THE ATARI FORUM

A Dialog For Atari Software Developers

Volume I, Number 2 May 198

THIS IS YOUR NEWSLETTER

by Joe Ferrari
Newsletter Editor

Welcome to the second editon of THE ATARI FORUM. In the last issue, I mentioned that this is your newsletter. Please let us know what subjects—technical or marketing related—can help you be more successful.

In the last letter, I also mentioned that a developer's conference was being planned; a November timeframe is presently under consideration, just after fall Comdex. Any input regarding topics would be appreciated. Please submit all ideas in writing.

Stolen Property: Do You Copy © Software?

There has been a great deal of discussion on the issue of the illegitimate use of software programs—software piracy. Contrary to the beliefs of many Atari ST developers, this problem is universal, i.e., it is prevalent on all the widely used computer systems (e.g. Mac and IBM). It is less noticeable on these systems primarily because their installed base is larger. In the past, the software industry has attempted to curb this activity by various means: hardware devices, disk protection, and some by purposely making their programs difficult to learn—thereby forcing the user to refer to the manual (harder to copy). This latter tactic sets a software company's objectives in reverse: in order to appeal to a greater audience, software should get easier to learn, not harder.

Each of these schemes has produced mixed results: i.e., they reduced the number of illegitimate users, but at the same time reduced the size of their legitimate market by making the product inconvenient to use. During the past few years, for example, large software companies have come under considerable pressure—especially by corporate customers—to remove copy protection from their products. Before a solution can be identified, however, a closer look at the problem is necessary. There are principally three forms of activity that constitute software piracy: illegal reproduction houses, electronic bulletin board systems, and casual copying.

The first is an obviously blatant infringement of copyright laws. The perpetrator reproduces the manual and the disk(s) and sells the illegal copies through various channels, at greatly reduced prices. (The resulting product can take two forms: high quality reproduction whereby the product is a replica of the original, or a less expensive approach whereby the product is reproduced with speed printing techniques.)

The rationalization that I have heard in defense of this activity is that the publisher is defrauding the consumer with outrageously high prices. This method of thievery is more easily identified and controlled; in terms of revenue losses to the software company, it has the least impact. The second form, the electronic bulletin board, is less blatant in terms of copyright infringment. The perpetrator, in this case, usually distributes the product not for monetary gains, but to further the growth of his service. It seems that some BBSs, to be widely used, must provide users with commercially available software programs—or at least provide the platform for the exchange of these commercial products.

In early 1987, Atari Canada decided to challenge this form of piracy in the courts. A BBS in the Toronto area was openly distributing many of Atari's software products—as well as other publishers'—some of which were not yet commercially available. After some considerable effort (by Atari) to gather the appropriate evidence, the RCMP shut down this particular operation and confiscated the computer equipment.

It has been a long, up-hill struggle—and Atari was warned that it would be. "To make this work," said one RCMP officer, "you will need to educate us and the courts." The level of knowledge within the police force regarding computers was almost non-existent. Now, almost 18 months later, the case is ready to be presented in court. From my understanding, it has become a precedent-setting case and both IBM and Apple are observing it closely.

Although this particular form of piracy is widespread, the most damaging to a software company's bottom line is the casual copying. This third form

Continued on page 7

The Atari Forum was created—in its entirety—using an Atari Publishing System: MEGA 4, SLM804, Atari Deskset, Microsoft Write, and Easy Draw.

Atari Automates Product Tracking

In conjunction with the next release of the TOS operating system Atari Corporation is introducing the Product Tracking System (PTS). The major features of this system include:

- User generated System Problem Reports (SPR);
- SPRgen, an Atari ST utility to generate SPRs;
- Review of all SPRs by Atari Engineering;
- Feedback to originator, if requested;
- Maintenance of a database containing open SPRs;
- Dissemination of open SPRs to developers;
- Comprehensive documentation of SPRs in product release notes.

The front-end of the PTS database is a utility, distributed by Atari, called SPRgen. Whenever a user identifies a hardware, software or documentation discrepancy, or wishes to specify an enhancement, he/she is encouraged to use the SPRgen utility to complete a SPR. Information, such as the user's name, phone number and location, system configuration, problem summary, and procedural steps to reproduce the system problem are gathered by SPRgen. Once this information is obtained and the user has verified its accuracy, the form may be saved to a text file, which is then submitted to Atari. Atari strongly recommends users to submit the file electronically. However, a printout of the SPR form may be mailed in lieu of electronic submittal. Those connected to USENET can use the Email address hplabs!sun!atari!pts. Other means of electronic submittal will also be available. Atari will not accept handwritten SPRs, nor electronically submitted SPRs not conforming to the standard format.

Each discrepancy and enhancement submitted is immediately entered into the PTS data base, reviewed for completeness and technical accuracy, and verified for uniqueness. SPRs are rejected if they are incomplete, not actual problems, or already exist in the database. Multiple reports of a problem will be tracked internally. Acceptable SPRs are categorized (hardware, software, or documentation) and distributed internally for review. After complete diagnosis of a reported problem, Atari will, if appropriate, issue a field notice documenting the problem, along with a solution or work-around procedures. For severe problems, Atari may release a patch.

The Product Tracking System is an effective means to collate, review, and respond to feedback submitted by Atari users. Atari is responsible for maintaining the database and keeping users appraised of problems with and enhancements to Atari systems. Users are responsible for ensuring that the system is used efficiently. Before submitting a report, users should consider whether the problem/enhancement being documented is limited to their own specific environment, or whether it is one which impacts the majority of Atari users.

A User's Guide to the PTS will be provided to all developers, more fully explaining the features and operation of the system.

Both end-users and developers will benefit immediately from this controlled tracking of Atari supported products. System problems will be identified and addressed quickly. At any moment the state of a product may be accessed by Atari's management. Overall, this control will enhance the quality of subsequent product releases.

HDX 2.0 To Be Released

The new Atari Hard Disk Boot Disk includes a number of hard disk utilities, including HDX 2.0 and HINSTALL 2.0. HDX 2.0 has a number of improvements over the existing version, including support for the new MEGAFILE 30 and MEGAFILE 60 (30 and 60 megabyte drives, respectively), and improved bad-sector handling.

HDX 2.0 includes the following features:

Format: Formats and performs destructive Markbad (i.e. write, read, and verify) on a physical drive, then partitions the drive with the default scheme for its capacity. Any bad sectors reported during Markbad are logged on the hard disk Bad Sector List (BSL) and marked in the File Allocation Tables (FATs).

Partition: Partitions a physical drive and marks sectors from the BSL in the appropriate FATs.

Zero: Zeroes the FATs and Root Directory of a logical drive and marks sectors in the BSL belonging to that drive in the FATs. The boot sector is also rebuilt.

Markbad: Performs a non-destructive Markbad on a logical drive. An attempt is made to read each sector; if the attempt succeeds, the sector is assumed to be good. Otherwise, the sector is assumed to be bad, and is logged in the BSL. HDX 2.0 handles four kinds of bad sectors:

- The bad sector is part of an unallocated cluster. It is marked as bad in the FATs.
- The bad sector is allocated to a file. The user is informed of the sector and cluster number, and the name of the file it belongs to. The user can choose to delete the file, skip over the bad sector (mark the sector as bad in the FATs and relink the file without it), or ignore the bad sector.
- The bad sector is allocated to a subdirectory file. The user is informed of the sector and cluster number, and the name of the subdirectory it belongs to. The user can choose to delete the subdirectory with all files (the sector is marked as bad in the FATs), delete the subdirectory but save the files to the root directory (the sector is marked as bad in the FATs), or ignore the bad sector.

Continued on page 3

HDX 2.0

Continued from page 2

- The bad sector is part of a lost cluster. The user can choose whether to mark it in the FATs.

Ship: Parks the heads on one or more hard disks selected by the user.

HINSTALL 2.0 includes the following features:

Install: Installs the autoboot hard disk driver onto a physical drive selected by the user.

Remove: Removes the autoboot hard disk driver from a physical drive selected by the user.

COLDBOOT.PRG forces a complete and thorough system initialization. This is useful, for example, if you want to boot from a floppy after your system has been autobooted from the hard disk.

Atari Offers Marketing Support

by Neil Harris
Director of Product Marketing

Atari is interested in supporting our developers on the marketing side as well as providing technical support. There are several ways in which we can assist you.

Dealer News

Each month, an issue of the *Atari Dealer News* is sent to nearly 1000 dealers and sales reps throughout the United States. As a service to our developers, we welcome you to provide us with 1000 copies of a flyer for your product, which we will enclose with the mailing at no charge to you. We pay the postage, and we make sure your message reaches all the dealers here.

In order to participate in the monthly mailings, you must get in touch with me before sending in your materials. Let me know what to expect, and when. Please, limit your flyer to a single 8.5x11 sheet (printing on both sides is fine). If possible, orient the flyer to dealers—let them know how to order, who to order from, and at what price. And plan on having your material arrive at least one week before the end of the month so we can prepare the collation—it's a big job.

Online Support

You probably know that Atari sponsors quite a bit of support activity online. We will be glad to help you promote your product and your company using these media. Send me a copy of your press release and I will have it placed on GEnie and on the Atari BBS. Your news will reach thousands of Atari users almost instantly.

Furthermore, if you are interested in providing support online, give me a shout. Some of the best known companies in the Atari community got that way by maintaining a high profile on the boards. I will be happy to tell you what you need and to provide you with a section of the BBS on GEnie for your product. If you have something special, we may even set up a formal online conference with you. Conferences are attended by scores of Atari users, and the edited conference transcripts are hot downloads for use by user group publications.

Public Relations

Atari Computer (the U.S. computer division of Atari Corporation) has just switched to a public relations firm with many years of experience in personal computers. Our agency would be glad to help you out with your own PR needs. Winston & Winston is located at One Summit Avenue, Suite 905, Fort Worth, TX 76102. Our main contact there is Marty Winston. Marty's phone number is (817) 332-5222.

To make Marty's life easier, it would be great if you would include him on the list for any PR mailings you send out. Marty is equipped with a full Atari computer system for demos, so if you can send him samples of your software, it would help him a great deal.

Marty may be contacting you from time to time to help with editorial requests. Atari sends out systems to influential reviewers and industry analysts. The systems don't do much good without software, so depending on the needs of the particular person we may request that you send software to them. We only send equipment to people with the highest level of exposure, for us and for you.

Sales and Product Demos

Atari often does product demos for hot prospects. I often do these myself, and our force of field sales reps does them regularly as well. Again, we ask for your help. Please include me on your list of people to send copies of software at release. I'm happy to recommend good software to dealers, users, and the press—but if I don't see it, there's not much I can say about it.

If you'd like to help out our field sales force, just give me a shout and I'll give you the name and address of each rep. Send them demos and product information and they can help you out.

Remember, we all want the same thing—to sell lots of products. Atari can't sell hardware without your software. With your cooperation we can all work together and succeed.

My address and phone number:

Neil Harris, Director of Product Marketing
Atari Computer
1196 Borregas Avenue
Sunnyvale, CA 94086
(408) 745 2160

To reach me online: GEnie: NHARRIS; CompuServe: 70007,1135; BIX: neilharris; Delphi: NEILHARRIS; Usenet: hplabs!sun!atari!neil.

Floating Point Coprocessor for MEGA Computers

The Atari SFP004 Floating Point Coprocessor PCBA has been released and is available to registered Atari developers. The complete Developer's Kit will be available by late June. The card is designed to be used only with the Atari MEGA line of computers. It must be installed by an authorized Atari service center. To install the board yourself voids the warranty of the computer.

The SFP004 card contains a Motorola PLCC 68881 Floating Point Coprocessor chip, clocked at 16 MHz. An optional version clocks the chip at 20 MHz.

The card will have no effect on application performance unless the application code has been compiled to specifically support the SFP004. The application's development environment itself must be updated to support Atari's SFP004 PCBA.

Interfacing to Motorola 68000

The 68881 can be attached to the MEGA's 68000 bus only in what is known as peripheral mode, rather than co-processor mode. This means that the 68881 chip is programmed by loading data and commands into the chip's registers, and by polling its status register to determine completion or exception.

Impact on compilers

The 68881 expects operands to be presented in a binary format compatible with the IEEE P754 standard. Three of the four major compilers used widely on Atari systems support this format: the Manx (Aztec C), MEGAmax (Laser C) and Alcyon. Mark Williams C uses a different internal representation for floating point values, consistent with DEC VAX formats.

Assuming that the compiler is able to generate or otherwise handle object code compatible with the 68881, the method of compilation for 68881 compatibility must be determined. There are three basic possibilities.

In one, the programmer must specify at the time of compilation whether the compiler is to generate object code for the 68881 directly, or is to generate calls to emulation routines which are provided via the software library. A variation on this theme is that the compiler provides two libraries, with similar entry points (or routines), one of which is customized for the 68881 and one of which provides software emulation. Code which is generated assuming the absence of the 68881 will execute normally whether or not the chip is present. Code which is generated assuming the 68881 is available will fail with a bus error if executed on a system which does not have the 68881. While this approach reduces overhead in detecting the presence of the 68881, it has the disadvantage that applications need to be recompiled (or relinked) and that special versions must be provided to support the 68881 if the extra performance is to be achieved.

In a second possibility, the provider of the compiler and its associated libraries may decide to offer a single floating point library. In this case, each library routine is coded so that it checks for the presence of the 68881 and uses it if it is present, emulating the function in software is the chip is not present. The advantage of this approach is that only one new version of any application is required, and that it will take advantage of the chip if it is present.

In a third possibility, the compiler may choose to generate code to check for the 68881, and if it is not found then causes the appropriate call to a library routine to be executed. This is similar in effect to the second case, but generates even faster code since a "call" is not invoked if the chip is present. However, code length will be greater.

Atari has no control over when or how the various compiler suppliers will offer support for the 68881. Developers should contact these suppliers directly. Atari is in contact with them in an attempt to ensure support is indeed provided.

Impact on applications

As indicated above, current applications have no way of using the 68881. They must be recompiled and/or relinked to generate object code which will exploit the chip.

If the first method of compilation, as described above, is used, there will be two versions of an application, one of which will only provide software emulation but will run on any system (i.e. the version which exists today), and one of which will be specifically for systems which include the 68881.

If the second compilation method is used, a new revision of the application must be generated, but it will be able to operate on any system, taking advantage of the 68881 if it is present.

It is important to note that existing applications will work as they do now, even if a 68881 is present. There will, however, be no performance improvements.

Sample library routines

The diskette which is included with the MEGA SFP004 Developer's Kit contains source and objects files for a set of high precision mathematical routines. These routines are compatible with the Alcyon compiler, and are provided as samples only. If you wish to incorporate the routines into your programs, to verify the operation of the MEGA FPC, the routines should be linked into your application program by specifying them on the LINK68 or ALN linker command line before "libm," the standard Alcyon routines.

Questions & Answers

Here are the latest questions from the Atari developers' mailbag as answered by John Feagans, Director of Software Technology. Leave questions on Compuserve for PIN 70007,1072 or GO PCS57 for Atari developer SIG information.

BIOS

Q: How many different European keyboard versions are in production?
A: As of this date there are nine: USA, UK, Germany, France, Spain, Sweden, Turkey, Swiss French, and Swiss German. Some countries use the same keyboards but with a translation of TOS in their native language. For example, Finland uses the Swedish keyboard and Holland uses the UK keyboard.
Q: What must my application do to handle the different keyboards and character sets?
A: Since the character set and character codes are the same in all versions, the best thing to do is nothing. Let the keys come in naturally from the BIOS and do not use the scan code portion of the long word returned. The reason for this is that the 8-bit character codes are constant between all ST computer country versions but the positioning of the characters on the keys is the thing that varies. Here in the USA we use a QWERTY arrangement—in France it is AZERTY.

These comments only relate to interfacing between the application and the keyboard. Applications intended for international use must obviously consider differences in currency, date, and time formats; character collation sequences; the need to translate all text strings; and the need to accept user input in native mode. (Tip: Use function keys instead of Yes/No, etc.)

DOS

Q: What is the maximum number of files which can be opened by an application?
A: GEMDOS has a compiled constant which limits the total number of open files to a maximum of 75 at one time. This number can be reduced since it depends on the OS pool for storage of certain information. Other users of the OS pool include Malloc, directory nodes, and active subdirectories.
Q: How many files can be stored on a disk?
A: There are three parts to this answer. First, there is a limit to the number of files and subdirectories in the root directory of 256 on a hard disk and 112 on a floppy disk. Second, the file selector in the AES allows a maximum of 100 files. Third, the total number of files in desktop windows may not exceed 400. (The last two limitations have been removed with the forthcoming TOS upgrade.)

AES

Q: Why doesn't the form library respond correctly when I try to use the 6x6 system font in a dialog box?
A: It is O.K. to use the tiny font for everything except editable text fields. The logic of the object editor is independent of the character size, but the code in the AES which redraws and keeps track of the cursor position is moving in 8 pixel increments. We recommend that you do not use this font for anything other than free strings.
Q: My program uses the menu_bar call to display two menus. The application works fine but when I return to the desktop, mouse clicks are ignored on a desktop window other than the one currently selected.
A: With some investigation we were able to find that the number of clicks you have to make corresponds to the number of times you switched menu bars. We found a solution that has no harmful effects: Issue a graf_mkstate(&dummy,&dummy,&dummy,&dummy) after each menu_bar call and the mouse click problem will be cleared.

Accessories

Q: I have heard that desk accessories cannot open files. But I see many accessories which do open files—how can this be?
A: Desk accessories are a special type of application known only to the desktop program. You may already be familiar with the state of memory being shrunk when they are first executed. Also they do not have a base page of their own. It is the base page that GEMDOS uses to determine the ownership of files and blocks of memory which were Malloc'ed. Anything owned by a process is closed when that process is terminated.

If a desk accessory opens a file when the desktop first calls it then it probably will not be closed because the desktop is the parent of all applications which will be run. On the other hand, if the accessory opens a file while an application is running, the accessory may crash if the file is not closed before the application terminates. There is no message to warn the accessory of this event. Worse, the handle may be assigned to another application which is invoked with subsequent unpredictable damage to the file by the accessory.

BASIC

Q: Are there any compilers for BASIC on the ST?
A: Yes. There is one provided in the box with the ST, as well as several third-party compilers for different dialects of BASIC. Check with your software dealer for some of the possibilities.

Development Tools

Q: What am I doing wrong? Madmac does not assemble branches correctly.
A: A common error is to define symbols which are the targets of long branches as "symbol = *". You should never equate a target symbol to a constant which in this case was the assembly location counter. The problem is that at load time, the address does not get fixed up since you technically branched to a constant. Be sure to write your labels as "label:" and you will not have this problem.
Q: I have a program written in C which uses the function fopen("A:\filename"). My program never finds the file.
A: In C, the backslash is interpreted as a literal. You need to put two slashes one right after the other so things will compile correctly. GEMDOS can then find your file as you have specified.

Summary of TOS Upgrade

Atari is currently working on the new TOS upgrade, with ROMs scheduled for release later this year. Beta versions have already been shipped to Atari subsidiaries. Future issues of this newsletter will contain more information as available.

What follows is a preliminary list of the enhancements/changes to TOS, for each of the major operating system areas: Desktop, AES, VDI, and GEMDOS. Note: This list is subject to change without notice, and comprehensive release notes will accompany the release of this TOS upgrade.

Desktop

- GEM programs can be autobooted from disk.
- The Desktop now formats disks with an MS-DOS compatible boot sector.
- A file can be "moved" as well as copied.
- Copy, Delete, and Move operations can be interrupted with [Undo].
- The Copy, Delete, and Move dialog box shows destination folder and file name as the operation progresses.
- Disk Copy and Disk Format have been combined into one dialog.
- The "Format Disk" dialog now defaults to "Exit" if [Return] is pressed.
- "Install Disk Drive" has "Install" as its default, and when you install a drive that already exists, it updates the existing icon.
- "Install Application" has a "Remove" button, and "Install" is now the default.
- "Set Preferences" determines if the system confirms file name conflicts.
- Upon a "Name conflict during copy", the user has three choices: Copy, Skip, or Quit.
- Desktop shows as many files as possible inside each window, limited to available memory. This removes the static allocation limit of 400 files.
- "Show Info" now allows a folder to be renamed.
- Show/Print text file functions have been completely rewritten.
- When files are copied, the pointer changes to a "busy bee" even if the Desktop is set to copy without confirmation.
- When copying files and an error occurs, the arrow becomes a busy bee when Retry is clicked.
- Cancel or Retry now work as expected when an error occurs while formatting a disk.
- When copying disks between drives A: and B:, with no disk in source or destination, an error occurs. Picking Cancel now returns the user to the Disk Copy Dialog.
- Single drive disk copies require as few disk swaps as possible.
- If an error occurs when a drive is 'opened,' a blank window no longer results.
- All background windows are updated after a file copy, move, delete, disk copy, or disk format operation.
- When recovering from an application crash, wind_update(FALSE) is set before going into the main event_multi that waits for user interaction.
- If you try to get a directory of a drive without a disk, 'Cancel' now aborts the operation and returns you to the Desktop.
- Many dialogs are more concise.
- The Desktop's copyright notice now lists 1986, 1987, and 1988.
- Date separators in "show file info" and "show folder info" are now the "/".

AES

- The File Selector has been reworked. The improvements are as follows:
  - An application can now send a "title" string to the File Selector.
  - FS provides 16 drive label buttons, for easier drive selection.
  - FS now handles [Return] correctly on text editing: after editing a pathname, pressing [Return] enters the path and redisplays the FS. After editing a filename, the FS exits.
  - FS remembers where it was in a listing of files.
  - The static file allocation of 100 per FS has been removed.
  - FS now handles long pathnames.
  - FS now handles multiple "abort/continue" errors correctly.
  - FS preserves current DTA buffer addresses, clip rectangles, and default directories.
- The "appl_init()" call returns a version number of 0130 in global(0).
- Executing a program through the AES shell sets the default directory to the one that contains that file.
- A wind_get() call with field parameter of WF_SCREEN returns the address and length of the AES menu/alert buffer.
- Toggling between True and False on Menu Bar no longer corrupts the semaphor.
- AES now handles editable fields followed by non-editable characters in dialog boxes.
- If a diskette is removed when the file selector is called, the system now handles "Cancel" on the resulting error dialog correctly.

VDI

- 'Ptsin' has been expanded to allow 512 vertices. This has been true (but undocumented) since 4/22/87 (Mega) ROMs.
- 'vqt_extent' now works correctly when rotation is 270 degrees.
- 'vq_mouse' has been modified to be more robust.

Continued on page 7

TOS Summary

Continued from page 6

GEMDOS

- The so-called "40-folder limit" is alleviated such that reasonable use is unlikely to cause a problem. Its limits are now very far away, and can still be expanded with FOLDRXXX.PRG.
- By and large, the restrictions on 'Malloc' have been lifted.
- An exhausted OS Pool now results in predictable (and safe) behavior.
- The OS Pool has been reduced to 11/20/85 size.
- The FAT searching code for hard disks and floppies is much faster.
- Sector buffering in GEMDOS has been improved, and the user can add buffers to it to improve system performance.
- 'Frename' can now rename a folder.
- GEMDOS now prevents duplicate filenames.
- 'Ddelete' immediately following a 'Dcreate' now works correctly.
- 'Dcreate' (mkdir) now detects and handles errors.
- 'Fread' and 'Fwrite' with a length argument of zero do not hang.
- The "archive" attribute bit (0x20) is now correctly maintained.
- 'Fattrib' checks the legality of what is being attempted.
- The entries for "." and ".." in subdirectories are correctly date-stamped.
- 'Fsettime' and 'Fsetdate' cause the BIOS time and date to be set, too.
- 'Fdatime' no longer byte-swaps the user's input values when writing a new date/time.
- 'Fdatime' returns EIHNDL for invalid handles and handles which refer to character devices (which have no date or time).
- Cconws' is faster than before when stdout is redirected.
- Redirection to the printer (handle 3, "PRN:") works correctly.
- Console handling of ^S and ^C is consistent.
- 'Cconrs' has been improved.
- Character I/O functions (including 'crawio' and 'Cconout') work predictably when redirected.
- Console input type-ahead buffers are implemented correctly.
- Keyboard repeating has been improved.
- Reset is available from the keyboard.
- Program startup is now as fast as MEGA ROMs.
- Floppies are checked for "bootability" on warm and cold starts.
- Closing a standard handle (0-5) causes it to revert to its default BIOS device definition.
- Disks formatted with this version of TOS are compatible with the IBM PC.
- 'Pexec' handles exceptional cases correctly.
- 'Rsconf(-2,-1,-1,-1,-1,-1)' now returns the last baud rate value set by 'Rsconf'.
- The structure of the private part of the DTA (used for 'Fsfirst'/'Fsnext') has changed. Applications that counted on its (reserved, undocumented) structure may break.
- GEMDOS now recognizes "media change" better.

Editorial: Piracy

Continued from page 1

appears as an innocent transaction with no physical removal of goods—so it can't be classified as stealing; one user buys the product and then allows his friend(s) to copy it—just like in the music industry, when an individual buys the album or compact disk and then allows friends to tape it.

No laws exist—or can exist—that can significantly curb the casual copying. But there is one step the software industry can take that will impact all forms of software piracy and reduce this revenue-draining activity to a more tolerable level. The technique has been used successfully to curb our drinking, smoking, and even our sexual activity; it is called educating the public.

Because the activity does not involve the removal of goods, the general public is either unaware that it is immoral or they have rationalized it in such a way that it has ceased to be an issue of morality. We, the software industry, must wage an all-out effort to educate the consumer; we must let them know that it is morally wrong to accept and use a software product that they have not legally purchased.

The now defunct software publisher, Batteries Included, in 1985, began a campaign to educate computer users by publishing a poster entitled "STOLEN PROPERTY: Do You Copy © Software?" The poster states that "if you have a 'copy' of a © program which you did not buy, then you have stolen property. Don't steal software! Buy it and support the industry you've invested in."

There are many other methods that can be used: the development of an international symbol to designate "Don't Pirate © Software," the use of that symbol on all disk labels, buttons, etc. If we can get enough support from the major publishers or the Software Publishers' Association, I am sure that we can significantly curb the level of software piracy that exists today.

Available Development Kits

GDOS Developers Kit

The GDOS Developer Kit is now available to registered Atari developers. The package consists of GDOS, GDOS-compatible drivers and fonts, and utility packages for GDOS, fonts, and output. It is available free of charge for development purposes only. Before releasing a commercial product containing GDOS or any of the Kit contents, however, developers must purchase a GDOS license for $500.

The GDOS license entitles developers to use of GDOS object code (GDOS.PRG, OUTPUT.PRG, and OUTPUT.RSC executable only; GEMVDI-compatible printer drivers, and fonts designated by .fnt) in an unlimited number of MEGA/ST software products. Without purchase of a license, developers must sign an agreement that limits usage of the GDOS code. GDOS.PRG, the .sys files, .fnt files, and the related utility packages are all copyrighted and cannot be included in third-party software.

If you would like to receive the GDOS kit, send a request letter to Cindy Claveran at Atari Sunnyvale. You may also leave a message for Cindy on GEnie, but please don't call directly. Cindy will send you the appropriate documents for signature.

The GDOS Developers Kit will only be shipped upon receipt of the signed limitation agreement or funds for purchase of a license.

Upgrade Your Atari Developer Kit

The Developer Kit Upgrade is available to developers who have older Developer Kits. The upgrade includes the following:

1) Question and Answer Newsletters
2) Introduction to GEM programming (7/25/86)
3) A Hitchhiker's Guide to the BIOS (11/26/85)
4) ATARI GEMDOS Reference Manual (4/4/86)
*5) GEM Programmer's Guide, Vol. 1—VDI—Figures and .IMG Format (1987)
*6) GEM Programmer's Guide, VOL. 2—AES (1987)
7) GEM DOS Programmer's Guide
8) C Language Programmer's Guide
*9) S.A.L.A.D. (Still Another Line A Doc) (2/88)
10) Intelligent Keyboard (IKBD) Protocol
11) Engineering Hardware Specification
12) Application Notes on the ACSI (DMA port)
13) Chip specifications:
    - 6850 ACIA
    - AY-3-8910 PSG
    - 68901 MFP
    - Programmable Sound Generator Manual
    - WDI770 FDC
    - 128K ROM Cartridge Schematic
    - 520ST Schematic
    - MEGA Bus Information
    - Blitter Chip

To get the Developer Kit Upgrade you must send a check for $50.00 to Cindy Claveran. The Developer Kit Upgrade does not include any software. Please allow up to 4 weeks for delivery.

**Indicates a recent addition or significant improvement and/or rewrite of material.*

Atari Corporation
1196 Borregas Avenue
Sunnyvale, CA 94086