$1.50

PORTLAND ATARI CLUB

JANUARY 1987

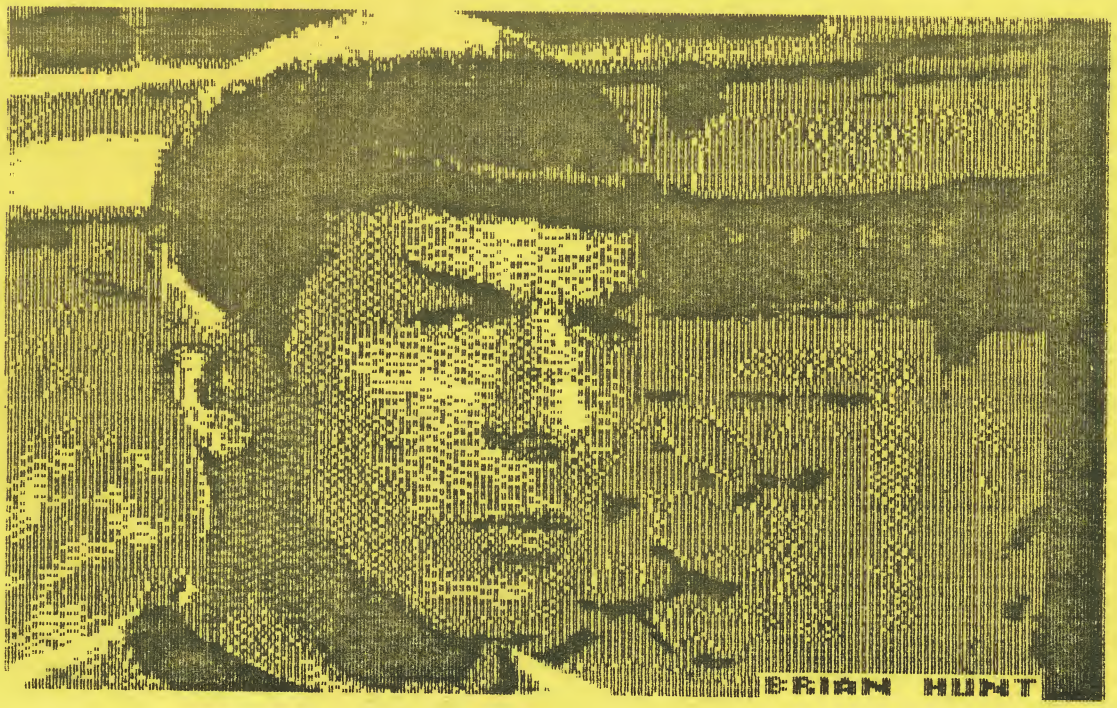
IN THIS ISSUE

NEXT GENERAL MEETING

Monday, January 5, 1987, at 6:30 p.m.
Northwest Service Center
1819 N.W. Everett St.

PAC Bulletin Board Systems
24 Hours - 7 Days a Week

#1 - (503) 245-9405 - 300/1200 BPS
#2 - (503) 245-4608 - 300/1200 BPS



HAVE A LOGICAL NEW YEAR !

P O R T L A N D A T A R I C L U B

(Not affiliated with ATARI, Inc.)

PAC Board of Directors

President:	David Holliday	642-4717
Vice President:	Verne Newsome	
Secretary/Treasurer:	Dan Gibson	665-6912
Membership Secretary:	Jim Miller	641-6356
Program Director:	Dean Nickel	639-2678
Club Librarian:	Jerry Andersen	655-3914
Special Interest Groups:	Richard Cowger	641-2782
Special Projects:	Michael Whiteley	635-4717
Sergeant-At-Arms:	Randall Leong	246-6354
Newsletter Editor:	DeLoy Graham	649-6993
BBS Director:	Steve Billings	246-1751
Advisor:	Vern Vertrees	647-2855

PAC NEWSLETTER STAFF / SIG MEMBERS

DeLoy Graham, Editor
Steve Billings - Art Director
Debbie Billings - Art Assistant
Lee Gassaway - Ad Coordinator
Chuck Hall - Columnist
Peter Hoesly - Mailing Assistant
Clyde List - Cartoons
Clyde Pritchard - Final layout
Debbie Pritchard - Production Coordinator
Kevin Roethe - Mailing Coordinator

This newsletter is written and published by members of the Portland Atari Club (PAC), a group of people with a common interest: the Atari Computer. All articles are written and donated by members or are reprints of public domain material from other groups. Opinions expressed are those of the authors and do not necessarily represent the opinions of PAC or those of any other organization. Material appearing in this newsletter may be reproduced for non-commercial use, providing credit is given to the author and PAC or other originating group. Commercial use must be coordinated through the editor. Material for publication may be submitted in the following formats: ST text files, Atari DOS II text files, LJK text files, printed, typewritten, or legible handwritten copy. Media may be sent to the editor at the address below and will be returned to the submitter. Contact the editor for instructions on uploading submissions to the PAC BBS.

Commercial Advertising Rates: full page (7 X 9 1/2) - $50, half page (7 X 4 1/2) - $25, quarter page (3 1/4 X 4 1/2)- $15. Ads must be prepaid and a 1/3 discount is given for 3 consecutive ads. The copy may vary in content, but the space must be the same in each issue. Send camera ready copy and check payable to PAC at the address below. Ad deadline is the 5th of the month prior to publication. Please **contact Lee Gassaway** (591-5252 or 642-2455) on all matters pertaining to advertising.

Membership is $20 per year and includes a subscription to this newsletter and access to members-only functions. Single copy price of the newsletter is $1.50. General meetings are open to the public and start at 6:30 p.m. on the 1st Monday of each month (2nd Monday in the case of holidays) on the date and at the location listed on the cover of this newsletter.

Exchange newsletters, articles, correspondence and ads should be sent to the following address:
Portland Atari Club, Attention: (appropriate board member), P.O. Box 1692, Beaverton, OR 97005

Printing done by Hillsboro Quick Print, 435-B S.E. Washington St., Hillsboro, OR 97123, 640-3649

PRESIDENT'S COLUMN

Dave Holliday

Well, I guess it really is happening. Somehow I ended up becoming your new president for the coming year. I guess only time will tell if it's for good or bad.

Before I get too far along here, I do want to say one thing. That is how much I personally appreciate the work Vern Vertrees and the other board members have done this past year. We had a good booth at the May Electronics Show, the October Atari Expo was just super, new members are joining every month, the newsletter is always good, and the SIG groups seem to be continuing well. All these things don't occur by magic. It takes hard work, some long hours and a lot of devotion to the club to make it all happen. I want all the board members, especially the retiring ones, to know that all your hard work is not lost even if the rewards at times seem small. I hope that this year's board can continue with the good work that you have provided the membership this previous year. Thanks.

Now, what do I have planned for the club in the coming year? Nothing. Well, let me qualify that. I have some ideas that I would like to try but at not time is this my club. You the members are the driving force in what gets changed or done. The other board members and myself can provide the leadership, but it is going to take your input and maybe some of your sweat to make it work.

One change I would like to have you consider is making all memberships due in July. Those currently due before or after would be prorated. We would then have a good idea of operating funds for the next 12 months and an accurate budget could be written up and given to the members for approval at the September meeting. We currently have over $12,000 in memberships alone. This seems like too much money to be managed piecemeal.

I would also like to see a wider dissemination of new software to be evaluated for meetings and the newsletter. We seem to have the same people always having to do it. They do an excellent jobbut, hey, everybody needs a break. Remember that sweat I mentioned earlier. I don't think a few hours of playing Silent Service and writing a two or three paragraph blurb on what you thought of it is all that terrible.

We are a very large club. As such we have to change. I can think of other large things that didn't change — the dinosaurs, for instance, and look how they ended up. True, the club can't be all things to all Atari people, but we can certainly try to get the most out of the people and resources we have or else there is no reason for existence. I feel the club and its membership are like a body. If only a few parts get a workout, the body is not operating at its potential. But, if all parts are given an occasional workout, maybe some to a greater degree than other, the overall body remains healthy. I hope you look at me and the rest of the board as a kind of Jack Lalaine. We'll workout with you, but we aren't going to do it by ourselves.

One change I'm going ask the rest of the board to approve is that we postpone the raffle for the nice pieces of hardware we have until the February meeting. The reason is that we didn't get all the software auctioned off at the December meeting so we want you to have a chance to save up your bucks to buy a ton of tickets. Plus there are some additional things I would like to look into about the raffle and having it in January doesn't give me enough time to get back to you.

I've written enough, I suppose. After sitting down and writing this I guess I really am looking forward to the coming year. I hope to be hearing from you about all those good ideas you have for the club. Better still, I hope to hear a lot of "I'll do it." Well, more next time. And 1, and 2, and come on work those muscles, and 1, and 2. . . .

****

IMPORTANT DATES

Newsletter Deadline	January 10
Board Meeting	January 28

****

EDITOR'S COLUMN

R. DeLoy Graham

A lot of our deadline problem in the past has been that articles are turned in too late. In order to speed up the process, I am not going to wait for articles anymore. The deadlines I print in the newsletter usually fall on a Saturday, because that is about the only free day I have to do the editing and preliminary layout. A lot of people wait till that day to submit their articles. The problem I have is that I need the articles during the day, not at 11:00 p.m. on that night, after I've completed printing the final copy. If you have something to submit, please get it to me before the deadline. The more time I have to spend with it, the better I am able to do my job. If I have to throw a lot in at the last minute, I often miss things. From now on, articles received after the deadline will be held until the next issue is prepared.

Now that PAC BBS #1 is hard-drive based, I would prefer you upload your articles there. Upload them as text in the newsletter SIG. Then leave me a message in E-Mail telling me what you have posted. Don't expect me to download everything just to see if it is for the newsletter. I am using Regent Word, but I can generally load anything into it. If uploading from an 8-bit Atari, please use ASCII mode rather than ATASCII. Please do not format your text. If using 1st Word, use non-wordprocessing mode. Especially do not justify your submissions. Many times they must be trimmed to fit available space, and any change to a justified document creates a mess.

As we begin a new year, I would like to remind the members of the Board that each of them is expected to write something about their responsibilities. In the past, few have done so. The members have a right to know what is going on in the administration of the club. Please accept this responsibility. Better yet, set a proper example for others by submitting your articles for the next newsletter at the monthly general meeting.

WHICH BBS IS BEST?

S. Lovejoy, P. Warnshuis
S. Billings, D. Lindquist
Downloaded from IBBS

Stephen Lovejoy: I am looking for general opinions of Pay-for-Play BBS's, those national boards that supposedly support ALL computers and users. I would like to hear pros and cons regarding the likes of CompuServe, GEnie, The Source, and Whathaveyou. I am interested mainly in the bang/buck ratio and ST support (programs, hints and tips, SIG's). Ease of use is also a big point, since I am trying to get my computer illiterate wife interested in this. All opinions are welcome.

Pat Warnshuis: I find CompuServe most useful. However, it is both expensive and difficult to become proficient on. I still resent a separate rate for 300/1200 baud! However, its message and download sections are terrific!

The best bang/buck is this board (IBBS)! Usually the best of CompuServe and others will show up within a week. You have to spend time on other boards to appreciate how much effort Russell Schwartz has invested in this one. I do find many picky faults with FoReM, but the price is right!

Steve Billings: I have recently signed on to Genie after having been a loyal CIS user for a year or so. After having used Genie only a couple of times and mostly stumbling around, I am impressed. It has a SIG devoted to the ST and once I figure out how to figure out the message base, it looks very informative.

There is a fairly extensive download file section for the ST, and best of all the rate is $20 for sign up and $5 an hour at 300 or 1200 BPS! This is a preliminary opinion since I have not used it much, but I spent too much money on CIS, so GEnie looks pretty good.

Dave Lindquist: I have not tried GEnie, but have been a user of Delphi for some time. Delphi has pretty good support for Atari, both 8 and 16 bit. Most of the PD programs that are on CIS can also be found on Delphi. The message base is a little harder to use than CIS (you cannot search by subject), but they do have a large selection of Degas files. The best part is that the cost is only $6.00 per hour for 1200 baud. I use both Delphi and CompuServe, but I think CIS charges too much at 1200 baud.

ELECTION RESULTS
Debbie Billings

The PAC election was held December 1 for all board positions except Sysop and Editor, which are appointed. It was unanimously decided to elect by acclamation all uncontested positions. Only the position of ST Librarian was contested and decided by vote, with Dave Roberts receiving 24 votes and Russell Schwartz receiving 40. There were 67 ballots cast. Our 1987 Board of Directors are as follows:

President:	Dave Holliday
Vice President:	Verne Newsome
Secretary/Treasurer:	Dan Gibson
Membership Secretary:	Jim Miller
Sergeant-at-Arms:	Randall Leong
SIG Leader:	Rich Cowger
Special Projects:	Michael Whiteley
8-Bit Librarian:	Jerry Andersen
ST Librarian:	Russell Schwartz
Program Director:	Dean Nickel
Newsletter Editor:	DeLoy Graham
BBS Director:	Steve Billings

Thank you to all who participated in the election.

MEMBERSHIP NOTES
Jim Miller

I wish to welcome the following new members and families to the PAC.

PAT BEHRENS	DON THOMAS
KAREN SUTTON	CALVIN PARTRIDGE
LYNDON O'BRIEN	JIM BELL
RICHARD HILL	RALPH HORN
RALPH O'HARA	LINDA HELSABECK
BILL MCCORKLE	AL SIEBERT
WILLIAM EPPLEY SR.	

I wish to thank all those who chose to run for office. Hope to have another good year, and I promise not send any of the dues to Iran. I would be interested in teaching someone this position so you could run for this office next year. Please let me know if your interested.

DEALERS CORNER

***Computerola**
6224 SE Main (residence)
Portland, OR
239-4315

Computers, Etc.
11504 E. Mill Plain Blvd.
Vancover, WA 98684
(206) 254-5849

Computers, Etc.
12145 NE Halsey
Portland, OR 97230
252-0179

***Computron**
1139 SE 11th
Portland, OR 97205
224-2220

Computron
11705 SW Pacific Hwy
Tigard, OR 97223
639-6780

Creative Computers
3275 SW Cedar Hills Blvd
Beaverton, OR 97005
644-1160

High Tech Toys
2865 SW Cedar Hills Blvd
Beaverton, OR 97005
646-3950

****IB Computers**
1519 SW Marlow Ave
Portland, OR 97225
297-8425

Toys R Us
Jantzen Beach: 289-4691
Tigard: 620-9779
Milwaukie: 659-5163

NW Telephone Systems
Residence
ST software only
282-6223

* Discount is available to PAC members.
** Monthly specials for PAC members.

Note: Some stores are ST dealers only.

Authorized Service Centers

Micro Care
1447 NE Sandy Blvd.
Portland, OR 97232
230-0770

NW Computer Support
10200 SW Nimbus, G1
Tigard, OR 97223
684-3280

Computron
1139 SE 11th
Portland, OR 97205
224-2220

IB Computers
1519 SW Marlow Ave
Portland, OR 97225
297-8425

EVERYTHING YOU EVER WANTED TO KNOW ABOUT HOW TO WRITE A REVIEW

Mike Fulton, President and Editor of ACAOC

Reprinted from the October 1986 **ORNJUCE** Newsletter

Ok, first realize that you don't have to go out and buy something just for the sake of writing a review. I don't want people to tell me they bought something just to write a review of it, and that now they don't really like it or don't need it. The idea is to buy just what you want, and then write about it. Got that? Good.

The process of writing a review is actually pretty simple if you break it into steps. Ok, so maybe you aren't the greatest writer in the world. But when you think about it, only one person, somewhere, is. You don't have to be a John Steinbeck or Robert Heinlein to write a review. Get the basic ideas down first, then you can worry about making it sound pretty. Or I can. That's what editors are for, right?

Obviously, you would not review a game program the same way as you would a sophisticated database program. But many of the same ideas apply to both. Here are some simple rules that apply to writing a review of just about anything.

1. Before starting the review, make sure you have used the program. The more of a beginner you are, the more important this is. How many times have you read a review of something you use all the time where it seems like the reviewer didn't look at half of the program? Don't let this happen to you. Make sure you have used every major aspect of the program.

2. Read the manual! Read it twice if necessary. It's amazing how many problems and misconceptions arise from people not reading the manual. Reviewers are no different from anyone else in this respect. Sometimes a program will have hidden functions which can only be learned about through the manual. Also, remember that it's usually necessary to review the manual somewhat, as well as the program. You can't do that unless you read it.

3. Make sure you know what the program is supposed to do. When wide gaps exist between this and what a program actually does, take note of it. Equally important is noting when a program goes above and beyond the call of duty.

4. Give note to the user interface. Note how the program is controlled. For a game program, this might mean the reaction with the joystick. For something like a database or word processor, it might mean the menus or commands used. How fast is it? Or how slow? How easy? How hard?

5. If you don't like the program, realize that not everyone will share your opinion. You aren't trying to talk people out of buying something, you are trying to tell them enough about the program so that they can make up their mind by themselves. It's ok to say that you don't like something, but you should be very specific as to why that is. Someone else may not care about that particular thing.

6. On the opposite side of the coin from #5, if you really like a program, it's ok to say so, but be certain you get across the idea of why you like it. If it's possible that someone won't share your opinion, consider that. You don't want a whole bunch of people coming to you later complaining how you misled them about how great a program was.

Ok, now let's go over the major components of a game review as an example. How you open the review depends on the type of game and your own personal style, and there are no absolute rules about it. For example, for a chess game nearly everyone will already know the basic idea, so the beginning might concentrate on what makes this chess game different from other ones. But for a program like a graphic adventure game, you might give a brief description of the game's scenario. What does the hero have to do to save the day? And so on. Most games will fit the second idea better than the first, but do whatever works best for you.

If you don't do it in the opening, the next step is often to tell who makes the game, and what systems it works on. Now you might tell what the game looks like. How many people can play? What music effects are there? What controllers does it use? How many players? What options are there? Any problems with anything?

Towards the end of a game review, you should tell a little about the program manual, if the program is copy-protected, etc. How much does the game cost? What did you really think about it all? Was it good, or bad? In what way? Where did you buy it?

You may notice that there are a lot of questions to be answered in a review. Well, that's what a review is, the answers to the questions of what a program is like. You may find it helpful to write down a small list of questions about the program before starting to write.

I hope this helps some of you to try your hand at writing something. In fact, I'm depending on it. Don't wait for the other guy to do something, because it might be a long wait.

WHAT'S WRONG WITH MEETINGS

Chet Cox, Atari Club of Denver
Reprinted from the November 1986 **Mile High Atari Magazine**

There seems to be an overwhelming wave of boredom at the Atari Club of Denver meetings [might we not insert "at the Portland Atari Club meetings"? -- Editor], and it's not anyone's fault. Member's sit like couch potatoes [or, as Dave Holliday says they're called in California, sofa surfers], watching and waiting for something to happen. Few volunteer to "shake and move," and less than 2% responded to a survey. What do folks want from the club?

Perhaps you've never asked yourself that question. Many of you don't go to meetings for various reasons: too far, too late, I just joined for the newsletter. I can't blame you. If it weren't for my insufferable curiosity, I wouldn't go either. . . . I believe that you're all just as excited about Atari -- whether the 8-bit or the ST. So why are meetings so hum-drum?

First of all, real computer bugs tend to be loners. (Might make a good bumper sticker: "Real hackers don't socialize.") . . . Secondly, there's nothing exciting about the administration of the club (or about the administration of anything, Ayn Rand to the contrary). And thirdly, what's there to do at a meeting -- watch a demo? How very interactive, and how easy is it to see that little see that little screen at any distance?

I propose various subroutines to be added to meetings which might make them more interactive.

Demonstrations are a terrific idea. Small print on a small monitor further away than 18 inches is not a good idea. Only colorful, graphic demos are discernible at that point; for demos of programming techniques, can we get the use of an overhead projector, or even an easel? The demonstrator could put it on that instead of a tiny screen. Cox's First Law of Briefings: "If you can't see it, you aren't interested."

We need to find out if our members want technical demos ("Here's a demo of Darnitdos."), fun demos (Here's a demo of Bounty Bob Vs Donkey Kong."), tutorials ("Here's how to simulate chocolate string arrays in ATBASIC.") -- or if they'd rather demos weren't at every meeting, or even if they are tired of demos.

Redundancy is a problem with any demo or tutorial given. There are always new users, and any experienced user is going to be bored by the very things that fascinate newbies. (I, for instance, am fascinated by ACTION!, and would love a good tutorial on it. Each ACTION! programmer, however, is first a programmer, not an instructor, and is moving on to new things instead of going over the basics again and again. Thus, no ACTION! tutorial exists.) Also, a person who is a good blend of knowledgeable programmer (or hardware hacker) and entertaining speaker is rare. Presentation is an acquired, learned skill; few (if any) are born with it. Businessmen are paying Big Bucks to be able to present ideas (any idea, no matter how great, is worthless if poorly communicated). Teaming a speaker with a programmer may be an elegant solution to this problem.

Our meetings could take a hint from science fiction, comics, stamps, and coin clubs. There, members not only meet, they swap and sell each meeting. Turnout is excellent, and interaction is at the point of chaos at such club meetings. The worry here seems to be that everyone would shop and no one would attend to business. And this is unseemly to the person to whom computers represent organization, especially in programming (Of course, they've never looked at any BASIC program I hobbled together, or they'd run screaming!). Though the club allows occasional swap meet meetings now, they're intended to be no more often than twice a year ("mebbe three times"). My position is that a little chaos never hurt anyone, and always keeps things interesting. Note the large turnout during the previous swap meet. Proposing a "Huckster Night" every other meeting would, IMHO (In My Humble Opinion), would be a Good Thing.

Club meetings are basically so people can get together and talk Atari to other Atarians, without the hassle of people who hate or fear computers. If people are permitted to just mill around and talk, the meeting is still successful. Let's see a little less businesslike atmosphere.

COMDEX SHOW REPORT

Neil Harris, Atari
Reprinted from **ZMAG**, 18 November 1986

It was easy to visit all the stations within the Atari booth -- if you are a fullback. The Atari booth was jam packed. One measure of the show's success: I have the biggest stack of business cards I ever got at a show. Why was the Atari booth so crowded? It is not so easy to answer this question, even after so many people in other booths asked it. (True story: three folks with shiny golden IBM badges came to the Atari information counter and asked us to send over some people! You could park a truck in IBM's booth.) My theory is that we were the only one with something different. All alone in a sea of clones...

The award for the most unique exhibit goes to the Video Touch people. They displayed a system using a touch-screen that replaced a cash register for restaurants. It made nice use of ST color graphics to display the different menu items. The final order receipt was printed on a teensy little printer. This looks like a solid commercial application -- the local McDonald's franchisees were reportedly very interested.

Altogether there were more than 50 developers sharing 36 stations within the booth (some folks had to share space on a morning/afternoon basis). The theme at the booth was "Atari Means Business." We were there to show the industry that the ST computers are powerful systems with all sorts of professional applications. Desktop publishing was a buzzword on the show floor and good programs for it were in abundance for the ST. CAD was another big area, along with development tools and personal productivity.

The major planned attraction at the booth was the center aisle, which held 12 companies. Each setup had a 1040ST, a hard drive, and a laser printer. The printers were loaned to the booth by QMS, Hewlett Packard, Canon, and others. The software on display included desktop publishing, CAD, and graphics. Two WYSIWYG publishing packages, **Publishing Partner** from Softlogik and **Fleet Street Editor** from Mirrorsoft (a U.K. company), attracted quite a bit of attention from industry experts. Also showing was the new **Typesetter Elite** from XLent and the command-driven **LaserType** from Softlab.

The folks from Progressive Computer Applications had their **Graphic Artist** program working with a huge Houston Instruments plotter, creating all kinds of architectural designs, page layouts, and pictures.

In the CAD area, Foresight Resources had a very nice package called **Drafix I.** This program was ported from the PC-compatibles. According to Foresight, it runs as fast on an ST as it does on an IBM PC-AT running at 8 mHz with math coprocessor! Generic Software's **First CADD** was also a popular exhibit, as was Abacus Software's PC Board Designer.

Video Digitizers also showed up in force. Digital Vision finally unveiled their **ComputerEyes** color digitizer, which has software to control the contrast, brightness, and color mix. Michtron's booth had the digitizer from Printechnik (an Austrian company), and Hippo had their's as well.

Color graphic programs were shown by Aegis (the ST version of **Aegis Animator**), Batteries Included (**DEGAS Elite**, shown by its author, Tom Hudson), and Antic (with **CAD 3-D** using their forthcoming 3-D liquid-crystal goggles).

The outside corner of the booth showcased MIDI applications and was shared by 4 software vendors: Hybrid Arts, Electronic Music Publishing, Beam Team, and MidiSoft.

The lighter side of ST computing was displayed by Activision, Electronic Arts, Michtron, and FTL Systems. In particular, this was a first for EA, which now has a dozen ST titles in the works for release in the next few months. They were demonstrating **Skyfox** and **Chessmaster 2000.** England's Eidersoft were showing another kind of 3-D with their games, using red and blue lenses -- not as high tech as Antic's, to be sure.

Productivity applications were shown by TimeWorks (**Swiftcalc** and **WordWriter**), Abacus (**PowerPlan**, **TextPro**, and **DataTrieve**), Quickview (**Zoomracks II**), **Maxthink**, and Royal Software (**EZ Calc**). Royal is adding a new desk accessory to **EZ Calc** which opens a separate window for graphs -- drag the numbers from the spreadsheet and they are automatically graphed in a different window. Very slick. Infocom was also there, showing a pre-release version of **Cornerstone** for the ST. They were "testing the waters" to see if this program was wanted by dealers -- the reaction seemed very strong.

Program development was represented by Mark Williams (**MWC**), Metacomco (**Lattice C**, **Cambridge Lisp**, **Macro Assembler**, **MCC Pascal**, **BCPL**), Computer Tools International (the **IDRIS** operating system, with the Whitesmith compilers), and Beckemeyer Development (**MT C Shell**).

Educational software was displayed by First Byte (four programs in the new **SmoothTalker** series featuring speech synthesis), Unicorn Software, and by Atari Corporation, which has licensed programs from Arrakis.

Speaking of Atari, we were showing the SX212 modem, Microsoft **Write**, the VT100 emulator, and the 1.0 version of **NEOchrome**, as well as a 1040ST with the "blitter" chip. There were many rumors of new hardware products that had been expected at the show, and there was some speculation that some of these were in fact shown off in the hotel suites of Atari executives. These rumors are neither confirmed nor denied. <grin>

In all, the software shown at the booth demonstrates that programmers are learning to make full use of the ST's speed and graphics. The professional applications stood up well next to their more expensive cousins running on PC clones and other systems throughout the show. Just for fun, the folks at Xanth brought along their new game **Kill a Happy Face**. At 5:00 on Thursday afternoon, they hooked up a cluster of 6 ST's with MIDI cables for an all-star shootout. This game is similar to the **Maze War** game on the Mac, with smoother animation and full color. Players see the maze from their own perspectives, with other players appearing as brightly colored "happy faces". If you see their faces, they can see you too; otherwise you're looking at their backs. Three hits on a face kills it, and ten kills wins the game. A great party game. I even won once, as did Tom Hudson and one of the DUST user group members.

No show report is complete with just a list of products. The many personalities in the ST world are what really makes these shows special. Gary Yost of Antic spent most of his time earning his title of "Mister Enthusiasm" while showing the 3-D animation in the center of the booth. Gerry Humphrey of Haba added to his legend as a flake by wearing his famous moose-horn hat to the booth as well as the parties. Tom Hudson, when not being a DEGAS virtuoso, managed to play Tempest in an arcade for an hour on a single quarter. Jeremy Berger of NYACE, who was showing a 1040ST-to-composite-video adapter, proved that a 16-year-old could gamble in Las Vegas. Scorpio Systems earned the nickname of "Bimbosoft" by their promotion of a craps-tutorial program, featuring a young lady who could serve as a good source of shade on a sunny day.

Another great show for Atari. Next stop: the Consumer Electronics Show (CES), just after New Years, back in Las Vegas.

PAT'S HINTS

Pat Warnshuis, PAC

Editing BBS Messages with FLASH

You'll save on-line time and enter more literate messages if you compose and edit them off-line with an ASCII editor (such as Jerry Cole's **PROED**). Don't forget to end them with '/S'. When the BBS prompts you to enter the message, simply press 'Alt-U' to upload an ASCII file. Then click on your saved file.

If you name your message something like 'OMSG1.DO,' it will be at the top of your dialog box for file selection. Your ASCII upload options probably are still at the default pause of '1' which works for a slight pause at the end of each line to allow the BBS editor time to absorb the line. ASCII uploads are echoed back to you so you can monitor the message as it is sent.

Desk Top Trick

Did you see the cute little trick Gary McAllister uses for his desktop directories? Within a crowded folder, he renames his most often sought for file with its original name but puts a '0' before it. That forces the desktop to list that file first in the window so Gary can find it immediately! That works just great for folders such as FLASH, which has beaucoup files and you always want to execute FLASH.PRG. Just rename it OFLASH.PRG and it'll always be listed at the top of the window when you open the folder. Gary uses this trick to find his execute file within a folder full of pictures also.

Sure, it's obvious--after he thought of it!

****

THANKS FOR YOUR HELP AT THE ATARI EXPO

Larry Gold, ACE

For the many of you who participated in making our Expo a success, I am reprinting an excerpt from an article by Larry Gold, Vice President of Eugene ACE:

"We had an Atari Expo in Portland and it seems to have gone over pretty well. Atari had a booth manned by Atari executives. Along with them there were 3rd party vendors and various other people showing their wares. It was quite good for a first time event in the Northwest. If it becomes an annual event it will be because of the Portland Atari Club which put it on. My hat goes off to you for the fine job you did."

HAPPY HOLIDAYS FROM MIGRAPH!

FAST™.....$49.95

Here's the desktop accessory you've been waiting for! FAST contains several of the most popular programs all in one: ST DOS, ST Editor, a Card File plus a calculator, calendar, ASCII table and clock. Many of FAST's perameters can be set to meet individual needs and it operates in all three resolutions. Once installed, it is accessible from within any GEM program.

LabelMaster™.....$39.95

Once you try LabelMaster you'll never use plain labels again! This great program is GEM based and containsa Graphic Editor to create your own label designs and a mailing list manager. Packed with features plus over 100 ready to use label designs. The possibilities for home and business are endless.

Toall S. Towners
Attn: Happy Holidays
123 Some Ave.
Anywhere, USA 12345

Easy–Draw® 2.00.....$79.95

New features and a new price make Easy–Draw a great gift for Christmas. Version 2.00 contains all the powerful features in version 1.03 like object oriented graphics, pop up menu, specific area zooming, and rubberband size and stretch, PLUS:

Flip & Mirror, Single GEM file format, Edit Polyline, New Change Arc and Arrange commands, and Automatic copy between windows!

With Easy–Draw you can create anything from simple graphics to technical illustrations, and add lots of text too!

MORE EXCITING PRODUCTS COMING SOON!

To Order see a Dealer near you or call MIGRAPH at:

1(800) 223–DRAW or (206) 838–4677

This ad created entirely with Easy–Draw, except for the label.

WHAT ARE THEY TRYING TO DO TO ATARI?
Is Piracy the Real Issue?

R. DeLoy Graham, PAC

I generally avoid the numerous tirades about software piracy. The reason for my lack of interest is neither a feeling that piracy is not a problem nor a feeling that piracy is okay. It is simply that I have already decided that I will purchase the software I use. I have not even complained much about the many hundreds of dollars of software I have purchased that has proved to be of little use to me.

But recently I've noticed something as I've skimmed over piracy articles -- there seems to be a concerted effort to discredit the Atari computer and its user. The comments I hear and read in piracy discussions are only a small part of this effort, but I'm getting tired of it. I am offended by the name calling.

For this reason, I have retyped the accompanying article about software piracy. I appreciate Mr. Haynes' thoughts. Even if you don't agree with him, you should find something in his article to mull over.

I specifically want to say a little more about the comment by Bill Stealy that Mr. Haynes quotes: "Atarians . . . are the biggest pirates in the world." As a computer educator in secondary schools for the last four years, I have been acquainted with many Apple and Commodore users. Over the years I've heard many of these users brag about their pirated software. I've even known a few Commodore users who have told me they have never purchased a single software program, even though they have hundreds of them. I've heard them talk about pirate BBS's that they either run or frequent.

During this same time period, I have also been acquainted with Atari users through the Portland Atari Club and ACE of Eugene. Although I have not looked for it, I have never seen such blatant piracy in the Atari community as that which I have seen in the Commodore and Apple communities. I'm not saying we are guiltless; I'm just saying that to call us the biggest pirates is unfair and, in my opinion, unsubstantiable and untrue. And considering that the size of Commodore and Apple communities is so much larger than the Atari community, I am beginning to wonder why so many stones are being thrown at us. Is this part of the greater conspiracy against Atari?

I am not the only one who feels this way. Here's an excerpt from an article by Jim Woodward, printed in The Pokey Press!, November 1986:

"I am the assistant manager of The Electronics Boutique in the Boynton Beach Mall (a great place to buy software, end of commercial). This gets me close to lots of Commodore, Apple, Macintosh, Amiga and IBM owners. And guess what? Atari owners aren't the biggest pirates in Palm Beach County, Commodore users are. I wish I had a dime for every time someone said, "I got this game from a friend!" or "My friend George gave me this one!" or "This is out now, my friend John gave it to me on Saturday!" when I know that it hasn't been released. (Note to Microprose: there are lots of folks who say they have the C64 version of **Gunship**. You didn't even release it till 9-25.) If I had those dimes, I would be phoning this column in from the Virgin Islands. So, Trip Hawkins (Where is Marble Madness for the ST?) and everyone else who keeps using the lame excuse for not releasing Atari software, come spend a Friday or Saturday with me. I'll show you Commodore owners who didn't buy their copy of your game, and Atari owners who are more than willing to spend money on your products."

One problem is that some software developers are producing "junk" and blaming us and piracy for their lack of sales. Maybe the term "junk" is a little strong, but many designers are overly enthusiastic about the value of the programs they develop. If I purchase a program which I cannot use either because it does not meet my needs or because it is too unreliable, that program I call "junk." Very expensive "junk, but "junk" nonetheless!

For the ST we are seeing some real quality public domain software being produced and shared, while at the same time seeing commercial products in the $30-$50 range that offer the same things as the public domain software. If one person can write a program and give it away, I find it difficult to understand how the commercial developer of a similar program expects to sell thousands of copies. Will he assume that piracy is the reason for lack of sales?

If software developers give us truly useful software, most people who need it will pay for it. This name calling is not going to help.

SOFTWARE PIRACY -- ANOTHER VIEWPOINT

Bob Haynes, Atari Computer Club Encompassing Suburban Sacramento

Reprinted from The ACCESS Key, October 1986

Introduction

Software piracy is a vital issue for all of us to consider at this time because:

a) articles in ANALOG, Antic, and Compute! have appeared on the subject the past couple of months;

b) a two-hour symposium on the subject -- the infamous June 21 CompuServe Conference -- was put on line for all to see;

c) a number of our fellow Atari newsletters have bought into the discussion, bowing and scraping to the software pros like so many moist-palmed Uriah Heeps;

d) much of the material is a forum for the software pros to bash Atari users -- hard;

e) the ingredients of these discussions are typically about one-fifth filberts, and three-fifths pure fudge.

These articles tend to focus on only one aspect of the subject; namely, that pirates are thieves who rob the honest, hardworking software pro of an already meager livelihood. [The picture conjured up by such statements -- the haggard mother in her shapeless print dress; the father putting cardboard in his worn, scuffed shoes to make them last a few days longer; the children pleading with eyes sunken in the dark hollows of their gaunt faces for a second helping of thin gruel with reconstituted skim milk -- is so hardtravelling tinlizzie grapesofwrath Oklahoma dustbowl that we evil Atari pirates must take care not to laugh long and loud and rudely in the faces of the software pros who speak of such things in good earnestness.] The plain truth is that the subject is complex and multifaceted, too full of ramifications to be dismissed in a page or a few short paragraphs. Indeed, as sprawling as this article is, it leaves much of the ground uncovered.

Typical Comments

"This [software] rental business is a rather thinly disguised excuse for copying. . . . People are trying to legitimize piracy of our programs under the guise of saying, 'Let's take it home and take a look.'" -- Smith McKeithen, Activision V.P.

". . . At least half of my potential income (and that of most other program authors) is lost to . . . software piracy. . . . Many people . . . rob me and my fellow programmers of the already miniscule royalties we receive for the (average) three to six months of work that goes into each and every program that we sell." -- John DeMar, Quantum Microsystems.

"Estimates for 1985 losses [due to piracy] are in the $800 million range. . . . According to industry observers, piracy is largely restricted to software that runs on personal computers, and the bulk of the loss comes from individuals who make copies as gifts for others rather than from organized counterfeiters who operate their thievery for profit. . . . Since it is the good software that gets copied, it's the good, innovative developers being driven out of the business." -- David Thornburg, Compute! staff writer.

"Atarians are some of the most sophisticated computer users anywhere. . . . They are the biggest pirates in the world. [Emphasis added -- editor] . . . I know I get about one-fourth the units [Atari sales] I used to get from a new release." -- Bill Stealy, Microprose CEO.

"Sales figures don't lie, folks." -- Matthew Ratcliff, ANALOG staff writer, commenting on the above Stealy statement.

Towards a Definition

Although it does not suit the purposes of most software pros to define the phrase software piracy, define it we shall. To define a term is to put limits on it.To mark a boundary around it. To make it definite, rather than infinite. So, here goes:

Software piracy: a semi-abstract noun denoting (1) the act of making an unauthorized copy of a commercial computer program and (2) giving or selling that copy to another party, (3) who uses that program a significant amount (lots and lots), (4) causing the software developer(s) to incur a virtual loss of sales and income.

Note that this definition makes some significant exclusions. Making backup copies of one's own purchases is all right, as most SD's seem to agree. Giving or selling a purchased program to another party is legal if it is the

originally purchased item and if no copies are retained by the original buyer. If the recipient of a copy never uses it, as often happens for various reasons, one would be hard put to assert that the SD's had lost a sale thereby. (Most SD's who address this proposition in print disagree vehemently. It appears that they favor consumers buying software for which they have no use.)

Is Piracy That Big a Deal?

David Thornburg claims that software companies lost $800 million to piracy in 1985. John DeMar puts software piracy at 50%. Bill Stealy's comments imply that the piracy rate for String One Atarians is about 75%. Common courtesy dictates that we not spit ;in the eyes of these misguided gentlemen if we should encounter them at a computer faire or cocktail part. Rather, we should help these lost souls grope toward the truth.

Start with Thornburg's $800 million. Assume that the average retail price of a disk is $50. That's 16 million pirated disks. Unless, of course, we are speaking of the software company's share of the pie, twenty-five simoleons per disk. This would make the number of pirated disks 32 million. Mr. Thornburg is not crystal clear as to which point of view his figures embrace. If we estimate the number of annual home software sales at about 60 million pieces -- which may be as accurate as any figure -- DeMar opines that each of them gets pirated at least once. Stealy might project the number at over 160 million, five to ten times the amount that Thornburg's mysterious "industry observers" cite. So, here we have people who supposedly are on the spearpoint of computer technology, whose stock-in-trade is supposedly more data, faster and truer. SLOPPY work, guys! Get your acts together.

Let's try out some other figures. The latest solid number on U.S. households with computers is around 12 million. Allow that one million of these might own disk backup systems, although that figure seems high. Allow further that each owner of a backup system BUYS five disks a year and makes a copy of each disk to give to a friend. That's five million pirated copies. A little under eight percent of the total commercial output, assuming sales of 60 million. Figure conservatively that a full half of these so-called pirated disks will never be used. Now we are down under 4% true piracy. And <u>that</u> is probably the high end of an honest reckoning. If all piracy were eliminated today, it is unlikely that software sales would grow by even as much as three percent tomorrow.

And where do <u>my</u> lowball figures come from? The same place the software pros get their highend, bluesky numbers. The difference is, having nother to gain by telling statistical lies, I am more likely to be near the truth.

Why All the Fuss Over Piracy?

People in the software industry are walking near the edge of a cliff. And they know it. And it scares them. The days of 40 and 50 percent annual growth, when any fool could make a fortune in computers, are over. Current computer owners have about all the software they can use. There are too many companies trying to sell too many software titles to too few customers. The problem facing the SD's is very real. Their response to it is pure Fairy Tale Town. Wishful thinking has created a HUGE potential market, if only piracy would go away -- or so they would like to have everyone believe. Alas and alackaday! Piracy is a fact of life that will no more go away than house flies and horse flies and tsetse flies. The only things that have a chance of disappearing are the SD's bloated image of software piracy and a pretty big chunk of the software development houses. The ones that survive will be the ones that face reality and learn to work smart.

How To Work Smart

About three years ago, TV audiences were treated to a commercial depicting some nutty guy bursting into a crowded auditorium and hurling a sledge hammer at a giant video screen. It had nothing to do with computers, but it helped the Apple Corporation to prosper mightily. A year later, Apple tried another equally bizarre ad, this one depicting an endless line of business-suited corporate types marching over a cliff. This ad came much closer than the first to depicting reality. But it was a flop, which lends support to the notion that people prefer a cute fiction to a drab reality. Inevitably this second ad reminded people of the small Scandinavian mammals that respond to overpopulation pressures by taking a brisk jog to the coast, followed by mass communal bathing in the ocean. Of course, we humans would never be guilty of such mass hysteria, would we? We can. We have. We continue to be so. The software industry is, as indicated

above, overcrowded. Ripe for a wholesale panic dash over the cliff to oblivion. The classic irrational response rules the SD's -- trying to rearrange the deck chairs on the Titanic just after it hits the iceberg.

In a nutshell, here are some key rules for working smart, for staying afloat while all about you are taking a financial bath. Keep in mind that any one of these rules could be explored in an article of its own. But this is not Business Week Magazine.

a) Stop wasting time, energy, and money on the piracy issue. Piracy won't go away. And it causes only minor irritation and damage. Besides, **calling one's potential customer base a bunch of thieves is not generally recommended as a marketing strategy.** [Empahsis added -- editor]

b) Trim corporate fat. Start at the top and work down. Attack perks. Get rid of deadwood in the organization.

c) Move to a lower rent location. Taiwan and Korea have much to offer in this regard.

d) Since only one U.S. family in eight has a computer, find out what it would take to get the other 7/8 to buy one. Design software to meet their desires.

e) Study seminal "hit" programs like **VisiCalc, Print Shop,** and **Pinball Construction Set.** Understand the essence of these -- namely that they allow users to customize their own stuff easily on the computer.

Atari and the Pirates

As a postscript to this exploration of piracy, here are some comments on the assertion that Atarians are the scurviest of bucaneers to sail the seas of computerdom. All of us "sophisticated" Atarians know otherwise.

a) None of the BIG THREE home computers (Commodore/IBM/Apple) affords such ready protection against the casual copier as does Atari.

b) The amount of pirated software that makes its way home from business offices (dominated by IBM PC's) and from schools (dominated by Apple) makes Atarians' efforts in the field of freebooting seem feeble.

c) Threatening to pull out of the Atari String One market because of piracy is not the talk of rational beings. Any company that cannot port a program from Apple to Atari for $30,000, market it for $30, and make the 2,000 sales needed to break even needs to trim some corporate fat and/or take another look at the quality of the products it is bringing to market.

d) If String One software for the New Atari is not selling briskly, don't blame piracy. SD's need to get their Atari stuff into more stores; cultivate sources like mail order houses, where Atarians were forced to go when they were abandoned for two years by the Industry; understand that they are competing with some pretty good public domain stuff now.

e) To paraphrase a current TV jingle for a grocery store chain, "We work an honest day, and we want an honest deal."

GREAT COVER-UPS

ATARI

Protect your investment with opaque vinyl covers.

8 BIT COVERS

65XE/130XE/800XL..... 4.95
1050 SINGLE DISK DRIVE..... 2.95
1050 DUAL STACKED DISK DRIVE..... 3.95
1027 LETTER QUALITY PRINTER..... 3.95

ST COVERS

520ST/1040ST..... 5.95
MONO OR COLOR MONITOR..... 6.95
3 1/2" DISK DRIVE..... 1.95
3 1/2" DISK DRIVE - DUAL STACKED..... 2.95

PRINTER COVERS

CHOOSE FROM PANASONIC, OKIDATA, EPSON, TOSHIBA, SILVER REED, CANON, FUJITSU, STAR, LEGEND OR CITIZEN..... 4.95

10% DISCOUNT TO ATARI CLUB MEMBERS

GREAT COVER-UPS
6805 SW 8th Avenue
Portland, Oregon 97219

Phone: (503) 246-8977
9am - 1pm weekdays

SO YOU WANT TO SELL SOFTWARE!

Rich Anderson, Vice President, BAAUG
Reprinted from the Bay Area Atari Users Group Newsletter, July 1986

Ok! You got this really hot idea for a new software program. You just know everyone will want to buy a copy -- perhaps two copies. You slave over your keyboard for untold hours and de-bug and de-bug some more. Satisfied, you turn to your notes and write the world's finest documentation. You're ready for the world! But, is the world ready for you?

Just how does one actually market that masterpiece? How do you actually make a buck out of all your hard labors? That will be the subject of this column this month and in the months to come.

I expect most of us are aware of the five (or if there are more please write and let me know) ways you can get paid for writing software. First, you write a program for one person's or firm's exclusive use. A friend of mine, also in the insurance business, has paid a fellow who has a computer software business well over $10,000 to develop a program which will allow him to quickly quote and compare group insurance policies. This program has really fantastic features which adhere to the standards and goals my friend set. Interestingly, the program has a few bugs and rough edges but at the $10,000 mark my friend said "Stop!", and closed the purse.

A second way to make money with your creative labors is to market the program through an established publisher of programs like OSS, Broderbund, Peachtree, Atari, etc. This way you have the opportunity to reach the most people as the publisher will take responsibility for advertisement, packaging, and literally everything. You, naturally, will make only a small percentage of the sale price but the marketing worries won't be yours.

A third election would be to sell your program to a magazine like Antic, ANALOG Computing, or Compute!. These publications will pay you but you may retain the right to publish the software commercially later.

A fourth method is selling directly to consumers by advertising in computer publications and dealing by direct mail. You also could sell directly to retailers. This would reduce your profit on each item but would expand your market. In either case, marketing is your headache.

Finally, there is the "freeware" method. Nothing could be much simpler than this method. Make a few copies, give the copies away and encourage duplication and on every disk leave the message that you would welcome contributions. The author of **PC Write** and **PC File** (to name only two programs) for the IBM and compatible MS-DOS machines) have make a TON of money this way. Really not sure if anyone has made much money this way with Atari programs, but maybe you?

In a matter of days (perhaps as you read this) our BAAUG President, Frank Nagle, will have his utilities being commercially sold in retail stores. As I write this, Mark Blumenkamp (our SYSOP) and myself are working on a Genealogy program which we intend to direct market. In the months ahead I will report to you on our experiences -- good and bad. Perhaps our experiences will be educational to you if you are thinking of bringing one of your own ideas to fruition. If you have personal experience in this area we welcome your input. (I'm sure Bill Wilkinson could tell us about starting OSS, but I'm thinking on a bit smaller scale than that.)

****

PAC Help Hotlines

The following people have generously offered to take telephone queries in the areas indicated.

Adventure Games	Jim Miller	641-6356
	Zant Burdine	206-695-5604
Assembly Language	Leroy Baxter	653-1633
BASIC Programming	Nick Yost	981-0838
	Lee Gassaway	642-2455
BBS Usage	Steve Billings	246-1751
	Don Adams	245-7168
	Russell Schwartz	646-6418
C	Randal Schwartz	626-6907
Cassette Operation	Lee Gassaway	642-2455
DOS Operation	Wayne Winterbottom	669-1367
FORTH Programming	Ron Chaffer	283-5691
	Ricky Wooldridge	224-7163
Operating System	Nick Yost	981-0838
	Leroy Baxter	653-1633
Pascal	R. DeLoy Graham	649-6993
ST General	Chuck Hall	626-3717
ST Fundamentals	Richard Barhitte	206-573-0292
ST Logo	Randal Schwartz	626-6907

BATTLE OF THE TITANS
Sargon III vs. Chessmaster 2000 vs. Colossus 3.0 vs. Two Fidelity Chess Calculators!

Bill Carpenter and Jim Holman, PAC

The release of **Chessmaster 2000** with the claim that it is better than **Sargon III** motivated us to hold a little computer chess tournament to test this assertion. Also, we were interested in seeing in general the quality of some popular chess programs for the 8-bit Atari, and in seeing how these programs compare with some of the earlier chess calculators. We must make clear from the start that we do not claim to be experts at chess. We are not gifted amateurs, nor even amateurs. We are just a couple of simple woodpushers who enjoy a good game of chess.

The contestants in our tournament were **Chessmaster 2000, Sargon III, Colossus 3.0,** Fidelity Electronics **Chess Challenger** (about 10 years old - referred to here as "Old Fidelity"), and **Mini Sensory Chess Challenger** also from Fidelity (about 5 years old).

We realized very early that to test fairly the capabilities of each program would require a large number of games between each program, played at different time levels, with each program alternating between the black and the white sides. Given the number of programs and that we did not want to spend the better part of the remainder of our adult lives in this endeavor, we decided against fairness, and settled for a simpler tournament format. Before describing this format, a few introductory words are in order.

We felt strongly that most chess players would be most interested in seeing how the programs performed against each other in games with reasonable playing times. Therefore, we did not select time levels of greater than one minute per move. Even at this level, a game would often take nearly two hours. This is also why we selected progressively shorter times for the playoffs between the two best programs. Yet we were afraid that in so doing we might unfairly handicap a program which might be a wimp at one minute per move and a lion at two minutes per move. But in our tests, we found that the programs would usually find the best moves they could find within one minute, although at longer times they sometimes would choose other continuations. At very long playing times (8 to 10 minutes per move) they often did not find lines of play different from those found at two minutes per move. We suspect that this is because the move selected is a function both of time selected and whatever move selection criteria are built into the program. It seems that the programs fairly quickly find the "best" move, and that any additional time usually just allows for a deeper analysis of the position.

The format of the tournament was as follows: One minute per move, two games between each program with each playing black or white alternately, 30 seconds per move in the playoff round, 15 seconds in the second playoff round, if necessary. (It was necessary.)

The results of the tournament are as follows (winner is underlined - no underline for a draw):

ROUND 1 - one minute per move

WHITE	BLACK
Mini Sensory	Old Fidelity
Old Fidelity	Mini Sensory
Mini Sensory	Chessmaster
Chessmaster	Mini Sensory
Mini Sensory	Sargon III
Sargon III	Mini Sensory
Mini Sensory	Colossus 3.0
Colossus 3.0	Mini Sensory
Old Fidelity	Chessmaster
Chessmaster	Old Fidelity
Old Fidelity	Sargon III
Sargon III	Old Fidelity
Old Fidelity	Colossus 3.0
Colossus 3.0	Old Fidelity
Chessmaster	Colossus 3.0
Colossus 3.0	Chessmaster
Chessmaster	Sargon III
Sargon III	Chessmaster
Colossus 3.0	Sargon III
Sargon III	Colossus 3.0

Assigning one point for a win, one-half point for a draw, and no points for a loss, the scores at the end of round one were:

Mini Sensory	0
Old Fidelity	2
Chessmaster	5
Sargon III	6.5
Colossus 3.0	6.5

Round 2 Playoff - 30 seconds per move

WHITE	BLACK
Colossus 3.0	Sargon III
Sargon III	Colossus 3.0

Score after round 2:

Colossus 3.0	7.5
Sargon III	7.5

Round 3 Playoff - 15 seconds per move

WHITE	BLACK
Colossus 3.0	Sargon III
Sargon III	Colossus 3.0

Score after round 3:

Colossus 3.0	8	
Sargon III	9	WINNER

Discussion

The programs played predominantly Kings Pawn openings, although there were some Queens Pawn openings, and a few odd openings. Some comments on the style of play by the different programs are in order. We should first note that all three Atari chess programs play very good chess. But there are some differences. **Chessmaster 2000** seemed weaker than **Sargon** or **Colossus**. Out of four games with the other programs, **Chessmaster** won only one. Frankly, we found **Chessmaster**'s claim of superiority to be unsupported. In addition to losing more than the other programs, it seemed generally to play a weaker game of chess. Especially when in a difficult or losing position, **Chessmaster** made moves which seemed to lack strategic purpose. Its play in these situations might be described as "confused".

Colossus played a very tight game, but often was not a very exciting opponent. **Sargon**'s game did not seem as tight, but it often made moves which were quite surprising and exciting. Of the three, **Sargon** seemed to play the "prettiest" game. The difference between the style of play is that with **Colossus** you feel that you are playing against a machine; with **Sargon** you feel more like you are playing against a human opponent. Overall, even though **Sargon** was technically the winner, we judged **Sargon** and **Colossus** to be about equal in strength, as the tournament results show.

The Fidelity chess calculators were clearly outmatched, especially poor **Mini Sensory**, beaten by everyone including "Old Fidelity," a competitor five years his senior. (**Mini Sensory** wins the "Egregious Blunder Award" for castling his king into a position which allowed him to be mated the very next move.) Even though the Fidelity calculators lost to the Atari programs, they are fun to play, and both are good enough to give the average woodpusher a run for his or her money.

All the Atari programs have some very nice features. All three have books of opening moves. **Sargon III** has a book of 68,000 moves, although the book is on a separate disk. **Chessmaster** has a book of 71,000 moves, and **Colossus** a book of about 3,000 moves. The larger books did not prove to be as much of a competitive advantage as might be thought. This is because there is no guarantee that the program will play the strongest continuation within each type of opening. This is actually a nice feature, because you would want an opponent who would give you a chance to play against a lot of different openings, not just against a handful of the strongest. All the programs let you save games to disk, and **Sargon** and **Chessmaster** will print a hardcopy of the game. Both **Sargon** and **Chessmaster** come with collections of famous chess games on disk. All the programs allow you to see the moves they are considering, but only **Sargon** and **Colossus** display the full tree of moves considered to be the most likely line of play. **Chessmaster** displays only the move under consideration and the opponent's most likely next move. We felt this to be an unfortunate limitation. **Sargon III** and **Colossus** also display a rating of their positions, based on material and positional strengths and weaknesses. **Chessmaster** does not have this feature. All the programs have other features which some users may like, but which we will not discuss here.

In summary, we feel that any of the Atari programs is worth having. But we must again question the claim that **Chessmaster 2000** is the best. We could not detect this in either the quality of play or the program features. We feel that **Chessmaster 2000** really has nothing significant to commend it over either **Sargon III** or **Colossus. Sargon III** is the nicest program we saw, but **Colossus 3.0** runs a very close second. And since **Colossus** sells for less than half as much as **Sargon III** (or **Chessmaster 2000**), we believe **Colossus 3.0** is the best buy among the programs we tested.

Special thanks to those who lent us copies of **Sargon III** and **Chessmaster 2000**, and for the loan of an Atari 130XE.

PASCAL
Lesson 3: Program Construction

R. DeLoy Graham, PAC

In previous lessons, we learned about the special symbols used in Pascal and about Pascal identifiers. In this lesson we will examine program construction.

A Pascal program must be constructed according to certain rules in order for the compiler to translate it into machine language (or, in some cases, p-code). These rules might vary slightly from compiler to compiler, but we will talk in terms of the standard definition of the language.

The accompanying figure depicts graphically the block structure of a simple Pascal program. Procedures (and functions) are optional, but will greatly improve the modularity of the code.

As the diagram shows, we begin a Pascal program with the word PROGRAM followed by an identifier indicating the name of the program. The word PROGRAM is a reserved word which tells the compiler that we are starting a program. Because we will write many programs, each program must have a unique identifier or name so that we can access it. It will be helpful later if we choose an identifier whose meaning indicates the purpose of the program. The program header or block delimiter ends with a semicolon.

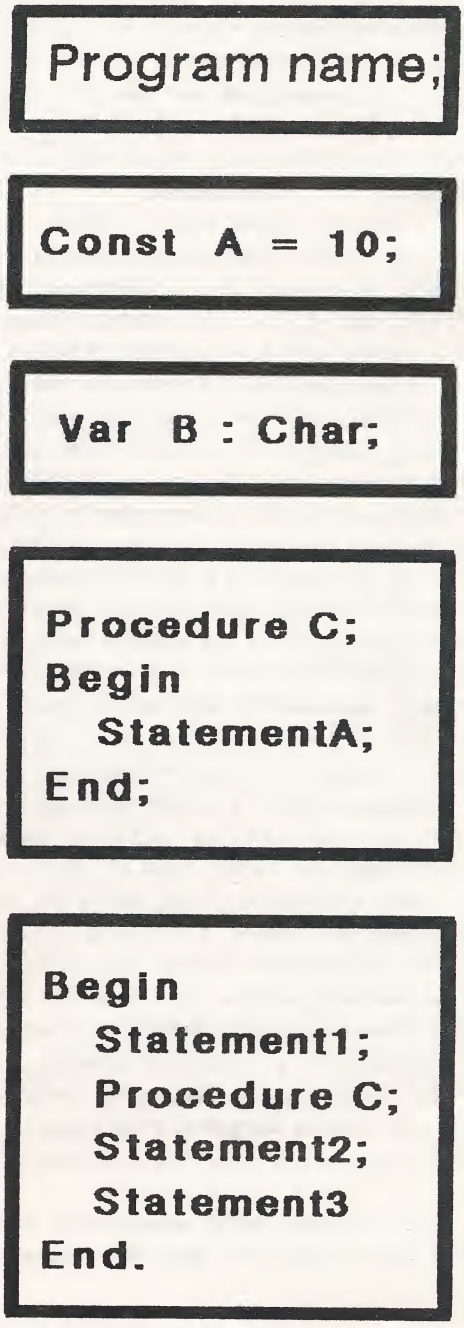
Next we can define constants that we want to use later in the program. In Pascal, all constants must be defined and all variables declared before they are used. To start the constant definition block, we use the reserved word CONST. Each definition ends with a semicolon. The value assigned to the constant also declares its type.

Our next block is used to declare variables that we plan to use. Notice that what we are doing is telling the compiler what the type the variable is, so that it knows how much storage space to reserve for the variable.

Next come any procedures and functions that we will call from the main program block, which is the last block of the program. A procedure is like very much like the program itself, with its own header, constant and variable decalarations, and statements. It is, however, a subprogram or subroutine that will be executed only when called from another part of the program.

The main block contains the controlling statements of the program. A statement is an instruction, or a call to a procedure or function that contains instructions (other statements) for the computer to execute or carry out. The main block ends with the reserved word END followed by a period. That period is the only period in the program and tells the compiler that it has reached the end of the source code to be compiled.

Next month we'll learn about procedures and functions. You might want to look at the program accompanying my review of **Kyan Pascal**. It is a little advanced for the beginner to Pascal, but a careful reading will show you how a Pascal program is put together. Which parts to code as procedures or functions is a decision the programmer must make. His goals will include efficiency (time and space), but should also take into account readability.



A REVIEW OF KYAN PASCAL

(from Kyan Software Inc. -- $69.95)

R. DeLoy Graham, PAC

Last year, I had a student in my Advanced Placement Computer Science course who had an Atari 800XL and wanted me to recommend a Pascal compiler that he could use to write programs for our class. The only Pascal I knew of for the Atari at that time was **Draper Pascal**, which was unacceptable for use in the course because of its lack of TYPE definitions. Discouraged, he sold his Atari and purchased a Commodore. I was unhappy to see him make that mistake, but what could I say? I have long felt that one of the reasons 8-bit Atari computers have made such a poor showing in schools was the lack of a good Pascal compiler, since Pascal is required for the AP course and so popular in colleges and universities.

Although **Kyan Pascal** was out at the time my student was looking, I had not seen it, nor was it being sold locally. Our friends at IB Computers have since ordered some copies, which, I must add, have been selling quickly. Eight-bit Atari owners now have a really nice Pascal compiler available.

Kyan Pascal offers much more than just a compiler. Included is **KIX**, Kyan's own command line processor, essentially replacing the need to load DOS 2.5; it reminds me a little of **DOS XL** from OSS, although the commands are more CP/M-like. Commands include **LS** (list directory), **CAT** (show contents of a file; copy many files into a single file), **CP** (copy files), **MV** (rename files), **RM** (delete files), **FORMAT**, **CHMOD** (lock and unlock files), **SD** (screen dump), **CD** (change drive). The commands are easy to learn, but if you prefer there is a menu that can be loaded on your command.

Kyan Pascal, which conforms to the ISO standard, is also available on the Apple II line of computers. It is a substantial product. The documentation is among the best I have seen for any computer software, regardless of computer type. It comes in a 6 X 8" three-ring binder with a fairly good index and a 167-page tutorial. Although I did not work through the entire tutorial, which is designed to teach the fundamentals of Pascal, what I did use impressed me. The authors have included many examples illustrating the various concepts presented. I believe you will still want to supplement this instruction with a standard Pascal text, but this is a great start and will introduce you to Kyan's particular implementation.

For those of you who enjoy writing assembly routines which you run from your BASIC programs, there is a facility included which allows you to write assembly code in your Pascal programs. To begin a block of assembly code, the programmer types **#a** in the first and second columns. The block is concluded with a single **#**. What makes this possible is that Kyan's compiler creates assembly code which is then assembled into machine code. Although the assembler included is not as powerful as Kyan's **Macro Assembler and Linker**, it is quite sufficient for most needs. It even has a macro facility.

The programming process begins by calling up Kyan's text editor and entering the program source code. The editor is very nice! Besides the standard Atari full-screen editing, this editor allows you to scroll up and down through the entire code. Several control commands allow quick movement through the text. You are able to delete or move text and to search and replace a string of characters in the text. Other files can be merged with the one you are working on.

After writing the source code, the programmer calls up the compiler, which translates the source code to assembly code. If an error occurs during compilation, the process is aborted and an error message is given which indicates the line at which the error occurred. Once back in the editor, the programmer can use the GOTO command to jump directly to the line in question.

The final step in the process is to run the assembler to create a machine language program. To run the program requires a runtime library. If that is a real problem to you, Kyan will sell you the source code for the runtime library so that you can append portions of the library directly to you program.

To review this product, I decided I would write a program and find out just how well it all worked. At first I used my Atari 800 with one 810 disk drive, but I became somewhat impatient waiting for the various parts of the program to load (I spend most of my programming time on a 520ST). Vern Vertrees was nice enough to loan me one of his 130XEs. Since the Kyan environment makes use of the 130's ramdisk capability, everything was greatly speeded up.

Besides the Pascal package, I also used Kyan's **Toolkit #1** which is a library of over 90 assembly language "include" files which provide

system utilities unique to the Atari that are not part of the ISO standard. These include Input/Output utilities such as **Stick**, **Ptrig**, **Strig**, **Fill**, and several file operators such as **Delete** and **Lock**; system functions such as **Disable_Break**, **Enable_Cursor**, **Disable_Attract**, **Freeze_On_Reset** and **Disable_Key_Click**; and screen management utilities such as **Clear_Screen**, **Clear_To_EOLn**, and **GotoXY**. There are just too many to name. This toolkit sells for $39.95, but it is worth the price if you are interested in getting things done. To use an include file in your program, you simply declare it in your source code (as I have done in my example program) and then call the routines as if they were functions or procedures in your program.

The program I wrote is a simple flashcard program. I used it to practice recalling the meanings of some Spanish verbs. The program draws a box on the screen and writes either the Spanish or the English verb and then waits for the user to enter the corresponding word in another box. In a third box, the user is told whether his response was correct or not.

I wrote the box-drawing procedure as a separate include file to illustrate how you can create segments of code that might be useful in many programs and start your own library. Of course, these would be more efficient if coded in assembly language, but this is a Pascal review, remember? The other include files are are part of Kyan's **Toolkit #1**. I used the following commands from these libraries: **Disable_Cursor**, **GotoXY**, **Random**, and **Get_Char**. The **GotoXY** command can be implementing using the **Position** command included on the Kyan Pascal disk. The others would have to be implemented by you if you do not have **Toolkit #1**.

Please note that I used only infinitives for this program so that I would not have to deal with special punctuation marks found in Spanish. An enhancement would be to create a special character set containing these marks. My **Personal Pascal** version on the ST will use the special Spanish characters already available.

Further note that **Kyan Pascal** looks very standard. It has been fun working with it. Most definitely having a ramdisk and two drives improves the overall programming environment, but it is quite workable with a 400/800 and one drive.

If you are interested in programming in Pascal, get this program. It is well worth the money (in fact, worth much more). This company deserves our support. Their products are not copy-protected for user convenience.

Figure 1
PASBOX.I

```
PROCEDURE Box (Row, Col, Row2, Col2 : Integer);

VAR  Width : Integer;

(*********************************************)

  PROCEDURE Vert (Width: Integer);
  VAR I : Integer;
  BEGIN
     FOR I := 1 TO Width DO
       WRITE ( CHR(18) );
  END;  (* Vert *)
(*********************************************)

  PROCEDURE Top (Width: Integer);
  BEGIN
     WRITE ( CHR(17) );
     Vert (Width);
     WRITELN ( CHR(5) );
  END;  (* Top *)
(*********************************************)

  PROCEDURE Sides (Row, Col, Row2, Col2
                                    : Integer);
  VAR I : Integer;
  BEGIN
     FOR I := Row+1 TO Row2-1 DO
         BEGIN
            GotoXY (Col, I);
            WRITE ( CHR(124) );
            GotoXY (Col2, I);
            WRITE ( CHR(124) );
         END;
  END;  (* Sides *)
(*********************************************)

  PROCEDURE Bottom (Width: Integer);
  BEGIN
     WRITE ( CHR(26) );
     Vert (Width);
     WRITELN ( CHR(3) );
  END;  (* Bottom *)
(*********************************************)

BEGIN
   GotoXY (Col, Row);
   Width := Col2-Col-1;
   Top (Width);
   Sides (Row,Col,Row2,Col2);
   GotoXY (Col, Row2);
   Bottom (Width);
END;  (* PROCEDURE Box *)
```

PROCEDURE L **Figure 2**
FLASH.P

```
PROGRAM FLASHCARD (INPUT, OUTPUT);

(* Written by  R. DeLoy Graham *)
(* for the Portland Atari Club *)
(*       December 1986         *)
(*   Written in  Kyan Pascal   *)
(*     on an Atari 130XE       *)

CONST
  FILE1 = 'SPAN.DAT';

  Size  = 25; (* number of flashcards *)
  StrSize = 15;
  Maxstring = 38;
  Blank25 = '                         ';
  Blank38 =
  '                                     ';
  R1 = 6;     R2 = 8;  (* row and     *)
  R3 = 12;    R4 = 14; (* column      *)
  R5 = 18;    R6 = 20; (* coordinates *)
  C1 = 6;     C2 = 32; (* for boxes   *)
  C3 = 0;     C4 = 38;

TYPE
    String38 = Array[1..Maxstring]
               of Char;
    String = Array[1..StrSize] of Char;
    Card = Record
             One,
             Two  : String;
             Check : Boolean
           End;  (* record *)
    Cards = Array[1..Size] of Card;

VAR Score,
    Row, Col, Row2, Col2 : Integer;
    Ans : Char;
    Words : Cards;

#i D1:Gotos.I   (* include files  *)
#i D1:Cursor.I  (* from Toolkit 1 *)
#i D1:Randoms.I
#i D1:GetChar.I

#i D1:PASBOX.I  (* my box routine *)

(*********************************************)
```

```
PROCEDURE LoadData(Var Words : Cards;
                   Var Score : Integer);

VAR I : Integer;
    First,
    Second : String;
    f : Text;

BEGIN
   Reset (f,FILE1);   (* Open file *)
   For I := 1 To Size Do
     BEGIN
       Readln (f, First);
       Readln (f, Second);
         (* Put words in array *)
       WITH Words[I] DO
         BEGIN
           One := First;
           Two := Second;
         END;
     END; (* For *)
   Score := 0;   (* Init score *)
END;  (* LoadDAta *)

(*********************************************)

PROCEDURE SetUpScreen;

VAR Row, Row2,
    Col, Col2  : Integer;

BEGIN

   Disable_Cursor;

     (* Clear screen *)
   Write (Chr(125));

     (* Draw question box *)
   Box (R1, C1, R2, C2);

     (* Draw answer box *)
   Box (R3, C1, R4, C2);

     (* Draw response box *)
   Box (R5, C3, R6, C4);

END;  (* SetUpScreen *)

(*********************************************)
```

```
PROCEDURE RunTest (Var Words: Cards;
                   Var Score: Integer);
VAR I,
    Status,
    Which,
    Index : Integer;
    Reply : String;
    Ch : Char;

 (*****************************************)

  PROCEDURE CheckAnswer
             (Ans1, Ans2 : String;
              Which : Integer;
              Reply : String;
              Var Score : Integer);

  BEGIN

    IF Which = 2
       THEN
         IF Reply = Ans1
            THEN
              BEGIN
                Write ('Bueno.');
                Score := Score + 1;
              END
            ELSE
              BEGIN
                Write ('No, se dice ');
                Write (Ans1);
              END
       ELSE
         IF Reply = Ans2
            THEN
              BEGIN
                Write ('Bueno.');
                Score := Score + 1;
              END
             ELSE
              BEGIN
                Write ('No, se dice ');
                Write (Ans2);
              END

  END;  (* CheckAnswer *)

(*********************************************)
```

```
  PROCEDURE Randomize (Var Words : Cards;
                       Var Which, Index : Integer);

  BEGIN
      Which := Random (1, 2);
      Index := Random (1, Size);
      While Words[Index].Check Do
        Index := Random (1, Size);
        (* Set check true so not used again *)
      Words[Index].Check := TRUE;
  END;  (* RND *)

  (*****************************************)

BEGIN  (* RunTest *)
    (* Initialize Check Flags *)
  For I := 1 To Size Do
    Words[I].Check := FALSE;

  For I := 1 To Size Do
    BEGIN
        (* Get Random Num *)
      Randomize (Words, Which, Index);
        (* Clear question box *)
      GotoXY (C1+1, R1+1);
      Write (Blank);
        (* Print question *)
      GotoXY (C1+1, R1+1);
      If Which = 1
         Then
           Write (Words[Index].One)
         Else
           Write (Words[Index].Two);
        (* Clear answer box *)
      GotoXY (C1+1, R3+1);
      Write (Blank);
        (* Get answer *)
      GotoXY (C1+1, R3+1);
      Readln (Reply);
        (* Clear response box *)
      GotoXY (C3+1,R5+1);
      Write  (Blank38);
      GotoXY (C3+1,R5+1);
        (* Check Answer *)
      CheckAnswer (Words[Index].One,
                   Words[Index].Two,
                   Which,Reply,Score);
        (* Wait for user to read response *)
      Status := Get_Char(Ch);
        (* Clear response box *)
      GotoXY (C3+1, R5+1);
      Write (Blank38);
    END;
END;  (* RunTest *)

(*********************************************)
```

```
PROCEDURE ReportResults (Score : Integer);

Var Ch : Char;
    Status : Integer;

BEGIN
     (* Clear question box *)
   GotoXY (C1+1, R1+1);
   Write (Blank);

     (* Clear answer box *)
   GotoXY (C1+1, R3+1);
   Write (Blank);

     (* Clear response box *)
   GotoXY (C3+1, R5+1);
   Write (Blank38);

     (* Print results *)
   GotoXY (C3+1, R5+1);
   Write ('You answered ',Score,
          ' correct for ',
          Trunc(Score/Size*100),
          '%.');

     (* Wait for user to read respons *)
   Status := Get_Char(Ch);
   GotoXY (C3+1, R5+1);

   Enable_Cursor;
     (* Clear screen *)
   Write (Chr(125));
END;  (* ReportResults *)

(*********************************************)

BEGIN  (* Main Program *)
   LoadData (Words, Score);
   SetUpScreen;
   RunTest (Words, Score);
   ReportResults (Score);
END.   (* Main Program *)
```

Figure 3
SPAN.DAT

```
andar
to walk
hablar
to speak
contar
to tell
decir
to say
bailar
to dance
cantar
to sing
comer
to eat
viajar
to travel
llevar
to carry
traer
to bring
llegar
to arrive
salir
to leave
bajar
to step down
subir
to climb up
poner
to put
poder
to be able
tener
to have
ver
to see
ir
to go
dormir
to sleep
oir
to hear
estudiar
to study
escuchar
to listen
jugar
to play
venir
to come
```

A REVIEW OF CRYSTAL

(from **Antic** - $24.95)

Clyde Pritchard

Crystal is a multi-function desk accessory written by Jim Thompson and marketed under the **Antic** label. It is available at local dealers as well as through **Antic**'s "The Catalog", and lists at $24.95. It is distributed on an unprotected disk that also contains a documentation file for you to print.

According to the documentation, the first goal of **Crystal** is to enhance the features of the GEM desktop by adding features for printing and easier manipulation of multiple files to enhance file maintenance. The second and main goal is to provide the features of the GEM desktop from within any GEM application.

Crystal will format disks, display disk directories, copy files, move files, delete files, print files, print disk directories, make folders, delete folders, rename files, display disk free space and display file information. GEM has most of these functions, but not all of them, and sometimes requires that you repeat a function, i.e. copy or delete files for each page of a disk directory. **Crystal** can display up to 150 files per page.

GEM has a print file function, but all it does it dump the file to the printer. **Crystal** allows you to produce a paginated list with headers, footers, file names, and page numbers. You can also set the width of the listing and even use some printer control characters. **Crystal** allows you to select multiple files to print, so you can put it to work and go do something else without having to come back and pick the next file to print. The file print function has become my favorite - it is really slick.

The directory print function lets you print disk directory listings of 1 to 9 columns on paper or labels. The listings can be for all or just selected files on a disk. You can specify a title for the listing, and use printer control codes.

The move file(s) function saves you from having to select files once for copy and once for delete as GEM requires. **Crystal** copies the files from the source disk to the destination disk, then deletes them from the source disk. This means that you can be sure that only the files that you wanted moved are deleted, you don't have to worry about selecting the wrong files for deletion as you do under GEM alone.

Crystal has several options for file selection. You can use the mouse to select files one by one like you do with GEM, select all files via a special icon and then deselect the files you don't want, or you can specify a wild card mask for file selection, i.e. "*.DOC" for all files that have the extension "DOC". Once you have selected the files to be processed, you select the function to be performed, and that's it.

Files can be renamed one by one as under GEM, or you can specify another wild card mask for the rename function, i.e. "*.TXT" to rename all selected files so they have the extension "TXT". The rename mask allows for more complicated renames, even to the character level for each position of the file specification.

Crystal will work with color or monochrome monitors, and has a separate ".RSC" file for each type of monitor. If you have both types of monitors, you can have both ".RSC" files on your disk a the same time. **Crystal** will use the proper file for the type of monitor that is active.

The **Crystal** accessory has about 30K of code, but it allocates about 55K of RAM. This seems OK given its functionality.

Crystal also has an install/customization program that can be used to change the defaults used in the print function dialog boxes to fit your personal preferences and printer needs.

Sounds like a "GEM", huh? Well, **Crystal** does everything it says it will, and it seems to work without error, but unfortunately it does so at a pace even slower than GEM itself. It takes 1:59.79 minutes to format a double sided disk vs 1:42.79 minutes under GEM. A file copy of 15488 bytes takes 17 seconds vs 11 under GEM. I was hoping that **Crystal** would speed up some of these functions, but that wasn't the case.

In spite of its slower operation, I still like **Crystal**, especially the print functions. I don't use it for general copy functions where there is no advantage over GEM, but its special functions can be useful. For people that like the GEM interface vs command processor shells, **Crystal** could become a real favorite.

The documentation file does a fair job of describing how to use **Crystal**, but I'm sure that it could be expanded a bit for beginners. The best way to get used to it is to use it, just like we all have with the standard GEM desktop.

REVIEW OF DEGAS ELITE

(from Batteries Included - $79.95)

Steve Billings, PAC

DEGAS Elite is an upgraded version of the original **DEGAS** for the Atari ST. This includes an upgraded price tag also. The original **DEGAS** was a very popular drawing program and is touted to be the best selling title yet released for the Atari ST. It probably does not, however, have quite as wide a circulation as the Atari drawing program **Neochrome** due to the fact that **Neochrome** was free and included in the ST sales package.

When **DEGAS** first came out I took a look at it and thought it was nice, but kept going back to using **Neochrome** simply because it was so simple to use and choosing colors was so easy. I loved having the **Neochrome** color map available on screen to pick colors and the 'fat bit' area to do more detailed areas and all the tools just one click away. **DEGAS** did not have the color map and enlarging an area to do detail work and picking tools was more involved. You had to jump between two screens. One had the picture and one had the tools and color choices.

DEGAS Elite is a much more complex drawing program than **Neochrome** and a big improvement over the original **DEGAS**. You still have to jump between screens. (Although all functions can be called up while in picture mode with the right keystroke, you would have to have quite a memory.) It is easy to use if you are reasonably careful, and it has some excellent tools. I could find very little that was missing. The manual is like a paperback book and essential to using the program. The great flexibility of **DEGAS Elite** provides many options and it does take some practice to learn all the tools at hand.

DEGAS Elite works in all ST resolutions. It also does some tricks like saving files in a condensed format to save disk space or saving files in non-condensed mode to be fully compatible with the older **DEGAS**. Another feature that made me jump for joy is that it will load **Koala Pad Micro Illustrator** files. Did you read that right? I mean that you can load 8-bit Atari Koala files. I tested it out by downloading a couple 8-bit pic files off the BBS using the ST and, sure enough, it loaded them right up. The Koala pictures looked better than they ever did before. Now if I could just figure out a way to use the Koala pad with the ST!!! The pad is great for tracing small pictures onto the screen. The mouse does not work too well for tracing.

I don't wish to describe all the drawing tools available with **DEGAS Elite**; instead, I will just try to hit some of the ones that impressed me the most.

You can load in multiple screens. Depending on your computer's memory you can load up to 8 pictures, or use 8 work areas. I found it useful to use one area to build my picture and another to test ideas or draw components before adding them to the actual picture I was working on. Be careful though. I found out the hard way that you need to be careful which picture buffer you are saving! I accidentally saved one of the work buffers and deleted the main picture I was working on. I was very disappointed when I reloaded the picture the next day and all I saw was a small portion of the picture I thought I had saved.

There are also multiple powers of magnification. Depending on how much detail you desire, you can control the size of the area and magnification. I found this very useful. Another feature is the skewing, rotating and distortion tools. With them you can resize, rotate, or reshape objects to fit your needs. This comes in real handy if you cut out something from another picture and need to resize or rotate it for your current project.

The palette control has also been improved from the original **DEGAS**. Now there is a map to pick from. It is not as attractive or as easy to use as the **Neochrome** map that I love so well, but it is a definite improvement over the 'guess a color by its numbers' process that had been used previously in original DEGAS. All of the tools like fill, draw, text, shadow, etc. allow a great deal of control over their qualities. The adjustability and flexibility are excellent. There are even auxiliary programs to create and modify fonts and convert pictures of other formats to the **DEGAS** format.

Now for a few things I did not like too well. I already mentioned the problem that I had saving the wrong buffer. I wish there was a caution or alert to indicate which buffer was being saved. When you get in a hurry it is too easy to make dumb mistakes; trust me, I speak from experience. Although they may get in the way sometime, I am all for additional safeguards against human error.

(continued on page 24)

BASIC GROWS UP

John Parchem

(reprinted from **ZMAG**, 18 November 1986)

The demise of BASIC as a serious programming language has been predicted by many computer experts for a long time. Many people felt that as computers "grew-up" they would require more powerful languages (ala Forth) to exploit their new potential and capabilities. Not too long ago David Thornburg of Compute! magazine realized that his prediction of Logo eclipsing BASIC as the primary computer language for beginning programmers was simply a case of trying to "wish" his philosophy into existence. BASIC is alive and well. Most of us have ranted and raved about serious programming languages such as Pascal, Forth, and even Fortran. No one hardly gives second thought to BASIC as a serious programming language. The problem is, at least in part, that we haven't really explored the "evolved" BASIC of today. The giant BASIC's of today bear little resemblance to the BASIC's of yesteryear.

The first major breakthrough for BASIC came with the advance of mass-memory for the new generations of computers. Finally the programmers could develop interpreters that were powerful monsters rather than the tiny 100 or so commands we were once restricted to. Powerful logic statements such as While/Wend, graphic commands, and extremely powerful debugging tools slowly worked their way into today's BASIC. Recently, many good BASIC compilers have appeared on the market. Once such compiler is the Logical Design Works BASIC Compiler. This program is 100% compatible with ST BASIC. It also fully support double precision floating point operations. The real advantage is not that it adds a plethora of new features, but rather in its ability to **compile** ST BASIC programs into equivalent machine language .PRG files. The resulting code is very compact, and blazing fast! Finally someone has done something about ST BASIC. Three cheers for LDW!

So, you see, BASIC is alive and well. It is THE most popular computer language. There is no computer today that I know of that doesn't have a BASIC interpreter available for it. What language can claim that fame? With the appearance of BASIC compilers, programmers will have a user-friendly environment to create professional quality software for commercial sale. Speed, transportability, power, and user-friendliness are all elements of BASIC that assure its longevity as the dominant programming language.

So the next time your ready to hack away at that major program you've been thinking about, or if you just want to experiment with your programming creativity, boot yourself up with BASIC. It's not just for beginners anymore.

****

(**DEGAS ELITE**, continued from page 23)

Another feature I kind of expected was the ability to lasso parts of the picture. This would allow you to draw a line around an area that you wanted to move, following the true outline of the object. **DEGAS** allows you to cut out a block, but it is always rectangular. This means sometimes you will pick up stuff in the box that you might not want. This requires you to clean it up a bit to get rid of things you did not need to pick up.

Another thing that is lacking is the ability to format a disk or check free space on a disk. I have run into the problem of wanting to save a file without knowing how much room I have left on a disk. It would be nice to be able to play it safe and format a new disk to save it on.

I like to use **DEGAS Elite** in conjunction with **CAD 3D** from Antic. It happens to be written by the same author, Tom Hudson, who is doing excellent work for the ST. If your needs are simple stick with **Neochrome**. It is still my second choice and you cannot beat the its price. However, if you need the best available and are about $80 richer than broke, this is the best drawing package I have seen for the ST so far. I would recommend it to anyone who wants or needs software for drawing and graphics.

COMPUTEROLA

239-4315

PRINTERS & RIBBONS

SILVER REED---CITIZEN---STAR---PANASONIC---TOSHIBA---OKIDATA

FOR THE ST

ACTIVION...ARTWORK...BATTERIES INCLUDED...COSMI...ELECTRONIC ARTS

EPYX...HABA...HIPPO...INFOCOM...MICRO PROSE...MINDSCAPE

POLARWARE/PENGUIN...SIERRA ON-LINE...SPINNAKER

STRATEGIC SIMULATIONS...SYNAPSE...TELARIUM ...UNISON...AND MORE

OTHER STUFF

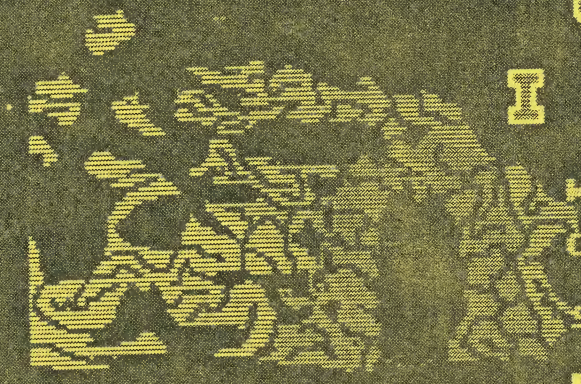
ST Printer Cables	25.00
Universal Printer Stand	25.00
Microbit Interface	50.00
Monitors	CALL
Wishbringer (one only)	15.00
Atari Modems 301	45.00
Atari Disk Drives	CALL

"CALL US LAST"

239-4315

As this add has a 45 day lead time I do not list many prices and product names. By the time you read this, we will have additional products available. CALL FOR ADDITONAL INFORMATION

You will find a new adventure in
COMPUTING at
I B COMPUTERS

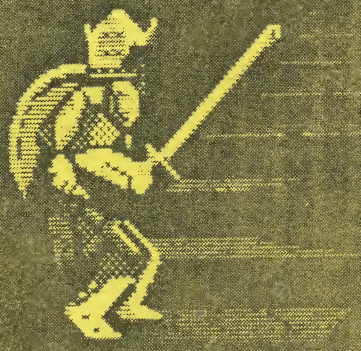


WE HAVE THE LARGEST SUPPLY OF ATARI HARDWARE AND SOFTWARE IN THE NORTHWEST

FOR PHONE ORDERS CALL 503 297-8425 (VOICE) OR BY MODEM 24 HOURS A DAY CALL IBBS 503 292-1321

WE WELCOME VISA MASTERCARD OR COD ORDERS. CATALOG ON REQUEST

I B COMPUTERS 1519 S.W.MARLOW AVE. PORTLAND, OR. 97113



PORTLAND ATARI CLUB
P.O. BOX 1692
BEAVERTON, OR 97005

BULK RATE
U.S. POSTAGE PAID
BEAVERTON, OREGON
PERMIT NO. 242