4715. Registration kits will be mailed later this month.

Stay tuned to *Focus* for further news about NADGUG 93 and NADGUG's 20th birthday bash.

Earnings watch

Data General reported a net loss of $7.6 million for its second quarter of fiscal year 1993, which ended March 27. For the second quarter last year, the company reported a net loss of $55.3 million, including a restructuring charge of $48 million.

Revenues for the second quarter were $267.4 million, compared with revenues of $273.8 million for the comparable quarter last year. Operating loss for the second quarter was $4.7 million, compared with an operating loss of $53.7 million, including the restructuring charge, for the second quarter last year. Revenues for the first two quarters of 1993 were $547.1 million, compared with $568.6 million for the same period last year.

Data General President and Chief Executive Officer Ronald L. Skates said, "We are not satisfied with the overall results, primarily in the revenue area, but we are encouraged by the continued success of our Aviion family of open systems computers. Despite having to compete in a very difficult worldwide economy and under very competitive pricing conditions, revenues for Aviion again showed significant growth over last year's comparable quarter." In addition, Mr. Skates noted that "Data General's financial position continues to be solid with cash and marketable securities of $196.2 million at the end of the second quarter."

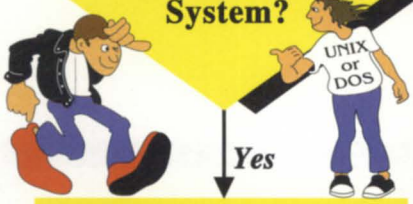
Watch for third quarter results to be announced near the end of this month. Δ

ATTENTION CEO USERS!

EAGLE Software Has Got You Covered!

Yes

Moving to Another Operating System? — No



Yes

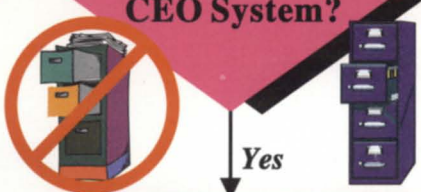
CABINET_CONVERTER™

New!

Migrate CEO Files to UNIX® or DOS®

- Allows mass conversion of CEO files to any word processor or spreadsheet supported by CEO_DOC_CONVERT or KEYpak.
- Moves information stored in the CEO document summary to the summary fields in WordPerfect.™
- Perform one mass migration or migrate in stages to meet users' needs.
- Flexibility allows customization to meet your specific migration needs.

Need to Optimize Your CEO System? — No



Yes

MAID™ for CEO®

CEO Optimization Utilities

- Recovers wasted disk space by the tens of thousands of blocks.
- Provides full-time notification of new CEO mail and reminders even if you're not in CEO.
- Helps avoid document loss and keeps CEO_INDEX errors from corrupting your files.
- Allows operations on the CEO filing system not normally available through CEO.
- Supercharges the performance of the CEO mail system.

Tight On Disk Space? — No

Yes

DOCtAR™

New!

DOCument ARchiver and Compressor for CEO

- Compresses and archives CEO documents up to 70% to any directory.
- Allows immediate restores by CEO users as a CEO user application or CLI program.
- Runs in conjunction with, or in place of, the CEO_ARCHIVER.
- Includes all the compression software you need at one low price.

Also from EAGLE...

FILE_MONITOR™
A superior file usage monitor and disk balancer.

DISK_PAK™ OnLine!™
Optimize your disks without shutting down your MV.

VS_TOOLBOX™
Powerful tools for MV management, optimizing INFOS®, and system security.

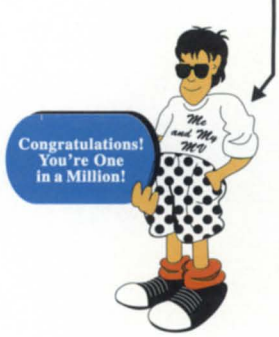
RUSH™
Now you can FAX documents directly from your MV terminals!

DISK_PAK™
The premier disk optimization product for MVs.

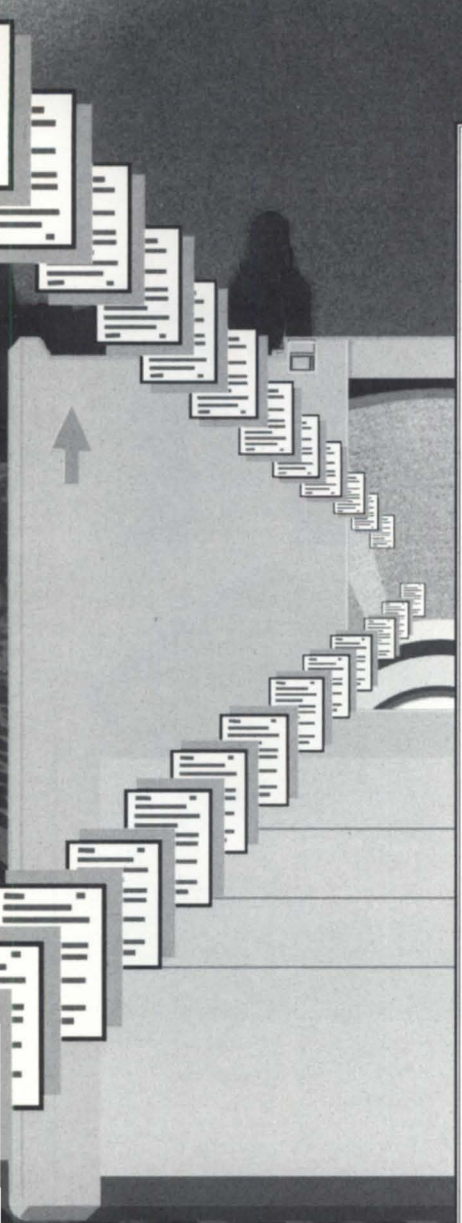
Call Today!

1-800-477-5432

EAGLE Software, Inc.
123 Indiana Avenue
Salina, KS 67401
Phone: 913-823-7257
FAX: 913-823-6185



Circle 17 on reader service card.



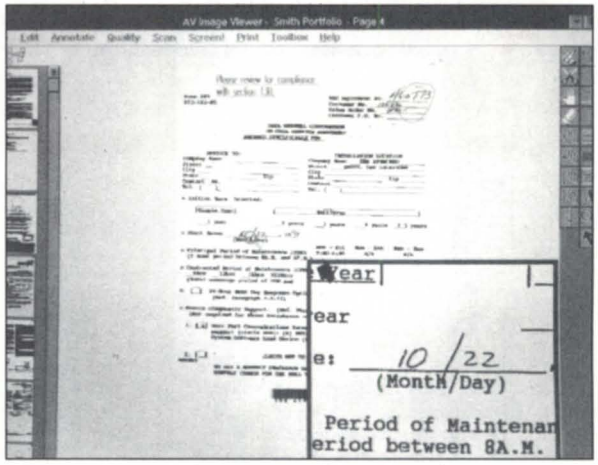
The wide, wide world of document imaging

SYNOPSIS

The intricacies and awesome capabilities of document imaging coupled with powerful data base management systems may be summarized in the following flippant oversimplification: you can look stuff up faster. But there's much more, and Data General's new AV Image product will attempt to lead the way in this burgeoning market.

by Doug Johnson
Focus staff

Someday in that whiz-bang Utopian future perhaps we will, finally, achieve the heretofore mythical paperless office. That *is* the goal, isn't it? ("Yeah, and while we're at it let's discover the famed Seven Cities of Cíbola," growl the cynical naysayers.) Maybe paperlessness will happen because we will have run out of paper to write on and chopped down the last tree to produce the paper. But just maybe Paperless World will succeed because we will at last have found a way to do everything electronically on our incredibly smart, wonderfully organized, stupendously powerful, and easily portable computer systems: keyboard-based, pen-based voice-recognition-based, wireless-file-transfer-based, and maybe even direct-human-mind-link-based. Read some science fiction if you want the dazzling details.



AV Image on screen (photo courtesy of Data General)

But even in the Utopian future we will stand at the summit of the Giant Cosmic Landfill and look back at a history littered and piled with countless quintillions of paper documents that somebody sometime stored in some form so that somebody else for some reason could look them up again someday. And then we can take a stab at the Big Question: Can we find that vital document again?

Search times for a particular document seem to have increased most appallingly. To a large extent computer technology succeeded only in replacing overstuffed filing cabinets with overstuffed hard disks, tape drives, and optical platters. A large company can produce literally millions of pages per year. The answer will have something to do with document imaging and advanced electronic document management, and other stuff no doubt resplendent with plentiful multisyllabic jargon. Document imaging is one component of that overall process called *workflow management*, but what's done with images may be what drives workflow management's future success.

Looking stuff up faster

The intricacies and awesome capabilities of document imaging coupled with powerful data base management systems may be summarized in the following flippant oversimplification: *you can look stuff up faster*. Certainly that's something we'd all like to do. It'd be like slurping all those pieces of paper into the computer system, preserving their original appearance (signatures at the bottom of correspondence and all).

Document imaging itself, as our language has evolved to describe it, is nothing new. All "imaging" means, really, is taking a picture of something. Webster's dictionary defines *imaging* as "to create a representation of. Every time you punch the "copy" button on your trusty Xerox machine you're engaging in simple document-imaging activity. More recently, we've added scanners to the office equipment fleet, and the resulting huge data files often have created more problems than they solved, especially when working with such things across networks.

Document imaging is rapidly becoming just like the rest of the computer industry—a wild profusion of products, and a chaotic battlefield of bloodied competitors. In its April 5 issue *Computerworld* did an extensive assessment of imaging as a coming thing. Already there are nearly 100 imaging products out there to choose from, ranging from the extremely expensive to the cheap-and-pretty-much-worthless. "Cost-effective imaging? Perhaps. Low-cost? Probably not," went the headline. The major considerations appeared to be price, scalability, customizability, and ease of integration with existing systems and applications.

There was no mention of any imaging product forthcoming from Data General Corporation, but it wasn't long after the *Computerworld* report that DG announced AV Image, its entry into the escalating race. The product has been scheduled for late-June release.

"We've done an awful lot of thinking about how to deal with document imaging," said William Zastrow, DG's vice president of imaging and office systems (a new Data General Business Unit). "And what we've found is that with document imaging people have

been predicting its leap forward in being a commodity, and really taking off, for a long time. And the reason we think that imaging has *not* taken off is because it's really too expensive, too complex, and takes too long to implement."

AV Image

DG offers AV Image as a flexible solution for paper-swamped office applications such as government, law, or financial services environments—anywhere you do document archiving and records or forms management. AV Image promises capabilities comparable to the traditionally high-end imaging solutions but at a much more pleasant-to-consider price. *Computerworld's* report pointed out that the most confusion and activity in imaging products is occurring in the lower price range.

AV Image is a new concept because it offers high-end imaging for a commodity price, Data General claims. "It's essentially the same price that you would have for a word processor or a spreadsheet," said Zastrow. "In small quantities it's $695 per user, and it goes down with larger quantities."

AV Image supports open systems, allowing users to choose from a variety of desktop clients, networks, and data bases. And it integrates with existing business applications. "So what we have is the capability to not only provide a robust document management system, with imaging, but we also have the capability to image-enable existing applications, regardless of the platform," said Zastrow. "And that includes all of the MV applications running on Infos, or any other character-based application that they may have."

The standard package consists of the WIN-Track Document Manager and the AV Image Viewer. A pair of options are available: DB Links and the AV Imagizer Toolkit. Based on SQL [Structured Query Language—"sequel,"] client-server architecture, AV Image runs on Microsoft Windows, PCs, OSF/Motif workstations, or X-terminals networked to a relational data base on Unix-based Aviion server. It runs over Novell Netware or TCP/IP networks, and through the DB Link option supports such data bases as Sybase, Oracle, Ingres, Informix, and Progress.

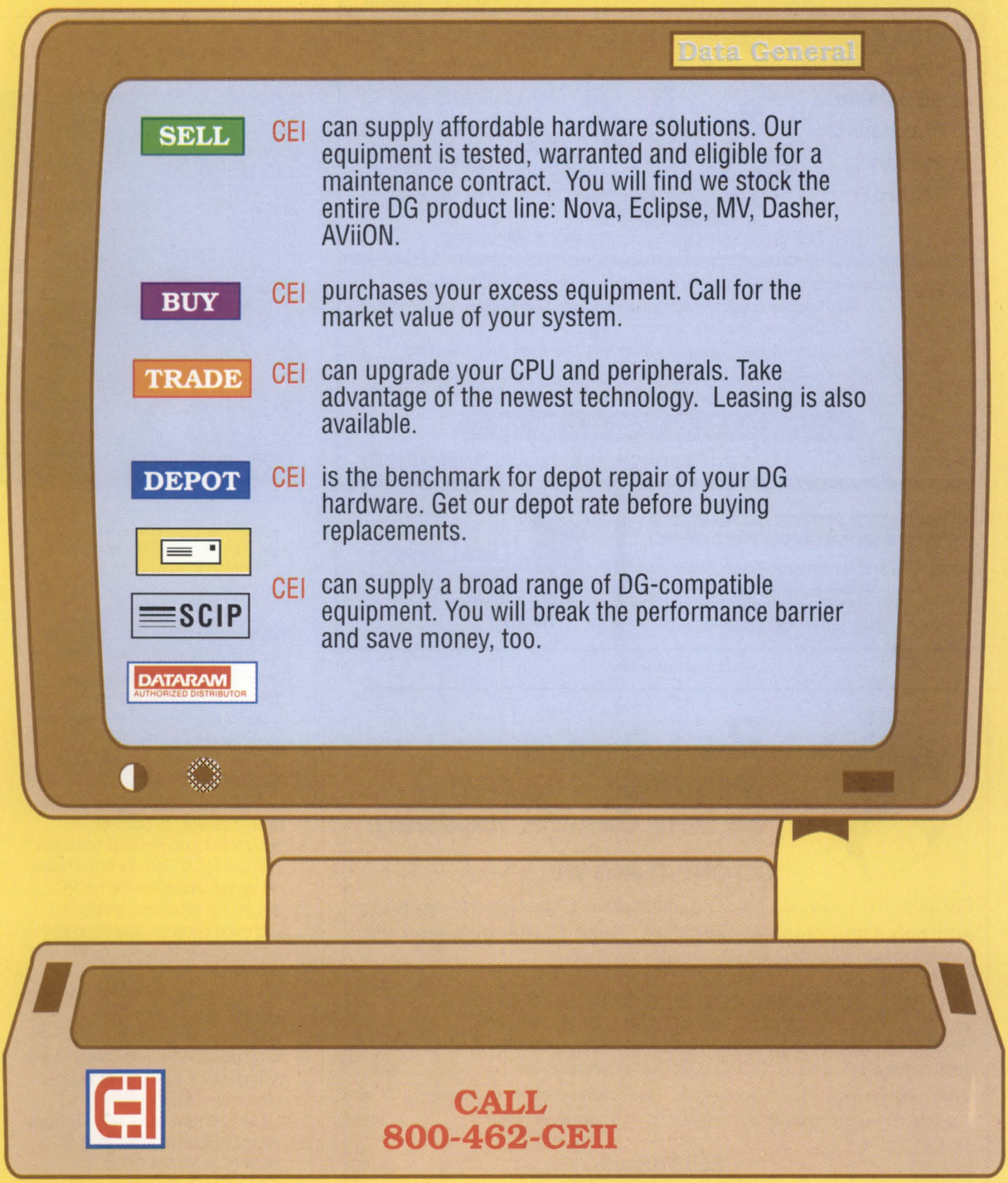
WIN-Track is a Windows-based, universal document manager for document images or other application objects. Users can create, scan, index, search, edit, delete, mail, and print documents and images.

The AV Image Viewer allows users to view the imaged document. "And we have a unique system that allows you to view literally dozens of documents, with hundreds of images, all at the same time in rapid succession, and browse through them at 5 or 6 pages per minute," said Zastrow.

File compression

DG claims that its unique combination of image overview files and CCITT Group 4 Tiled compression sets AV Image apart from other products by

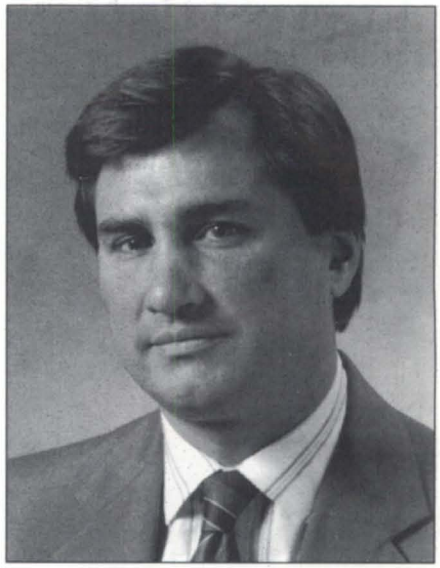
Continued on page 18



Computer Engineering International

2231 Star Court
P.O. Box 81755
Rochester, MI 48308
MI Phone: 313-853-0770
FAX: 313-853-0775

Circle 5 on reader service card.



William Zastrow—Vice President, Imaging and Office Systems, with Data General Corporation

William Zastrow is responsible for Data General's Office Systems Group, the Imaging Systems and Optical Storage Products Group, the Communications and Networking Products Group, and the Electronic Mail Systems Group. He came to Data General from Dell Computer Corporation, where he was director of advanced systems and software marketing. At Dell, Zastrow headed up product planning for PC-based workstations, graphics, high-end servers, operating systems, disk arrays, and networking products.

Prior to Dell, he worked for ICL North America for more than seven years in various director-level positions: international marketing, office systems product planning, VAR marketing and OEM software sales. ICL Designs and markets Unix-based RISC servers and multiuser office-automation software.

Zastrow earned an ASEE degree from St. Paul Technical Institute in St. Paul, Minnesota. Δ

AV Image in business

(Excerpts from an interview with DG's William Zastrow)

Where there's lots of information online

"We use the tag, 'It's 1993: Do you know where your documents are?' People are spending an awful lot of time trying to figure out where they have their documents. People in legal departments, for instance, [are] looking for precedents, looking for case histories, things like that. In banking, looking up customer credit information; all kinds of different things in businesses. But then again, in offices just look around you; you probably have piles of paper all over if you're like all of us. And white-collar workers, whose costs are increasing continuously, are being saddled with the task of gathering up and trying to find all this information to do their jobs.

"So that's why this document management is extremely important. And it's not good enough just to have a file system, a hierarchical file system. A hierarchical file system assumes that you know something about what was put in there. And [in] larger workgroup systems or enterprise systems, the documents will be indexed by people you may not even know. You don't know how they're going to index it. So you have to have a global system that uses a common set of keywords and a common set of conventions in order to store and recall documents. Those common conventions would be things like document titles, dates, authors, subjects, maybe customer numbers, account numbers, that sort of thing. And then you set up a keyword dictionary to define the attributes, and let the people pick those attributes that match up with that document. Then you have a much better shot at finding it again [even] if you are not the author.

Use your data base

"Now that's all fine and good, but in order to utilize it, you need to store all of these records in some sort of a data base. What people hate is being told what data base they have to use. And if they've already got one, they abhor having to buy another one. So what we do is, we provide an open system that allows our document-management system front end to tie in to anybody's data base. Essentially we can use Oracle, Informix, Ingres, Progress, Sybase, Paradox. And we can store files in any of those, and the front ends all look the same. If you've already got the data base you don't need to buy another one. We also support just about any type of network, whether it be TCP/IP-based, or LAN Manager, or Netware. We also support all the different desktop environments that are common in an office. We support MS Windows PCs, we support Motif workstations of all types, we support X-terminals, and on your Macintosh we could run Mac X and put it in a finder window. So it's very open." Δ

How Data General solved imaging network bottlenecks

by Jules H. Gilder
Imaging Magazine

There are three critical bottlenecks that degrade performance in imaging networks: disk I/O (input/output), network bandwidth, and PC memory capacity. Data General (800/DATA-GEN) has built common-sense solutions to all three into its new document-imaging product, called AV Image.

An 8.5-x-11-inch page scanned at 300 dpi [dots per inch] results in a 1 MB file, compressed to about 50 KB on disk. Retrieving a 50-page document stored on a server will cause major network havoc, even if it's compressed. Worse yet, you probably won't have enough client (your PC) memory to decompress the entire document and view it.

Data General discovered a little secret: *In most cases you don't need all the data just to view an image—your computer throws most of it away anyway.* Data General says that for screen resolutions as high as 1024 x 768, a 5 KB file will show you everything you would normally see in the 1 MB file. The secret? They use CCITT Group IV Tiled Compression, and screen snapshots they call "screen overview files."

Look at the numbers. A Super VGA screen with a resolution of 1024 x 768 displays a maximum 786,432 bits of information. Since there are eight bits in a byte, that equals 98,304 bytes of data. Even if the document took up the entire screen—which it can't, because the image is vertical, and the monitor is horizontal—it would use only one-tenth of the original data. But that's not where it ends. Group IV compression can shrink data 20 to 1. Apply that to the bytes available on the 1024 x 768 screen. The result is a 4,916-byte file to view the document.

There's also a substantial speed advantage. A regular compressed file of an 8.5-x-11-inch page takes 7 to 10 seconds to decompress. With AV Image, you can view 100 full-screen zoomed images in less than 17 seconds.

AV Image is based on an SQL [structured query language] client/server architecture and runs on PCs using MS Windows, OSF/Motif workstations, and X-terminals networked to a relational data base on a Data General Aviion Unix-based server. It runs on Novell Netware and TCP/IP networks. An optional program called DB Link adds support

for Sybase, Oracle, Ingres, Informix, or Progress data bases. For small networks with less than 20 users, AV Image includes a Paradox PC data base engine with networked multiuser tile management on the Aviion server.

AV Image has a "browser" with miniature versions of images in a scrolling window. It also has multiple zoom modes including dynamic zoom, pop-up zoom (variable zoom from 1:1 to 8:1), and a scale-to-window feature. To speed display even more, AV Image only decompresses the portions of images you're looking at. It accepts many image formats: TIFF, CALS, PCX, BMP, IMG, RLE, CCITT Group III, Group-IV Tiled and Group-IV Striped.

Images can also be rotated, cropped, de-skewed, and have barcodes added to them. To use images in another application, you move them to the Windows clipboard, then "paste" them into your word processor, or spreadsheet, or whatever.

Annotation features add value

Each image can have up to 255 layers of annotation, each in a different color to identify the author. You can prevent or allow changes to each layer.

There are three ways to add annotated text: You can type it on a sticky note; you can enter it in a temporary box on the screen, then transfer it to the image overlay; or you can type text directly onto an overlay. In all cases, you can use any font available in Windows, at any size, and have the text go in any direction. Images can be dynamically linked to other images or files in the data base, and those links can be linked to still more files or images. The program makes heavy use of Windows' OLE features.

AV Image also contains WIN-Track, which is a Windows-based universal document manager that handles image documents or other application objects (e.g., spreadsheets, text files, etc.). With it, you create, scan, index, delete, search, edit, mail, and print image documents. WIN-Track supports up to 20 separate data bases or file cabinets, each with its own user-defined document types, field configurations, keyword dictionary, security levels, and functional permissions. Documents can be searched by fixed fields, user-defined fields, wildcard substrings, or Boolean operators. Queries can be stored to reuse later.

AV Image, including WIN-Track, costs $695 per client/server user. DB Link for any of the five data bases it supports is an extra $300 per client/server user. The minimum-size network is five users. There are price breaks for larger configurations. Δ

Reprinted with permission from IMAGING Magazine, 12 West 21st Street, New York, NY 10010; 212/691-8215.

Continued from page 12

solving the disk access, network bandwidth, and PC memory contention problems commonly associated with other imaging systems. Whole images are not required for viewing or zooming. (See related article, page. 16.)

"We're only moving very small 5 K files across the network, where everyone else is moving 50 K images across the network," said Zastrow. "That's a 10-to-1 reduction in the amount of disk access required. It's a 10-to-1 reduction in the amount of network bandwidth required. And it's a 50-to-1 reduction in the amount of PC network memory that's required."

Annotation and routing

Document routing is the next step beyond mere imaging in attaining the ultimate goal of sophisticated workflow management. "Document routing really makes a big difference now, because now you can have electronic soft copies of images," said Zastrow.

If a draft document requires review by a number of people, they can do it in parallel with AV Image, eliminating paper photocopies. Sophisticated annotation features allow users to add notes and drawings on each image in a setup that offers up to 255 overlays. Annotations appear in different colors. "But this doesn't change the underlying document," said Zastrow.

Developers Toolkit

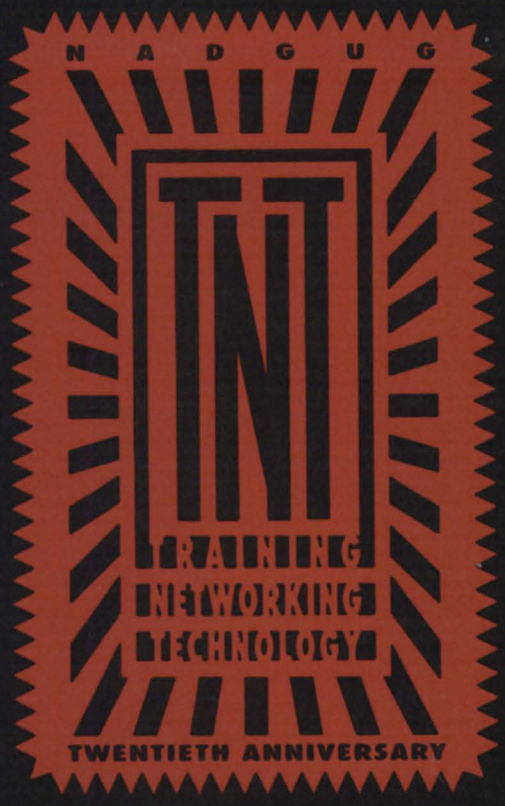
The AV Imagizer Toolkit is a programmer's package that includes runtime C-language and dynamic-link libraries, as well as documentation for VARs (value-added resellers) and system integrators to help with modifying and "image-enabling" existing applications.

"What we in fact are doing is creating image-enabled applications from legacy applications," said Zastrow. "The value of that is that you don't have to populate a whole new data base. You don't run the risk of having to modify your existing program, or having to change any of the data in that program."

AV Image's products complement those already available from Data General's imaging VARs and ISVs (independent software vendors). These solutions are directed at customers with specialized requirements, such as advanced document or workflow management, scanning and editing large engineering drawings, supporting special peripherals, backfile conversion services, computer output to laser disk (COLD), and intelligent optical character recognition (OCR).

Pricing

AV Image is available worldwide through Data General's direct sales force and its VAR channel. DB Link for any of the five data bases mentioned earlier is an additional $300 per client/server user. Group packs for 5, 20, and 100 users are available for both products. The AV Imagizer Toolkit is priced at $995. Δ



OCTOBER 25-28, 1993
THE INFORUM
ATLANTA, GEORGIA

NADGUG 93

will blow you away...

For more information call 1-800-932-6663 ext.1234

Circle 30 on reader service card.

David Novy

DG's "imaging" niche

SYNOPSIS

Our columnist perceives that Data General's new AV Image system achieves remarkable sophistication, but wonders anyway where this emergent document-imaging market will go. One thing seems certain, though: competitive pressures will force companies to move away from paper.

This month's "Focus on:" theme notwithstanding, I don't exactly understand why there is such an emphasis on images and imaging. I guess the people who talk about imaging are actually referring to handling scanned documents. And I don't like the use of the term "images" to refer to large data files. Using today's word processors, it is very easy to encapsulate X-Windows bitmaps into word-processing documents and to make huge data files.

As an example, I recently completed a training manual that contained more than 40 X-Windows bitmaps, and it required more than 9 MB of disk space. By comparison, a Postscript image of the document needed only 8 MB of storage. I never thought that I would someday see an original file that was bigger than its resulting Postscript image.

At any rate, when I stored the document, a backup copy of the document, and the Postscript copy of the document I needed in excess of 25 MB of storage. You can see that working with large data files can use a great deal of storage space in quite a short time. In addition, I am not a great fan of scanned documents, except for archiving purposes, because scanned documents often are merely electronic versions of paper documents. Generally they are much larger than the word-processing document that they represent.

For example, a 200-bits-per-inch representation of an 8 x 10 paper document would require 400 K of storage (200 dots per inch x 200 dots per inch x 8 inches x 10 inches / 8 dots per byte.). By comparison, a one-page document created on this word processor requires less than 5 K bytes.

New data-compression algorithms will greatly reduce the size of the scanned document file, perhaps to as small as 30 K bytes. But this is still six times larger than the file's native form. And at the same time, modern word-processing technology is reducing the need for those scanned images in the first place as a data-sharing medium. Many word processors now offer the ability to handle files from other programs in native form. You don't need a scanned image if you can translate or import the file using one of many available data-interchange formats.

Data General's niche

Data General Corporation's officials have stated that they intend to make the company a major player in the emergent imaging market. In fact, DG has stated its intention to be the low-cost market leader for the imaging market. Data General has recently released a new product, AV Image, for dealing with scanned images. It is based on the system used to keep track of the documents used to build the English Channel tunnel between Great Britain and France.

The system is quite sophisticated (meaning that if somebody wishes to look at a document, the system does not download the full document, but only the information required to allow the user to view the document on a computer screen. If the user then wants a full local copy of the document,, the system will download the entire document

This can result in considerable time savings if you're dealing with 5 MB data files. To help users back up the tremendous amounts of data created by a scanned-image system, Data General is also working to reduce the cost of storing data on optical media.

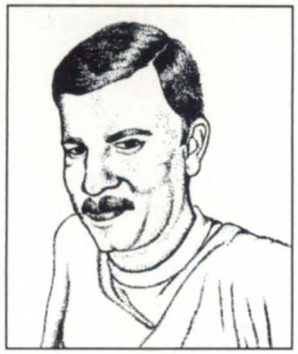
I think that if you're in the insurance business, health care, or in some other business where you're using a large amount of scanned paper documents, the new DG product announcements are quite exciting. However, they do not do much for someone in my business. I am responsible for helping my users store the documents required to produce new products in the shortest time possible. The data stored by my user base are created using word processors and CAD/CAM systems.

What I need are excellent facilities for doing product-data management. Product-data management is concerned with the electronic vault storage of data, bills of materials, and change control. There are no software products available on DG gear that can help me meet my product-data management needs, but such products are available from most of DG's major competitors. The only way I can use Data General equipment in my environment is as a very reliable and powerful server. I can use a product-data manager to control graphic images, but not vice-versa.

The Data General system handles scanned images much more efficiently than can many product-data management systems. But I don't use scanned images, and I think competitive pressures will force companies to move quickly from paper documents to electronic documents.

Dealing with scanned images is a niche market. The mainstream is heading toward product data management and workflow products. Δ

David Novy is a technical computer specialist at 3M in St. Paul, Minnesota. He is past chairman of the AOS/VS special interest group, and current chairman of NADGUG's SIG/UX.



Tim Boyer

Peaceful optimization

SYNOPSIS

Imagine the complexity of a disk optimizer written for a multiuser machine—files constantly created and deleted, multiple users accessing multiple records in the same file, Infos volumes. An optimizer peacefully coexisting with all this chaos? Not bloody likely, say you? Well, check out DISK_PAK.

A few years ago, the folks at Eagle Software brought out an online version of their DISK_PAK utility called, appropriately enough, "DISK_PAK Online!". The only drawback is that it ran under AOS/VS II only. There's been a beta version in the works for, as far as I can tell, about one and a half years, and each NADGUG conference I remind the good people from Eagle that I want to be first in line when the VS Classic product is released.

My wishes have been answered. DISK_PAK Online! for AOS/VS Classic is now available.

Eagle's DISK_PAK is a defragmentation utility, much like the ones you may be running on your PC (DISKOPT or SPEEDISK, or the like). It takes your files, parts of which are scattered over your disk, and brings them together into a contiguous area. This will greatly reduce the seeking a disk must do when sequentially accessing these files. In addition, frequently used files are placed toward the center of the disk, where they can be accessed more quickly.

This is a (relatively) simple concept to picture on a standalone PC, where the optimizer is the only program running. Well, for years that's how DISK_PAK treated your MV—if you wanted to run the optimizer, the *whole* system had to be brought down for a couple of hours. That's the main reason I never took a good look at buying it. The only time that our system could be down is on the weekends, and I spend enough time at the office without volunteering for Sunday work. With an 8 mm tape drive, it was darn near as easy—though certainly not for the faint-hearted—to do a complete dump of the disk, format it, and restore the disk, which would have the same effect as the optimization (albeit without the file placement enhancements).

Optimization out of chaos

Not that I blame anyone for having to run standalone. Imagine the complexity of a disk optimizer written for a multiuser machine! Files constantly being created and deleted, multiple users accessing multiple records in the same file, Infos volumes—and an optimizer peacefully coexisting with all this chaos? Not bloody likely, says I.

Well, Online does it. I spent three weeks trying to break it, and when *my* users can't destroy a product, you know it's solid.

Online is a cinch to install and use. The program uses a key code system familiar to most of us. After the program is installed, your system must be patched with a simple macro and rebooted. This patch enhances AOS/VS to allow for low-level file locking, without which an online optimizer would not be possible. The patch also caused other processes to pause while Online moves portions of the file to a new location on disk.

After the system is rebooted, the people at Eagle recommend that one of their utilities, called "DISK_KePR", be run for a couple of days. On a VS Classic system, this program keeps track of accesses to individual files and logs the information to disk. Revision 2 of VS II already has this feature.

(One of these days, I've *got* to switch over to VS II. I'll give my readers plenty of warning, because when I switch to it, VS II of course becomes a dead operating system).

In addition to its logging duties, "DISK_KePR" also assists the operating system with new file element placement, decreasing the amount of further fragmentation. The utility runs so quietly as to be almost invisible, consuming less than 0.1 percent of the CPU time.

So, I've applied patches and run the utilities. Now it's time to get to the optimizer. In cases like this, with my disk data and future employment at stake, I usually make it a point to read the manual first, much as it goes against my principles. My reaction when I got the manual was that they forgot to send the user's guide. The manual, in its entirety, is less than 40 pages. This could start an entirely new trend in documentation: a manual shorter than *War and Peace*, with just the necessary information in it.

DISK_PAK Online has only a dozen or so switches. The most important is the "/DEFINES" switch, which lets you tell Online the name of a text file used to handle certain files. There are some files that must not be moved, such as program registration files. SYSMGR's PERFMGR.SCF file comes immediately to mind, and, in fact, the sample "/DEFINES" in the manual uses just that file as an example. Probably the only other switch you'd use is "/TRIM=n". This tells Online to minimize the file size of any file not accessed in *n* days by adjusting the element size of that file.

Note that this switch *minimizes* the element size. The area in which I think DISK_PAK is sorely lacking is the optimization of element and directory hash frame sizes. Yes, there are other utilities around that will do that for you, such as BJ's Hazel, but it would sure be nice to be able to run just *one* utility to do it all. I think hash frame and element size optimization are just as important as file contiguity, and they're sure a sight easier to accomplish—or so I thought.

Milton Larson, the author of the program, explained why Online won't currently allow element and hash frame optimization. The problem lies in the online nature of Online. If a process tries to access a file that the optimizer is working on, Online immediately releases that file to the user. This ensures that no one is locked out of a file for any length of time, but it also makes optimization of element sizes impossible. Mr. Larson suggested that this might make a good command-line option—let Online hold onto the file for as long as it needs. In the meantime, do it yourself.

The test

But on to the test. I ran PERFMON from SYSMGR for a week before and after running DISK_PAK, being very careful to pick a time period with no unusual system loads (i.e., I didn't run it over monthend). My main FILES directory went from an average of 69 cylinders traveled per access to 55, a decrease of more than 25 percent. When I mentioned this number of Eagle, they were surprised—it seems the average system improves more than that.

Where I *did* see an enormous improvement was in my history files disk. As mentioned, I'm using Hazel, which tries to keep small files at one index level, and therefore contiguous. Larger files have more than one level. Well, my history file disk is composed of a small number of good-sized (at least, good-sized for ICobol, 20 to 40 MB) files. Hazel didn't make them contiguous, but DISK_PAK does, and my average cylinders dropped from 42 to 18. Now *that's* an improvement that can be noticed.

Online runs rather quietly in the background. I started it with a number of users on line, and reorganized my main files directory. Now this is not the optimal time for running the product—obviously, running along with a number of other users who are creating and deleting files is going to slow the process down. Still, Online only took a half hour to do a 200 MB disk, and the average CPU consumption over that period was a paltry 6.5 percent. For the rest of my disks I ran Online in batch mode at 4 a.m., and no one ever knew it was running.

So between the optimal file placement feature and the contiguous files, DISK_PAK can speed up your disk by a considerable amount, especially if you're doing a lot of sequential file accesses. And if you can't afford to bring down the system to run an optimizer, Online will allow you to optimize on the fly. If this utility can save you 25 percent of your access time, think what that translates to in dollars—and user satisfaction.

Product Info

DISK_PAK Online! is available from Eagle Software, Inc., P.O. Box 16, Salina, KS 67402; 913/823-7257 phone or 913/823-6185 fax. Pricing for the product ranges from $950 to $6,450, depending on CPU type.

New ICobol "feature"

I just discovered a brand-new "feature" of ICobol 1.7x. It seems that the compiler is now working as documented. And that's not good for a lot of us.

The code that made me panic boils down to this:

```
77  TOTAL-ORDERS PIC S9(8)

ADD 1 TO TOTAL-ORDERS.
```

Now, what could possibly be wrong with those two lines of code? All I'm

doing is trying to keep track of the total orders read, in order to do some percentaging.

Only it wasn't working. Finally, I compiled with the /D switch and went into the debugger, to find out what the value of TOTAL-ORDERS was on each go-around. It wasn't incrementing. TOTAL-ORDERS stayed at zero, no matter what. After a little experimenting, I changed it to a "PIC 9(8)", and it worked fine.

Panic time—call the bulletin boards, post a warning, send out an ICobol newsletter, rev back to 1.60. Integer arithmetic is broke! Only first, just to be absolutely sure, I called Envyr and Threshold and told them both what was happening. They promised to look at it and call me back. In the meantime I was getting ready for the backrev.

Fifteen minutes later, the phone rings and a fax comes through, simultaneously, both with exactly the same message.

INITIALIZE YOUR VARIABLES!

Sure enough I did, and it worked. And in fact, it has been in the documentation since 1.60 that uninitialized variables are undefined.

But here's the way it *used* to work. Variables that were undefined contain null values. When ICobol tried to add something to a variable that contained nulls, it initialized it at zero and carried on. The 1.70 compiler changed all that. It will still work for unsigned variables, but with the signed stuff you're on your own.

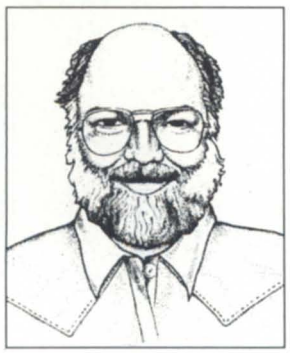
So be safe. If you don't initialize your variables, you'll never know what kind of results you'll get. Sometimes you will be lucky. I don't want to base my career on being lucky, so guess what I'll be doing this weekend?

Initialize your variables. I need all the readers I can get. Δ

Tim Boyer is EDP Manager at Denman Tire Corporation. He may be reached at 400 Diehl South Road, Leavittsburg, OH 44430; phone 216/898-2711 or fax 216/898-5256; on the NADGUG bulletin board at 415/924-3652, or on the CSC bulletin board at 800/DASH-CSC.

Circle 8 on reader service card.

Circle 32 on reader service card.



Brian Johnson

:WFMOVE wrap up

SYNOPSIS

With endurance that awes us lesser folks, BJ pushes inexorably onward toward the light at the end of the :WFMOVE tunnel.

Are you getting tired of ":WFMOVE" yet? Good; me too. This month I'm gonna wrap this thing up by showing you a turbo version of the single task "COPY_DATA" function I showed you last month.

:COPY_DATA_MT.C:DESIGN

If you compare the code for the single task version of COPY_DATA with the new multi-task version, the difference in size and complexity is striking. As a result, rather than describe the logic line-by-line I'll just give you an overview of what's going on.

The first order of business before I started coding was to decide how crazy to get in terms of overlapping the I/O. "LOAD_II" and "DUMP_II" plumb the depths of crazy by going to great extremes to overlap as much I/O as possi-

Figure 8: COPY_DATA_MT.C

```
/* Copy the data from one file to another. */

#define MAXPBES 8 /* No. of PBEs to allocate */

/* Define shorthand for common packet items. */
/* These really improve the code readability. */
#define Opch gopen_pkt.opch
#define Opeh gopen_pkt.opeh
#define Bchn pbep->blkio_pkt.bchn
#define Bsts pbep->blkio_pkt.bsts
#define Badr pbep->blkio_pkt.badr
#define Bbln pbep->blkio_pkt.bbln
#define Bblc pbep->blkio_pkt.bblc
#define Btbc pbep->blkio_pkt.btbc
#define Blbb pbep->blkio_pkt.blbb
#define Bban pbep->blkio_pkt.bban
#define Bbac pbep->blkio_pkt.bbac

/* These Packet & Buffer queue Elements */
/* (PBEs) are used for each read/write. */
typedef struct {
    _ELE elem;          /* Must be first! */
    char *pnp;          /* Pathname ptr */
    long file_blocks;/* File size in blocks */
    Ushort   pes;       /* Pseudo element size */
    enum {OPEN,WRITE,CLOSE} func;  /* Function */
    char *bufp;         /* Buffer ptr */
    P_BLKIO blkio_pkt;     /* ?BLKIO packet ptr */
} PBE;

/* Global storage. */
const char malstr[] = "malloc()";
const char recstr[] = "?REC";
const char xmtstr[] = "?XMT";
int      blocks=64;         /* Override with /BLOCKS=n */

/* Allocate the queue headers and counts. */
_QUEUE   pbq={QNULL,QNULL};     /* Inactive queue */
int      pbqcnt;                /* Cur PBEs on queue */
_QUEUE   dwq={QNULL,QNULL};     /* Disk Writer queue */
int      dwqcnt;                /* Cur PBEs on queue */
int      dwqcnt_max;            /* Max PBEs on queue */

/* Allocate the mailboxes for queue updates. */
int pbbox;   /* Whenever a PBE is added to pbq */
int dwbox;   /* Whenever a PBE is added to dwq */

#define DRPRI2
#define DWTID2
#define DWPRI1

static Boolean    first_time=YES;
static void disk_writer(void);

void pbe_init(void) {
    int      i,ier;
    size_t   bytes;
    PBE      *pbep;

    /* Protect output against simultaneous access. */
    if (ier = mfinit(NULL,1)) {
        error("mfinit()",ier);
    }

    /* Spawn the disk writer task. */
    if (ier = mtask(disk_writer,1024,DWTID,DWPRI)) {
        error("mtask()",ier);
    }

    /* Lower my priority below that of the disk */
    /* writer task to give writes an advantage. */
    if (ier = sys_pri(DRPRI)) {
        error("?PRI",ier);
    }

    /* Allocate all the PBEs. */
    for (i = 0; i < MAXPBES; i++) {
        pbep = (PBE *) malloc(sizeof(PBE));
        if (pbep == NULL) error(malstr,ERMEM);
        bytes = blocks * 512;
        pbep->bufp = (short *) malloc(bytes);
        if (pbep->bufp == NULL) error(malstr,ERMEM);
        insert_tail(&pbq,(_ELE*)pbep); pbqcnt++;
    }

    first_time = NO;
}

/* Allocate a PBE. */
PBE *pbe_alloc(void) {
    int      ier,msg;
    PBE      *pbep;

    /* Get a PBE from the idle queue. */
    while (pbq.head == QNULL) {
        if (ier = sys_rec(&pbbox,&msg)) {
            error(recstr,ier);
        }
    }
    pbep = (PBE *) remove_head(pbq.head); pbqcnt--;

    /* Reset the ?BLKIO packet. */
    memset((void*)&pbep->blkio_pkt,0,sizeof(P_BLKIO));
    Badr = pbep->bufp;

    return pbep;
}

/* Determine the pseudo-element size by finding */
/* the largest number <= blocks that is a factor */
/* of an element size in blocks (1..65532). */
int get_pes(int elem) {
    int      pes;

    if (elem < blocks) {
        pes = elem;
    }
    else {
        for (pes = blocks; pes > 0; pes--) {
            if (elem % pes == 0) break;
        }
        if (pes == 0) pes = 1;
    }
    return pes;
}

/* Copy ofnp to dpnp reasonably efficiently. */
int copy_data(char *dpnp, char *ofnp,
                                P_FSTAT *fstat_pktp) {
    int      access,blks,blocks_left,file_blocks;
    int      ier,modified,msg,pes;
    long next_block;
    P_GOPEN gopen_pkt;     /* Origin ?GOPEN packet */
    PBE      *pbep;

    /* Need to initialize? */
    if (first_time) pbe_init();

    /* Empty files are easy. */
    if (fstat_pktp->sefm == 0) return 0;

    /* Open the origin file. */
    memset((void*)&gopen_pkt,0,sizeof(P_GOPEN));
    if (ier = sys_gopen(ofnp,-1,&gopen_pkt,
                                        &access)){
        warn(ofnp,ier);
        return ier;
    }

    /* Must have read access. */
    if ( (access & $FACR) == 0) {
        warn(ofnp,ERFAD);
        if (ier = sys_gclose(Opch,&modified)) {
            error(ofnp,ier);
        }
        return ERFAD;
    }

    /* Tell the disk writer to open the file. */
    pbep = pbe_alloc();
    pbep->pnp = dpnp;
    pbep->func = OPEN;
    pbep = pbe_enque(pbep);

    /* Determine the pseudo-element size and the */
    /* no. of blocks to use per read. */
    pes = get_pes(fstat_pktp->sdeh);
    blks = (blocks / pes) * pes;

    /* Copy the data. */
    next_block = 0;
    blocks_left = file_blocks = (Opeh + 511) / 512;
    while (next_block < file_blocks) {

        /* Get a PBE. */
        pbep = pbe_alloc();

        /* Read the next allocated element(s). */
        Bsts = $BMNAB;
        Bchn = Opch;
        Bbln = next_block;
        Bblc = (blocks_left < blks) ?
                                (blocks_left) :(blks);
        if (ier = sys_blkio(&pbep->blkio_pkt)) {
            if (ier != EREOF) {
                warn(pbep->pnp,ier);
```

Figure continued on page 28

ble, including multiple creates and opens.

Unfortunately, the code complexity required to do that greatly obscures the logic. But fortunately, it adds little to the overall throughput. In other words, the biggest gain is obtained by simply overlapping the read and write I/Os, and little additional speed is gained by overlapping creates, opens, and closes.

Another consideration in this case was to limit the changes to a single module and not have to redesign a lot of the earlier coding to accomplish the speed-up.

So, the basic design involves changing the COPY_DATA function from a single task that does a two opens, reads and writes, and two closes into two tasks, where the main task does all the stuff associated with reading the origin file and the write task does all the stuff associated with writing the destination file (except for the create that was done earlier by MOVE_FILE).

The disk writer task is spawned and the pool of disk I/O buffers is allocated upon the first call to COPY_DATA.

This was done to compartmentalize the enhancement as much as possible and avoid having to change the main function source code. The basic structure involved is a thing called a PBE (Packet/Buffer Entry), which contains a function code (OPEN, CLOSE, or WRITE), a "?BLKIO" packet, and a buffer.

The buffer size is determined by a global variable called "blocks", with a default value of 64 that can be overridden by a global "/BLOCKS=n" switch that I added to the "GET_GLOBAL_SWITCHES" function.

I keep track of the PBEs using two MV queues: "pbq" to hold the "idle" PBEs, and "dwq" to hold the PBEs waiting to be written by the disk writer task. The advantage of using the MV's nice hardware queue instructions is that they are intrinsically re-entrant, so no locks need to be used to protect against simultaneous access.

PBEs are allocated and freed using only the front idle queue because their order on the queue is unimportant, and also because doing it this way minimizes the working set size associated with PBEs. On the other hand, the *dwq* is used in strict FIFO order as you might expect.

In order to assess the sufficiency of the number of PBEs (specified by the MAXPBES definition), the count of the number of entries on both queues is maintained and the maximum length of the disk queue is recorded and reported upon exit. MAXPBES is currently set to 8, and the maximum disk writer queue length that I've seen reported so far is only 3, so clearly 8 is more than sufficient.

This brings up an interesting potential pitfall—keeping track of queue lengths using a simple integer variable updated by multiple tasks is potentially risky.

For example, the statement "pbcnt =pbcnt + 1" will eventually mess up in

Figure continued from page 26

```
                pbep = pbe_free(pbep);
                break;
            }
            /* Last read ended at EOF? */
            /* If so then we're done. */
            if (Btbc == 0) {
                pbep = pbe_free(pbep);
                break;
            }
        }

        /* Determine the next block no. before */
        /* enqueing the PBE to the disk writer, */
        /* otherwise he might modify it before I */
        /* can compute the next block number. */
        next_block = Bban + Bbac;
        blocks_left = file_blocks - next_block;

        /* Enqueue the PBE to the disk writer. */
        pbep->pnp = dpnp;
        pbep->file_blocks = file_blocks;
        pbep->func = WRITE;
        pbep->pes = pes;
        pbep = pbe_enque(pbep);
    }

    /* I need one last PBE to signal the disk */
    /* writer task to close the output file. */
    pbep = pbe_alloc();
    pbep->pnp = dpnp;
    pbep->func = CLOSE;
    pbep = pbe_enque(pbep);

    /* Close the input file, hopefully overlapping */
    /* the disk writer's close of the output file. */
    if (ier = sys_gclose(Opch,&modified)) {
        error(pbep->pnp,ier);
    }

    /* Wait for the disk writer to finish. */
    while (dwqcnt) {
        if (pbbox) {
            pbbox = 0;
        }
        else {
            if (ier = sys_rec(&pbbox,&msg)) {
                error(recstr,ier);
            }
        }
    }

    /* Success. */
    return 0;
}

/* Wait for a CLOSE request. */
void wait_for_close(void) {
    PBE     *pbep;

    do {
        pbep = pbe_deque();
    } while (pbep->func != CLOSE);
}

/* Flush any blocks waiting to be written. */
int pbe_flush(PBE *pbep, Boolean last) {
    int     ier;

    /* Any pseudo-elements to write? */
    if (Bblc == 0) return 0;

    /* Write the pseudo-elements. */
    Blbb = (last) ? (Btbc % 512) : (0);
    if (ier = sys_blklo(&pbep->blkio_pkt)) {
        warn(pbep->pnp,ier);
        return ier;
    }
    /* Advance beyond what was written. */
    Badr += 256 * Bbln;
    Bbln += Bblc;

    /* Reset the count of pseudo-element */
    /* blocks waiting to be written. */
    Bblc = 0;

    return 0;
}

/* A task to process destination file */
/* open/write/close requests. */
void disk_writer(void) {
    int     access,ier,i,j,modified;
    long pebn;
    size_t  pesz;
    Boolean ignoring,last;
    char *pecp;
    PBE     *pbep;
    P_GOPEN gopen_pkt;

    /* Indicate that no file is open. */
    Opch = -1;

    /* Indicate that we are not ignoring WRITEs. */
    ignoring = NO;

    /* Loop processing disk write requests. */
    for (;;) {

        /* Get a PBE to process. */
        pbep = pbe_deque();

        /* Open? */
        if (pbep->func == OPEN) {
            if (Opch != -1) error(pbep->pnp,EROPR);
            memset((void*)&gopen_pkt,0,
                                sizeof(P_GOPEN));
            if (ier = sys_gopen(pbep->pnp,-1,
                                &gopen_pkt,&access)) {
                warn(pbep->pnp,ier);
                ignoring = YES;
            }
            pbep = pbe_free(pbep);
            continue;
        }

        /* Close? */
        if (pbep->func == CLOSE) {
            if (ier = sys_gclose(Opch,&modified)) {
                error(pbep->pnp,ier);
            }
            Opch = -1;
            pbep = pbe_free(pbep);
            ignoring = NO;
            continue;
        }

        /* If we get here then it must be a WRITE. */

        /* Ignoring WRITEs? */
        if (ignoring) goto ignore;

        /* Scan the buffer a pseudo-element at a */
        /* time looking for sparse pseudo-elements.*/
        /* Non-sparse pseudo-elements are simply */
        /* counted in bblc and then flushed upon */
        /* encountering a sparse pseudo-element. */
        Bsts = $BMIO; Bchn = Opch;
        Bbln = Bban; Bblc = 0;
        pecp = Badr; pebn = Bban;
        pesz = pbep->pes * 512;
        for (i = 0; i < Bbac; i += pbep->pes) {

            /* Determine whether this is the */
            /* last pseudo-element in the file. */
            last = (pebn + pbep->pes >=
                                pbep->file_blocks);
            if (last) {
                /* Yep: must write this pseudo- */
                /* element, even if it's null. */
                if (j = Btbc % pesz) {
                    j = j / 512 + 1;
                }
                else {
                    j = pbep->pes;
                }
                Bblc += j;
                if (pbe_flush(pbep,last)) {
                    goto ignore;
                }
            }
            else {
                /* No: check for a null pseudo- */
                /* element. If so, flush any */
                /* waiting non-null pseudo-elements*/
                /* and skip over the null one. */
                if (*pecp !!
                        memcmp(pecp,pecp+1,pesz-1)){
                    Bblc += pbep->pes;
                }
                else {
                    if (pbe_flush(pbep,last)) {
                        goto ignore;
                    }
                    /* Skip null pseudo-element. */
                    Badr += pbep->pes * 256;
                    Bbln += pbep->pes;
                }
            }

            /* Advance to the next pseudo-element */
            /* in the buffer. */
            pecp += pbep->pes * 512;
            pebn += pbep->pes;
        }

        /* We've checked all the pseudo-elements in*/
        /* the buffer.    If the last pseudo-element*/
        /* in the buffer was null then we may have */
        /* exited the for loop with unflushed non- */
        /* null pseudo-elements still unwritten. */
        /* If so, write them. */
        if (Bblc) {
            /* Note that last can only be NO at */
            /* this point of flush would have */
            /* occurred earlier in the for loop. */
            if (pbe_flush(pbep,last)) {
ignore:             ignoring = YES;
            }
        }

        /* We're done with this PBE now. */
        pbep = pbe_free(pbep);
    }
}
```

a multi-task environment because it involves three instructions: a load, an add, and a store.

A task switch between the load and add, or between the add and store, will corrupt the value of the variable. Protecting the statement with a lock is relatively expensive, but luckily an alternative syntax exists that generates reentrant code: "pbqcnt++". In this case DG C generates a single add-to-memory instruction that is guaranteed uninterruptible.

As far as I know, this "feature" is undocumented. Well, that's why God invented us "experts" who can read the compiler's generated code.

Okay, on to the logic flow. In simple terms here's how things work: when the COPY_DATA function is called it opens the origin file and allocates an idle PBE, loads it with the stuff to do an open of the destination file, and enqueues it to the disk writer queue.

Then it loops allocating a PBE, filling it with data, and enqueueing it to the disk writer queue for writing. It uses ?BLKIO's Next Allocated Block (NAB) option to automatically detect and skip over existing sparseness in the origin file. Finally, it allocates a "CLOSE PBE", enqueues it to the disk writer queue, and then closes the origin file. This effectively overlaps all destination file accesses with origin file accesses.

The disk writer task sits in an endless loop waiting for a PBE to appear on the disk writer queue for processing. The PBE will be one of three types: OPEN, WRITE, or CLOSE. Each is acted on accordingly. If any error occurs during the open or any write, the disk writer sets the "ignoring" flag and simply processes all further WRITE PBEs until a CLOSE PBE is seen.

That's pretty much it. Sounds simple, eh? Too bad it doesn't look as simple.

:SPARSENESS

In the first version of WFMOVE I preserved existing file data element sparseness using the NAB feature of ?BLKIO, but I didn't do anything fancy in the way of detecting potential new sparseness. That's all changed this time around; files moved are now made as sparse as possible. This is done both to conserve disk space and to avoid unnecessary I/O. As I say during my seminar, "The fastest disk access is one that is avoided."

Detecting and generating new sparseness both efficiently and with simple code is a bit tricky because it requires that I/O be aligned on element boundaries. One possible solution would be to simply make the I/O buffer equal to the element size, but huge element sizes make that scheme impractical and small element sizes result in too many writes.

Another solution would be to avoid writing any single disk block that has null contents. Unfortunately, lots of non-null elements have null blocks within them and skipping over the null blocks would generate lots of

extra writes.

The solution I chose was to set the buffer size to a nominal value of 64 (or whatever /BLOCKS=n is), and then for each file determine the size of a "pseudo-element", and read and write as many pseudo-elements at a time as will fit in the buffer. A pseudo-element is defined as the largest number of blocks that's less than the buffer size and a factor of the file's element size. The "GET_PES" function determines the pseudo-element size for a file given the buffer size and origin file element size. The actual buffer size used is then the largest number of pseudo-elements that will fit in the buffer ,and the sparseness check consists of scanning a pseudo-element at a time for nulls and skipping over them.

For files whose element size is less than the buffer size, this is optimal. For files whose element size exceeds the buffer size it is nearly optimal. Rarely will the buffer size be much less than the maximum size, and it can never be less than half the maximum size (32 blocks for the default case).

In order to maximize the size of each write, writing non-null pseudo-elements is postponed until a) a sparse element is encountered that is not the last pseudo-element in the file; or b) the end of the buffer is reached. At that point all the postponed pseudo-elements are written at once. The "PBE_FLUSH" function is used to do it.

The net effect is that minimal writes are done and sparseness is both preserved and, where possible, unnecessarily allocated elements are avoided.

:GOTCHAS

The DG C manual specifically prohibits manipulation of the heap by any task other than the main task when a dynamic stack/heap (the default) is used. WFMOVE's disk writer task does not explicitly access the heap, but many library functions do so implicitly. As a result, the new version of WFMOVE has to be LINKed with a ".STKSIZE=n/DEFINE" argument. A value for *n* of 10,000or more is quite sufficient for our needs since all bulky storage is explicitly allocated from the heap instead of being statically defined.

In a multi-task environment where several tasks can do I/O to common C files, you must protect against simultaneous updating of the structures pointed to by C's file pointers. This is accomplished using the "mfinit" function that is documented in the DG C manuals.

:JELLO

They say that the proof is in the pudding, so here are the results from the road test involving MOVEing the contents of the :SYSGEN directory from one disk to another using a variety of techniques:

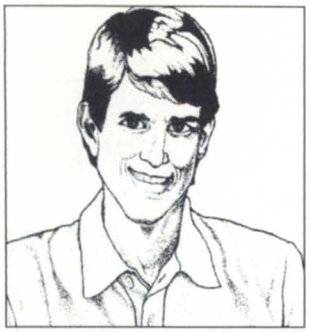
Test	Elapsed	CPU
WFMOVE 1.00	0:01:03	4.7
WFMOVE 1.10	0:00:45	5.4
MOVE	0:01:43	14.5
MOVE/BUFF=4096	0:01:24	18.1
MOVE/BUFF=8192	0:01:17	16.0
MOVE/BUFF=16384	0:01:15	15.8
MOVE/BUFF=32768	0:01:16	16.4
DUMP_II/LOAD_II (via a PIPe file)	0:01:01	32.1

The slight increase in CPU consumption going from rev 1.00 to rev 1.10 is probably due to the more aggressive sparseness check, the extra code logic, and the system calls required for inter-task communication.

I speculated back in April that I'd be able to halve WFMOVE's elapsed time using multi-tasked I/O, but I ended up reducing it by only about 30 percent. Still I did manage to get the elapsed time below that of the DUMP_II/LOAD_II using a PIPe file, but with substantially less CPU consumption.

Plus, as far as I'm aware this is the only program that generates new sparseness where appropriate. Δ

 BJ is president of B.J. Inc., a San Francisco-based consultancy specializing in system auditing, system management, and performance analysis. :SYSMGR is a division of B.J. Inc. BJ can be reached at 109 Minna St., Suite 215, San Francisco, CA 94105, 415/550-1444 (voice) or 415/550-1072 (fax). The :SYSMGR Bulletin Board Service number is 415/391-6531 (300/1200/2400 with optional MNP class 5; 605X, 8/N/1) or 415/550-1454 (voice).



Kim Medlin

A quick tour of the SQL language

SYNOPSIS

Though you've probably heard of SQL (structured query language), perhaps you've never used it. It's easy to learn and difficult to master, but it's a giant step forward for productivity, reliability, and portability. Major features of SQL we will discuss are: 1) Concurrency management; 2) Simple yet powerful data structures. 3) Non-procedural syntax; and 4) Set-at-a-time processing.

Open systems is certainly one of the hot topics of our time. Application portability ranks right up there with functionality as a critical feature for new systems. Here at Data General's Systems Integration Services, we don't specialize in selling—we specialize in delivering! Many new applications include relational data base management systems that utilize the SQL standard. Though you may have heard of SQL before, perhaps you've never used it. This article will take you on a brief guided tour of the SQL language (all syntax is Oracle SQL). You might as well come along on the tour. You've already paid the price of admission!

The terms *SQL* and *relational data base management system* (RDBMS) are not synonymous, even though it sometimes seems that way. Technically speaking, an RDBMS is basically a collection of data stored in two-dimensional tables. Relationships between the tables are ascertained by relating common field values from the tables. The relational model is based on relational algebra (although I don't remember my college math department offering that course). Fortunately, you don't need a math degree to be proficient with SQL.

Tables: The building blocks of RDBMSs

SQL *tables* are logically synonymous to *files*. Each table contains a

Continued on page 34

Figure 1: Create table statement

```
create table student
(
ssn             decimal(09)   not null,
last_name       char(20),
first_name      char(20),
middle_name     char(20),
status_code     char(01),
status_date     date,
addr            char(30),
city            char(30),
state_code      char(02),
zip_code        decimal(05),
school_code     char(08)
);
```

Figure 2: Table indexes

```
create unique index ssn_index on student (ssn);

create index stdt_name_index on student (last_name, first_name,
middle_name);
```

AD INDEX

Company	PG#	RS#
Ames Sciences, Inc.	18	1
Applied Computer Solutions	33	-
Asset Remarketing	11	2
Claflin & Clayton, Inc.	37	3
Claflin & Clayton, Inc.	24	4
Computer Engineering International	13	5
Computer Wholesalers, Inc.	14	6
Contemporary Cybernetics Group	C2	7
Data Bank Associates, Inc.	25	8
Data Bank Associates, Inc.	17	9
Data General Corporation	3	10
Data General—Hot Site	28	12
Data General Professional Services	14	11
DataLynx	21	14
Digital Computer Consulting	23	15
Digital Dynamics, Inc.	23	16
Eagle Software, Inc.	9	17
Eagle Software, Inc.	7	23
Gerry Manning & Associates	29	18
Hanson Data Systems, Inc.	6	19
Intelligent Information Systems	34	13
International Computing Systems	16	20
Mac Data, Inc.	23	21
Marc Alan Software, Inc.	15	42
Minitab Statistical Software	33	-
NADGUG	19	30
NADGUG	38	30
NADGUG	39	30
National Computer Dynamics	17	24
Park Place International	12	25
PereLine Data Systems	33	-
R&D Computers	30	41
RAVE Computer Association, Inc.	C4	26
Rhintek, Inc.	27	27
Sabra Systems, Inc.	36	28
SCIP	35	29
Scott McIntyre & Associates	C3	40
Security Computer Sales, Inc.	36	31
Strobe Data, Inc.	39	22
Sysgen Data Ltd.	25	32
:SYSMGR, a division of B.J. Inc.	18	33
:SYSMGR, a division of B.J. Inc.	30	34
:SYSMGR, a division of B.J. Inc.	33	-
Threshold, Inc.	18	35
Transoft Ltd.	5	39
Universal Data Corporation	38	36
Wild Hare Computer Systems, Inc.	1	37

PRODUCTS AND SERVICES

Company	PG#	RS#
Clarify, Inc.	38	50
Contemporary Cybernetics Group	38	51
Kermit Distribution, Columbia Univ.	39	52

ON-LINE HELP

***Who to call for answers about* NADGUG *and* FOCUS**

NADGUG address:
c/o Danieli & O'Keefe Associates, Inc.
Chiswick Park, 490 Boston Post Rd.
Sudbury, MA 01776
FAX: 508/443-4715

FOCUS Magazine address:
c/o Turnkey Publishing, Inc.
P.O. Box 200549
Austin, TX 78720
FAX: 512/335-3083

NADGUG

Membership, RIGs, SIGs
NADGUG staff **800/253-3902**
(Outside the U.S and Canada.) **508/443-3330**

Electronic bulletin boards
(300,1200, 2400, or 9600 baud modem)
Rational Data Systems **415/499-7628**
DASH bulletin board (2400 bps modem) **800/DASH-CSC**
Data General Customer Support Center **800/344-3577**

FOCUS Magazine

512/335-2286

Editorial comments, article suggestions.....Doug Johnson
(please send product announcements to the address listed above)

Information about advertising.....Susan de Blanc

FOCUS back issues .....Turnkey Publishing staff

Continued from page 4

Directions (very much like listening to Ron Skates and Steve Baxter at NADGUG).

The evening banquet was delightful, with everyone receiving a special anniversary bottle of wine and wineglass. Ron Skates was presented a beautiful crystal clock from the UK Users Group. The evening's entertainment was a hypnotist, which I really don't remember too much about (just ask Dennis Doyle). After the banquet Brian Johnson entertained us all at the bar, where he struck up various conversations with the Avon ladies.

Other highlights included a presentation on computer-supported telephony—getting ahead of the marketing using telecommunications, telemarketing software, and interfacing with MVs and Aviions. Brian Johnson's view of Unix and destination client servers led by Dun & Bradstreet Software Services. Ron Skates presented an address on Data General's corporate strategies and current financial conditions.

Overall, the UK conference was a huge success for Hugh Ross of Scotland, who collected money for a special charity. However, he was "required" to wear his Scottish kilt at the banquet.

I want to thank the following people for making the UK trip educational and enjoyable for me. And I extend to every one of them a special invitation to NADGUG 93 in Atlanta this October 25-28:

Data General User Group UK: Paul Hewton, Data General User Group Chairman; Bruce Irving, Data General User Group Incoming Chairman; Nigel Ockenden, Data General User Group England; Hugh Ross, Data General User Group Scotland; Olga Kennedy, Data General User Group Ireland; Sheila O'Reilly, User Group Coordinator; and Dave Robinson, Eclipse Magazine Editor.

Data General Limited: Bill Cadagon, Director of European Services; Michael Skehin, Customer Service Manager; Terry Radford, General Manager; and John Coon, Marketing Director.

Hope to see you in Atlanta this October for NADGUG 93! Δ

DIAL-UP BULLETIN BOARDS

:SYSMGR BBS specializes in file transfer of RDOS and AOS[/VS] DUMP files - no messaging facilities. XMODEM, YMODEM, and KERMIT supported. 415/391-6531(one line), 2400 baud (Microcom AX9624c), 8 data bits, 1 start /stop bit or 415/550-1454 (voice). System is MV/4000, terminal mode is CHAR/605X.

NADGUG

for **NADGUG**
Membership information contact:

Danieli & O'Keefe Associates, Inc.
1-800-253-3902 (U.S. and Canada only)
508-443-3330

The North American Data General Users Group is an incredible resource when you need answers.

So, don't go it alone - join **NADGUG** *today!*

SOFTWARE

Statistical Software
- Powerful
- Fast
- Easy-to-use
- Inexpensive

MINITAB
STATISTICAL SOFTWARE
3081 Enterprise Dr., State College, PA 16801
814-238-3280

DG410 Emulation/Modem Communication Software
Background file transfers, over 45 modem definitions, extensive script language plus additional terminal emulations. Security features. Only $49.95. Ask for Free evaluation copy.
Pereline Data Systems
(408) 364-2770

Communications Software
DG-DG DG-PC DG-??

Share data quickly and easily.
Use **BREAKTHROUGH**
X/YMODEM AOS/VS-AOS-RDOS
Script Language Terminal Emulator
Many free enhancements
(505) 434-0206
Applied Computer Solutions, 2336 Union Avenue. Alamogordo, NM 88310

FREE! to NADGUG members!

- Focus Magazine/12 issues
- Electronic Bulletin Boards
- NADGUG software library
- Member Directory
- Networking with other DG users
- Access to RIG/SIG network
- Communication channel to DG
- Discounts on conference

Continued on from page 31

number of records called *rows*. Each row is subdivided into *columns* (or fields). To create a table, you'd issue the statement in Figure 1(page 31).

It's as simple as that to create a table. We can now insert data into the table. Be aware, however, that as neat as SQL is, it isn't voodoo or magic. The table will need to have at least one index created for it (this ensures fast access). The statements in Figure 2 build two indices, one for the social security number (the primary key) and the other for the student name (the alternate key).

Boolean logic bites the dust

Notice the "not null" clause on the "ssn" field in Figure 1. This clause enforces a non-null value for each occurrence of the "ssn" field (i.e., a data value is required). The concept of a null value is a radical departure for data management systems like Infos or DG/DBMS.

Before, Boolean logic ruled. For example, every numeric value was either zero or non-zero, and every alphanumeric field was either spaces or non-spaces. However, a null value is different from zero or spaces. The null value means "unknown". Therefore, the null value introduces 3-way logic for all fields that allow null values. In our SQL table, the values of "zip_code" can be either zero, non-zero, or null. Therefore, if we were to ask SQL if John Smith's zip code is within zone "282", the answer would be "yes", "no", or "maybe". This more accurately reflects the real world of business data than ever dreamed of by the 2-value logic model.

Data manipulation

Now that we have a data structure we can insert some data into it. This is done with statements similar what is displayed in Figure 3.

Notice that the value of the "middle_name" field was not known at the time of the insertion; in this case, it was assigned a value of null.

The data can be modified just as easily.

The second "update" statement in Figure 4 exemplifies set-at-a-time processing. All student records with null values for the status_code field were updated to be "active," with the system date stored in "status_date". I dare say this is easier and more intuitive than writing some sort of third-generation language (3GL) loop to change this set of records.

Deleting records is just as easy. The statement in Figure 5 removes from the data base all students who are currently attending UGA.

Figure 4: Modifying data

```
update student
  set student.middle_name = 'Q'
  where student.ssn = 239764312;

—or—

update student
  set student.status_code = 'A'
  student.status_date = sysdate
```

Figure 5: Deleting records

```
delete from student
    where
    student.school_code = 'UGA';
```

Figure 3: Inserting data

```
insert
  into student
  values

  (239764312,'Smith','John',null,'A','123 Main Street','Any
  town','GA',30083,'UGA');
```

Data structure manipulation

The data structures can be altered as easily as data is modified. If we want to add a field to the student table, the following statement would do it:

```
alter table student
    add (birth_date date);
```

If we decide the alternate index on student name is no longer needed, we can eliminate it like so:

```
drop index stdt_name_index;
```

This gives you a flavor of how intuitive manipulating SQL data structures can be. Imagine, if you will, how difficult it would be to accomplish the above tasks in a hierarchical data management system. Please notice that at no time did we need to dump and load data. SQL is intended to come down only when a new revision is installed.

Data retrieval—the acid test

Creating data structures and inserting data into the data base is one thing. The real test of a DBMS is data retrieval. After all, users don't much care what technology is being employed. They just want their data reported to them in sometimes unimaginable combinations, and they want it *now*. It is imperative that the data base be able to provide flexible, complex, and *ad hoc* query capabilities. SQL fits the bill.

Let's bypass trivial cases such as "list all students alphabetically". A more real-world example might be to report the minimum, maximum, and average ages of students grouped by school and sorted in descending order by the number of students from each school. See Figure 6 for the SQL statement and resulting output would be:

Notice that SQL did some very important things for us in this example. First of all, there was no logic written to loop through the records, add values, divide by the total records, and so forth. This shows that SQL is nonprocedural. Also notice that we accessed many records with one statement using SQL's ability for set-at-a-time processing.

Another thing you should be aware of is that some of the records in the data base had null values for the "birth_date" field. These records were ignored by the above SQL statement. In general, that is the correct action for SQL to take. Since a null value means "unknown", it would probably be inappropriate for SQL to assume a null should be used at all. If, however, I had wanted the null values to be treated differently, there are approaches that will alter SQL's default behavior.

Transactions ensure data integrity

How many times has the following

Figure 6: Retrieving data

```
select
        school_code,
        min(1993 - to_number(to_char(birth_date,'yyyy'))) min_age,
        max(1993 - to_number(to_char(birth_date,'yyyy'))) max_age,
        avg(1993 - to_number(to_char(birth_date,'yyyy'))) avg_age,
        count(birth_date) stdts_in_calc
    from student
    group by school_code
    order by count(birth_date) desc;
```

SCH CODE	MIN AGE	MAX AGE	AVG AGE	STDTS IN CALC
ASU	21	25	22	413
UNC-CH	19	31	24	211
NCSU	20	27	24	132
...				

SQL: A new language approach

SQL (pronounced SEE-quel) stands for **S**tructured **Q**uery **L**anguage. It is the established syntactical standard for RDBMS access. Whether SQL is the best possible language to access an RDBMS is a moot point—it's been adopted by ANSI, IBM, Oracle, and others (though not necessarily in that order), so it's here to stay.

But that's not so bad. SQL is a neat language. Notice, however, it's *not* a programming language. Some of its characteristics are:

- It's a data base language. Its commands only access and modify the data base.
- Even though SQL stands for Structured Query Language, the language is much more than a query language. It also defines data base structure and modifies the data in the data base.
- It is non-procedural, so the language has no constructs such as looping or branching.
- It can perform set-at-a-time processing. That is, one SQL statement can access many records.
- It is implemented at a higher level than 3GLs such as Cobol or C. For instance, you tell SQL "what" you want to happen and SQL determines "how" to accomplish your request. This seems scary for many traditional programmers, but once you get the hang of it you'll wonder what took you so long to make the switch.
- Last but not least, SQL employs "transaction processing" (see main article). Δ

scenario happened to you?

The system crashed while your application was active. After your brought the system up and ran necessary verification routines, the users began using the application again. Later, you started getting a bad feeling that the data might not be consistent. Weeks or months later you noticed inconsistent data and you wondered which system crash to blame.

Every major SQL engine has transaction processing built into it. This means that logical groups of data base I/Os are posted to the data base in their entirety, or else not at all. If the familiar horror story in the previous nightmare scenario had occurred with a SQL data base in place, nothing bad would have happened. The system would have come back up. No data base verification routines would have been run, though. The data base would have been immediately available for user requests. You would have been able to sleep well that night knowing that SQL had ensured your data base's data integrity.

Please be aware that the primary concern here is that traditional data management products *do not offer a solution* to this problem. Many products will allow you log data I/Os that can be applied back to the data base after the system comes back up. But these techniques allow data integrity only to the I/O level, not the transaction level. This means that you can still have severe data problems even after applying the log files. The point is that *SQL does offer an elegant and transparent solution.*

Conclusion

As you've seen, SQL offers intuitive mechanisms for building data structures and dealing with the actual data. The data structures are simple two-dimensional tables that are easy to understand and work with. The SQL language itself is easy to learn and difficult to master. But then, a truly powerful language that is easy to master hasn't yet been invented.

The true test for SQL is a comparative one. Take the data-management software you're currently using as an example. Now consider the examples presented in this article and honestly ask yourself: How easy it would have been to implement solutions using your data management product? Had you been using one of the modern dialects of SQL, such as DG/SQL or Oracle, your life would be much easier and your company's strategic assets, called "data," would be safer. Δ

Kim Medlin is a Senior Consultant with Data General's Systems Integration Services group in Atlanta, Georgia. Systems Integration Services specializes in custom software design, development, implementation, and consulting. Kim's address is 4170 Ashford Dunwoody Road, Suite 300, Atlanta, GA 30319. He can be reached at 404/705-2653. His e-mail address is klm@atlanta.dg.com.

The latest products for DG systems

Backup management AV

Newport News, VA—Contemporary Cybernetics Group (CCG) announced that its 8 mm tape library systems, the CY-CHS10i and the CY-CHS120, are available with backup management software. Designed for multi-vendor Unix networks, the software automates backup and restore operations and gives users direct access to between 25 GB and 3 TB of data—without manual intervention. Any Unix system can back up any other Unix system on a network.

The CY-CHS10i features one 8 mm tape drive and 10 tapes in a desktop cabinet; the CY-CHS120 features up to four 8 mm tape drives and 116 tapes. Both libraries utilize a robotic tape handler to load and unload tapes without manual intervention. Unattended backups, either full or incremental, can be scheduled to run automatically at any time or date. Backups can also be performed with users on-line. The software catalogs each backup, creating an audit trail for recordkeeping and security.The software labels tapes electronically and recommends a tape for each backup. Multiple backups can be written to a single tape. Tapes are rotated to ensure even wear, and are recycled automatically. Before backup begins, the software verifies that both the device and the driver are ready; it also checks to ensure that the correct tape is mounted to eliminate the risk of accidental overwrites.

Contemporary Cybernetics Group, 11846 Rock Landing, Newport News, VA 23606; 800/873-9000.

Circle 51 on reader service card.

Business tools AV PC

San Jose, CA—Clarify, Inc., announced an expansion of its customer management service management (CMS) system. Clearsupport 2.0 and Clearquality 2.0 upgrades provide management with more tools to manage service and support businesses and provide a variety of reports and other methods for extracting valuable data.

Clearsupport is a technical-support management system; Clearquality is a defect-tracking system. Each are part of a

comprehensive CSM system designed for use by software companies that often have hundreds of engineers, thousands of customer calls per day, and millions of dollars of field-service inventory. The systems are client/server-based and native on Macintosh, PC/Windows, and Unix/Motif computers with Sybase and Oracle relational data base management systems.

Clearsupport 2.0 and Clearquality 2.0 now offer numerous standard reports, allowing managers to see, for example, how many times a given problem has been found or how often a given enhancement has been requested.

A typical Clearsupport configuration averages $6,000 per concurrent user, including all functionality, including the Clearsupport Diagnosis Engine. Clearquality averages $4,000 per concurrent user, including all functionality and interfaces to the Clearsupport system.

Clarify, Inc., 2702 Orchard Parkway, San Jose, CA 95134; 408/428-2000.

Circle 50 on reader service card.

Kermit the communications software

New York, NY—Kermit, Columbia University's portable communications software for just about every known operating system, offers a new revision: C-Kermit 5A.

For Unix, Data General's AOS/VS, MS-DOS, and a wide variety of others, C-Kermit offers a uniform and comprehensive solution for direct serial, dial-in, dial-out, and TCP/IP network communications.

Unix C-Kermit is available for all major strains of Unix, including DG/UX for the Aviion platform and Next Computer's Nextstep system. AOS/VS C-Kermit is available for DG Eclipse MV series minicomputers (also includes AOS/VS II). C-Kermit is already seeing heavy service at NASA where, for example, it's installed in the Spacelink information server, and at the USDA Forest Service.

Since its previous release—4E(072) in 1989—C-Kermit's file-transfer efficiency has been dramatically improved by the addition of sliding windows (up to 31 window slots) and long packets (up to 9024 bytes). It is now possible to have up to 280,000 bytes "in the pipe" before acknowledgment is required, bringing average text-file transfer efficiency—even over long-delay satellite and/or public network connections—into the 85 percent to 95 percent range, *before* compression.

C-Kermit's character-set conversion capabilities are not confined to Western European languages like Italian, French, German, Spanish, Norwegian, and Portuguese, but extend also to Eastern European languages like Hungarian, Czech, Polish, and Romanian, as well as to languages written in the Cyrillic alphabet, like Russian and Ukrainian, and to Japanese Kanji. C-Kermit is distributed in C-language source code form and selected binary formats by Columbia University, with a price range from $45 to $220 including documentation.

Kermit Distribution, Columbia University, 612 W. 115th Street, New York, NY 10025; 212/854-3703. Δ

Circle 52 on reader service card.

DASH Items...

Catagory: DG/UX

Author: Rodney Wright
Subject: **Making copy of QIC150 tape**

I would like to make a copy of the DG/UX contribution tape, but am having little success with *tar* or *dd*. I would like to dump the entire image of one 1/4-inch tape to another 1/4-inch tape. Anyone with more experience either with the contribution tape or with *tar*/*dd*, I'd appreciate your help.

Reply by: Pierre Asselin

There are many files on the tape (separated by file marks). *Dd* or *tar* will only look at the first file. The first two files or so aren't even *tar* files, so *tar* won't work.

Find out how many tape files there are, either from the release notes or with *sysadm lstoc*, and try something like this:

```
$ i=1; while [ $i -1t NN ]; do
> dd if=/dev/rmt/1n of=/dev/rmt/2n bs=???
> i=expr $i + 1
> done
```

(a Bourne shell loop). *NN* is the number of tape files. I don't remember what block size (bs=) I used when I copied my tape; probably 32b, or whatever the *tar* default is. Be sure and use the no-rewind tape devices (*rmt/ln instead of rmt/l*) or you will repeatedly copy the same file.

When the loop finishes, rewind the tapes:

```
$ mt -f /dev/rmt/1 rewind
$ mt -f /dev/rmt/2 rewind
```

Try out the copy! Either with *sysadm* or, manually, *mt fsf* to forward space to some tape file and *tar tvf /dev/rmt/x* to see if the *tar* file is any good.

Author: Mark E. Bouver
Subject: ***Setuid* shell scripts**

I would like to write several shell scripts to allow users to do various privileged functions on occasion. I am familiar with shell programming and have done this sort of thing before by writing a shell script and then simply setting the permissions to 4755 (owner is root or some other privileged user).

The problem is, *sh* (Bourne Shell) and *csh* (C Shell) both seem to ignore the *setuid* bit. The scripts seem to work fine when invoked by *ksh* (Korn Shell).

I can work around the problem by using *ksh* as mentioned above, but what is causing this strange behavior? Has someone at DG decided that *setuid* shell scripts are a "bad idea" for security reasons?

Reply by: Brian Fillette

csh requires the *-b* option for *setuid* scripts.

Reply by: Allen Barkley
Subject: **DG/UX 5.4**

I would like to be able to obtain a "line count" on a data file in a way which can be used directly in a script file. I can obtain the line count manually by using *vi* (the line count appears at the bottom of the screen), but I have not been able to invoke *vi* along with *grep* or *fgrep* to strip out the line count information I need. Are there any other commands I can use to obtain the line count?

Reply by: Beneficial Life Ins. Co.

You can use the *wc* command. The command *wc filename* will give you a response with 4 fields (line count, word count, character count, file name). If you need only the line count, use *wc -1 filename*, which will return the line count and file name. With whichever method you decide to use, you can pipe the results to cut and pull out the fields that you need (*sch* as field 1, the line count).

Category: AOS/VS

Author: Jim Hathaway

I'm having a problem with a Megatape under AOS/VS 7.70. The same tape unit worked fine under 7.69. Now I can write a 32 K block with no error, but I cannot read that same tape back using 32 K on the "load_ii" command. Any ideas? Unit works fine if I use 8,192 blocks on write and read. Sysgem has 32 K as max transfer for this tape setup as MTD dev 62.

Reply by: Tim Boyer

Which Megatape unit, the 8 mm? I believe there's a patch for 7.70 (GIGATAPE_OPAT sticks in my memory—something like that). I'm running a Megatape 8 mm under 7.70 with that patch installed, and have had no problems, but I'm set up as a data channel device (MTC). Δ

DASH runs on an Aviion 5200 server located at the Customer Support Center in Norcross, GA. The bulletin board is available 24 hours per day, 7 days per week, free of charge. Call 800/DASH-CSC (800/327-4272) for the modem rotary.

Bits and bytes...

Asynchronous communications

From: Mark Pagano

I am currently working on an MV/30, and I work with a homegrown program that controls data being input from a microprocessor via modem through an IAC-8. Every so often, the program seems to hang, and I need to use CLEAR/RXON to clear the port. I am thinking a CTRL-S is getting into the IAC. Is there any way to disable all CTRL characters? I tried using ?KIOFF, but the problem still exists.

From: Walter Mosscrop

"?KIOFF" disables the ^C^A, ^C^B, etc., sequences—it has *no* effect on control characters such as ^S and ^Q. What you probably need to do is use ?SECHR to set the ?XOFC bit off in the fifth word of the extended characteristics. This will prevent the IAC from interpreting incoming ^S characters as "stop sending" characters. You may also try binary ?READs. Δ

The NADGUG/Rational Data Systems electronic bulletin board is available to NADGUG members. Call 415/499-7628. No fees other than phone charges.

SCOTT McINTYRE & ASSOCIATES, INC.

"YOUR GLOBAL SOLUTION FOR D.G. & MOTOROLA"

313-299-0040

*** BUY * SELL * TRADE * DATA GENERAL & MOTOROLA ***

PRICE!

"It is with great pleasure that we have chose to work with Scott McIntyre & Associates, Inc. as our second source Data General supplier. The prices, service, delivery time and quality of the equipment we've purchased has never been equalled!"

Amaizo Yves
Managing Director
Sysinfo
Jauldes, France

PERFORMANCE!

"Scott McIntyre & Associates, Inc. has earned the respect of all the executives at our company for their honesty and diligence. They have always been flexible and accomodating!"

Andy Michael
Sales Director
Quiss Tech., Plc
London, England

QUALITY!

"Many other resellers in the USA would like to deal with us in Australia, and on some occasions we have brought from them. However, the quality of the goods and service has never matched what we get from Scott McIntyre & Associates, Inc."

Alan Frazer
Managing Director
Frazer & Associates PTY. LTD.

DELIVERY!

"It is good to know that there are still people out there who conduct their business with honesty and integrity."

William E. Herr
Director of Computer Operations
The Bon-Ton Store

VISA MasterCard AMERICAN EXPRESS

"JULY SALE-A-BRATION!"

"NEW" AV4300 W/16MB, 525MB CTD & 332MB DISK ONLY $7,720

6591 2GB CTD IN PHU ONLY $2,350

"NEW" 6841 2GB 3 1/2" SCSI DISK "STOCK!"

6631-A 600MB RAMS DISK ONLY $3,875

6299 1600/6250 BPI TAPE ONLY $1,250

D217 CRT W/KB ONLY $225

"Over $1,000,000 in stock. Call for our summer price catalog!"

"Remember the children this summer please drive safely"

"Call Today"

***Scott McIntyre * Terry Smith ***
*** Robert McIntyre * Michael Nowacki ***

313-299-0040

FAX: 313-299-0041

P.O. BOX 82115 • 1856 STAR-BATT DR.
ROCHESTER, MI 48308-2115

Circle 40 on reader service card.

Shop Our Million Dollar Data General Inventory And Get "Right-On-Time" Delivery.

• Million dollar inventory of systems, peripherals, parts and equipment • From Nova to MV -- you get "Right-On-Time" delivery • Highly skilled engineers • Valuable assistance, consulting, training and installation • Customer service reps with broad product knowledge • Problem solving advice • All backed by RAVE's 60-day warranty

• Sensible business solutions with significant bottom line value •

SELL • BUY
LEASE • RENT
Data General &
Compatibles

We Also Sell Sun Microsystems

Sun Microsystems Computer Corporation
Authorized Refurbishment Reseller

Announcing -

RAVE FINANCIAL SERVICES, INC.

- Complete In-House Financing
- Custom Leasing Packages
- Long or Short Term Programs
- Rentals Available

Rave Computer Association, Inc.

36960 Metro Court
Sterling Heights, MI 48312
(313) 939-8230 / Fax: (313) 939-7431
International office
(516) 929-5000 / Fax: (516) 929-5007
1-800-966-7283

Circle 23 on reader service card.