

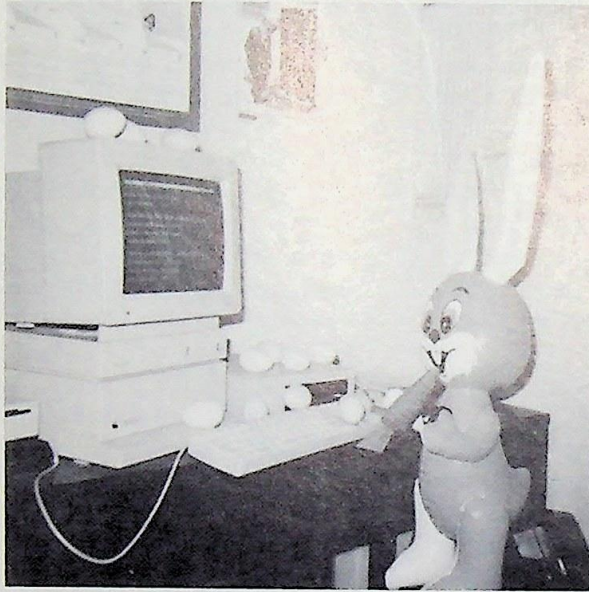


BULK RATE
U.S. POSTAGE
PAID
Chattanooga, TN
PERMIT NO. 616

March
April
1991

Volume 2
Number 4

The *First* Apple IIgs[®] Magazine + Disk Publication!



Special Easter Egg Edition!

How Many Can You Find?

(Hint: Holding down the Command key
while you read won't help.)

Feature Articles

An Interview With Dave Hecker Of Seven Hills Software
Working With The Toolbox

Programs

Quick NDA
The New Order
EGOed v1.33
Transfusion v1.1.1

Reviews

Harmonie vs. Independence • InWords • Allison Digitizing Software
MAX/Edit • The Software Of The Month Club
Super GS Award Maker • Talking Speller II • Halls Of Montezuma

Last Issue Will Be: V3.N4*
DONALD COHEN - 100CHE205W
205 W 95 ST #3E
NEW YORK NY 10025

Writer's Block

By Steven W. Disbrow

AppleFest 91

Well, if you have not heard by now, there is not going to be a Spring AppleFest in New Jersey. When I called up the folks at 1-800-262-FEST, they said that the reason there would not be a spring AppleFest was because, "we could not get Apple to attend last year, and they won't attend this year either. So, what's the point?"

This point is, AppleFest is not about *Apple Computer, Inc.*, it's about the *people* that use Apple II computers! Another reason that last year's show was such a "bust" was that they maintained that Apple *would* be there until a few weeks before the show. Then, when people arrived at the show and found that Apple *wasn't* there, they were ticked off. Of course, they could have a great AppleFest if they would just say, "Well, Apple won't be here . . . screw 'em. We'll have lots of vendors that *do* support the Apple II. Come throw your money at them!" Oh well, I suppose that if everyone calls the above 1-800 number and complains, we might just get an AppleFest this spring. While you're at it, call Apple Computer and tell them what you think. The number is (408) 996-1010. That will get you the front desk. The person you want to speak to is Ralph Russo. Be very polite.

Where's The Storm?

It certainly has been calm out there in IIGS land for the last few weeks. I just hope that the storm that follows is a good one. There are signs that it will be: business here at *GS+* is quite good, there are rumors from deep within Apple that the II will soon be making a "come-back," a new version of the System Software seems to be near release (Can you say HFS?), and Apple continues to hire people in the Apple II Business Unit. Of course, this information comes from a great many people and was pieced together from lots of different conversations. Each of these conversations began with, "don't quote me on this, but . . .," so it has to be taken with a grain of salt (and a couple of aspirin).

What Do You Think?

A new feature in this issue of *GS+* Magazine is the feedback form. If you could, take a few minutes to fill it out and send it back to us. We really want to know what you think about *GS+* and how we could make it better.

First-Class Mail

Last issue I said that we would probably raise our First-Class mail fee after the new postal rates went into effect. Well, the new postal rates are in effect and we have decided *not* to raise the First-Class mail fee—at least not yet. As opposed to blindly raising the rate, we thought we would wait a few issues and see exactly how much extra everything is going to cost. So, once again, if you want to get your *GS+* Magazine delivered to you via First-Class mail for only \$1.50 an issue, get your money in soon.

Spreading The Word

Several of you have written or called to tell us that, dammit, we need to *advertise*. One young woman I spoke with said that we were "the best kept secret" in the Apple IIGS world and it had taken months to track us down. This is a problem that we plan to address in the coming months. We will begin by advertising in some of the larger user group newsletters. Then,

when we get this whole name-of-the-magazine mess straightened out (see "Contest #4 Update" in this issue), we will begin advertising in some of the Big Name magazines.

Another step we have taken towards getting the word out is the America Online area that we are setting up. Although we had about a two-month delay in getting things going, we now have the area under development and it *should* be online by the time you get this. At this point, I do not know exactly *where* our area will be or what the keyword will be to access it. However, I suspect that if you check both the Industry Connection and the Magazine Rack, you *should* be able to find us pretty quickly. To go along with the new area, I have a new screen name, "GSPlusDiz," that I will be using almost exclusively from this point on. Please send all mail and submissions to "GSPlusDiz" from now on. You can still send mail to my old "Obnoxio" screen name, but I can't guarantee if or when I will answer it.

That's all for now. Until next time, be sure to visit us online!

Diz

GS+

Coming Soon In *GS+*

Features

An Interview with Matt Deatherage of Apple Computer, Inc.

Programs

MacZombies (Formerly "Night Of the Living Macs") by Bill Heineman! Explore a new kind of "synergy" between the IIGS and the Macintosh.

Tool Master by Joe Wankerl. Who started all those bloody tools?

Scrapper by Steven W. Disbrow. How would you like to have 10 Clipboards?

Autopilot (Formerly "PreLoad") by Joe Wankerl. Pick your own dang startup program!

Reviews

HyperStudio vs. HyperCard IIGS

McGee at the Fun Fair

Talking Classroom and Talking Multiplication & Division

CONTENTS

ARTICLES

Interview With Dave Hecker Of Seven Hills Software, Inc.....	3
Working With The Toolbox.....	14

PROGRAMS

Quick NDA.....	5
The New Order.....	7
EGOed v1.33.....	11
Transfusion v1.1.1.....	13
OS Library.....	16

REVIEWS

Harmonie vs. Independence.....	28
InWords.....	32
Allison Digitizing Software.....	34
MAX/Edit.....	36
Software Of The Month Club.....	38
Super GS Award Maker.....	39
Talking Speller II.....	41
Halls of Montezuma.....	42

DEPARTMENTS

Writer's Block.....	inside front cover
Letters.....	2
GS+ User Group Connection.....	19
Product Updates.....	20
GS+ Classifieds.....	20
Advertisers Index.....	21
GS+ Back Issue Information.....	22
Rumors, Wishes & Blatant Lies.....	23
How to Use the GS+ Disk.....	24
Icons.....	25
The Molehill.....	26
Contest #4 Update.....	27
Noreen's IIGS Tips.....	31
Apple Computer, Inc. Warranty Disclaimer.....	35
Buying Ad Space in GS+.....	45
GS+ Subscription Information.....	45
New Products.....	48
Feedback Form.....	inside back cover

No part of this magazine or its companion diskette may be reproduced without the written permission of EGO Systems. The programs on the companion diskette are NOT public domain or shareware!

GS+ is produced on an Apple IIGS using GraphicWriter III, EGOed, AppleWorks GS, and an Apple LaserWriter IINT. FingerPrint GSi is used to freeze the screen so the screen photographs can be taken.

STEVEN W. DISBROW
Publisher, Editor, Layout

NOREEN RIBARIC
Associate Editor, Layout

JOSEF WANKERL
Technical Editor

SUSAN THOEMING
Production Manager

GS+ is an independent publication,
not affiliated in any way with
Apple Computer, Inc.

GS+ is published bimonthly by:
EGO Systems
3535 Mountain Creek Road #A-17
Chattanooga, TN 37415-6734
(DO NOT SEND MAIL TO THIS ADDRESS!)

Opinions expressed in this publication are
those of the individual authors and do not
necessarily represent those of GS+.

All references to either Apple
or third party products are trademarked
and should be so noted.

If you have a submission for GS+,
send it to:
GS+ Submissions
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366

Subscription rates for 1 year (6 issues) are:
Magazine only - \$15
Magazine w/Disk - \$36
Tennessee residents add 7.25% sales tax.
Add \$9 if you want First-Class delivery.
Canadian and Mexican orders add \$9 for
Air Mail.
Other foreign orders add \$9 for surface
delivery or \$30 for Air Mail.

Send subscription orders, ads, inquiries,
and address changes to:

GS+ Subscription Services
c/o EGO Systems
P.O. Box 15366

Chattanooga, TN 37415-0366
or call (615) 870-4960

Monday-Friday 9 a.m.-6 p.m. EST
pro-gsplus BBS (615) 875-4607
2400 Baud/8 data bits/no parity/1 stop bit

Letters To The Editor

Dear GS+,

[Perhaps you should] produce back-issues as HyperStudio stacks with password protection. Include the runtime module and you have instant GS+!

Robert Deichert Jr.
Bronxville, NY

This is a good idea, but I don't think HyperStudio is the way to go. First of all, the password protection in HyperStudio only protects stacks from modification. Anyone can still run a password-protected HyperStudio stack. HyperCard IIGS would probably be a better solution because (if it is the same as HyperCard Mac) you can actually protect a stack so that no one can even look at it. Second, (the last time we checked) the licensing fee for the HyperStudio runtime module was about \$100 per stack (if you are using it for a commercial venture). This is not a prohibitive cost, but it is a consideration. Here again, HyperCard IIGS is the better choice—no licensing fee. Third, stack versions (both HyperStudio and HyperCard IIGS) of the magazine would be huge! The only one we have tried thus far (a public domain stack version of GS+ V1.N2) was over 300K and did not even include everything that was in the magazine! Even with all of these problems, this is a potentially workable idea. What do the rest of you think?

Dear Sirs,

... I would like to tell you how much I look forward to your GS+ Magazine. A few things I would like to see in future editions are "mini-courses" in computing with the IIGS. [Things like] how to use a modem, how to get the most out of different utility programs, how to get the most out of the slots that are available in the IIGS, etc.

I bet there are a lot of people like me who would like to learn more about computing on the IIGS, but only find complicated books on the subject. We need to know how to start from the beginning. A lot of

your material is rather difficult to understand when you don't have the basics.

John D. Cotugno
Castleton, NY

Dear Sir,

... I want to tell you that I consider GS+ the very finest magazine for any GS owner. You strike a fine balance between providing information which is understandable by the novice and I presume that the information which is over my head is useful to the serious user. I read it all, hoping to advance my knowledge.

I especially rely on your reviews. I could have saved some money had I read your review of "The Hunt for Red October" earlier and I intend to purchase "The Print Shop Companion GS" based solely on your review.

I enjoyed your brief reference to the founding of Applied Ingenuity. Could you do an article on the demise of the company? I purchased a 40MB OverDrive about three weeks before they closed their doors... It never did work correctly. Do you have any data on the new "Q" drives being marketed by Quality Computers?

Another subject which I have found little information on is the proper format for 3.5-inch disks. Some I have are formatted 4:1 and some 2:1. Is one more advantageous than the other?

C. E. Garrett
Hanford, CA

Thanks for the comments. It's nice to know that you trust our reviews. We try very hard to make them as thorough and informative as possible.

The Applied Ingenuity story is indeed a very interesting one. Unfortunately, all I know about it is what I hear second hand. I'd love to interview Dave Westbrook, but I hear that I'd have to wait in line behind Uncle Sam.

At this point, I have no experience with the "Q" drive. So, let it be known that we are looking for a "Q" drive review.

Disks with a 4:1 interleave should be used primarily with ProDOS 8 applications. Disks with a 2:1 interleave should primarily be used with GS/OS applications. There is no harm in mixing operating systems and interleaves; it's just that ProDOS 8 can more efficiently read disks with a 4:1 interleave and GS/OS can more efficiently read disks with a 2:1 interleave. I almost never use ProDOS 8, so I format all of my disks with a 2:1 interleave.

Dear GS+,

I have a PC Transporter installed in my IIGS. I use the PC Transporter's memory as a RAM disk, the default name of which is RAMAEP. Here is the problem: RAM Namer (GS+ V2.N2) renames the PC Transporter RAM disk and leaves the RAM5 RAM disk unchanged.

Ulrich Droegehorn
Germany

As far as RAM Namer is concerned, the PC Transporter RAM disk looks just like the System RAM disk. So, the problem is that RAM Namer only changes the name of the first RAM disk that it finds, in this case, your PC Transporter RAM disk. To be honest, I had simply forgotten that it is possible to have more than one RAM disk online. The next version of RAM Namer will address this problem.

If you have a question, comment, or criticism about GS+ Magazine, we want to hear it! Due to space limitations, we cannot answer every letter here in GS+ Magazine. If you want a personal reply, please enclose a self-addressed, stamped envelope. Please address all letters to:

GS+ Letters
P. O. Box 15366
Chattanooga, TN 37415-0366

GS+

Interview: Dave Hecker Of Seven Hills Software, Inc.

We first met up with Dave Hecker at the 1990 AppleFest in New Jersey. We had just switched to GraphicWriter III for the layout of *GS+*, and we had a bug list about 1.5 kilometers long. Dave listened patiently to everything we had to say and then, to our amazement, acknowledged just about every bug we had found! The rest of the bugs had not yet been reported to Seven Hills and he was very pleased to get the information. After that, we spent the next hour or so going over suggestions we had for the program. When we finished, Dave admitted that we had some good ideas, but that he really could not promise a delivery date for GraphicWriter III v1.1.

He did, however, promise to help us in any way that he could. In the year since then he's kept that promise—and then some. Dave put us in contact with our current printer, which has saved us approximately \$5,000 in 10 months. Dave also provided us with the specs for GraphicWriter III translators, which Joe used to write the TeachText Translator (*GS+* V2.N2 and *GS+* V2.N3) that we use to import formatted text created with EGOed. The amount of time this has saved us is immeasurable! Without Dave's help, it's quite likely that *GS+* would be a quarterly magazine!

So, when it came time to pick our next interview subject, Dave was a natural choice. The following interview was conducted on America Online during the months of February and March 1990. (Some of the questions for this interview were provided by Brian M. Winn.)

GS+ Tell us a little about Seven Hills Software. When did you get started in the Apple II market?

DH Seven Hills has published software for Apple IIs, starting with the Apple II+, since 1982. Our company consists of seven hard-working people who are strongly committed to creating and supporting the best IIGS software in its class. We publish software written by some of the best IIGS programmers in the world.

GS+ How and why did Seven Hills Software get involved with the Apple IIGS?

DH We were excited about the creative opportunities the IIGS offered us. The chance to publish GraphicWriter III presented itself and it was our first IIGS-specific project. It was great introduction to IIGS-specific software!

GS+ Give us a brief run down of your IIGS product line.

DH Disk Access is a Finder-like disk utility that comes as an NDA so a IIGS user doesn't have to quit his program and return to the Finder just to copy, rename, or delete a file, or any number of common tasks. This program is definitely a "sleeper." We consistently get registration cards saying "I don't know how I lived without it!" I just wish more people understood what it does, because it is a truly great program.

Font Factory GS is a font editor that is used primarily for resizing existing fonts to achieve the best screen and printed output. Of course, it can also produce new fonts and custom characters.

GraphicWriter III is a sophisticated desktop publisher which offers page layout, word processing, and graphics. We've been pleased to see the results sent in to us by so many people who use it for newsletters, in the classroom, for small businesses, and for other reasons. We're especially proud that *GS+* is produced with it!

Independence enables IIGS owners to get excellent 300 dots-per-inch print quality from Hewlett-Packard printers without having to purchase an expensive laser printer. It works with GS/OS compatible programs like GraphicWriter III and AppleWorks GS.

SuperConvert is our newest product and is definitely going to make waves. It not only converts graphics from a ton of different computers and formats for use with the IIGS, but it's also used to convert 320-mode graphics to something that will look good in

640-mode applications (and vice-versa). It has a lot of really nice features.

GS+ There has been a bit of controversy about how the Independence and Harmony (from Vitesse) printer drivers came into being. Give us your side of the story.

DH Our record of integrity speaks for itself. Let's just say that our business philosophy differs 180 degrees from theirs.

GS+ What, in your opinion, puts the Independence printer drivers ahead of their competition?

DH From the beginning, Independence was the only Hewlett-Packard printer driver that worked properly. What you see on the screen is faithfully reproduced on paper ("WYSIWYG"). Also, we provide a font disk that contains two families of the large fonts that are required to get the best output. Finally, our comprehensive 64-page manual provides complete setup, installation, and reference information, as well as helpful suggestions and compatibility information. And Independence does all that for \$10 less!

GS+ As you mentioned earlier, you are working with Jason Harper on the long awaited SuperConvert—a follow-up to his popular shareware graphics conversion program, SHRConvert. How did Seven Hills hook up with Jason and what new features can we expect in SuperConvert that will set it apart from its shareware ancestors (and the competition)?

DH Jason was looking for a publisher for SHRConvert so he could make it available to everyone, instead of just those who had electronic access. One of Jason's friends stopped by our booth at AppleFest in New Jersey and said Jason wanted either Seven Hills or Milliken to publish an improved version of SHRConvert commercially.

We looked at the program and were interested, so we began talking with him. During our talks it became evident that he

was another "find" like Steve Stephenson [the author of Disk Access and Independence]. So we worked out an agreement and are proud to be working with him! Jason and Steve are both extremely talented programmers who are fair and honest to deal with.

To set the commercial version apart from SHRConvert, we knew several things had to be done. Mainly, the program had to have some significant enhancements over the shareware version, and we wanted the name to be different yet similar enough to make a connection. Fortunately these were easily accomplished. We chose the name "SuperConvert," and Jason already had the "significant enhancements" in the first copy we saw. We just helped refine a few things, and added one major feature, and the program was ready to beta test.

GS+ What are some of those "significant enhancements?"

DH The concept of a "true color image" has been added. Graphic formats that the IIGS cannot directly display are loaded into memory as a "true color image." From that image you then can generate images that can be displayed on the IIGS. Several different options can be specified, and each variation is produced from the original true color image for the best quality.

Printing is new, and it has some nice capabilities (such as printing wall-size posters).

Also, several "extras" are included that aren't directly related to converting graphics, such as generating "font key charts" and "font sample pages."

Finally, more load and save formats have been added. One neat option is the ability to make any image appear as your "desktop background" instead of just the blue desktop.

SuperConvert is different from the competition in several ways. The biggest difference is that SuperConvert converts graphics from a wider variety of sources into graphics that IIGS-specific applications can use. Conversions are controlled with a variety of simple but powerful options—you can specify the graphics mode, the palette,

and a variety of other options when you "remap" an image. This gives you the power to convert 320-mode pictures into something a 640-mode program can use, or convert a full-color Atari picture into a 320-mode gray-scale picture, and so on.

GS+ Currently, GraphicWriter III is not done in-house at Seven Hills. Tell us a bit about the relationship between Seven Hills and DataPak software.

DH DataPak is responsible for the coding of GraphicWriter III and we handle all other aspects. But after version 1.1 is released that situation is due to change! DataPak has commitments to several Macintosh projects, so the development time on GW III is slower than either companies prefer. As a result, we've mutually agreed that it would be best if Seven Hills assumed responsibility for the coding of the program. We plan to speed up the development cycle of GraphicWriter III by contracting with one or more outside programmers to implement the many ideas we have—we'll see how it goes after we get a final v1.1.

GS+ When can we expect to see GraphicWriter III v1.1 update?

DH Progress continues to be made on GW III v1.1, but we still have not reached beta-test stage. Until we send a beta copy out for testing and get some feedback, we won't know how close we are to release of the new version.

Version 1.1 is mainly a bug fix version, and we plan to spend whatever time it takes to make it extremely stable. There are a couple new features though: the Line menu has been changed so a very thin line is available, and the Page Setup option now also provides access to the printer driver's Page Setup dialog box. This allows setting of special printer-specific features that could not be controlled before.

GS+ What will we see in v2.0 of Seven Hill's Font Factory GS program? Anything to do with PostScript fonts?

DH Version 2.0 will allow for smoothing or application of a style (e.g. italic) to the entire font at one time, instead of character by character. Also, using Apple

File Exchange, Macintosh fonts can be converted for use on the IIGS (great because there are so many specialized Mac fonts available). We're hoping to add some additional font translations before we begin beta testing. Font Factory GS is for generating *bitmap* fonts that improve the screen and print quality—it does not do anything with PostScript fonts.

GS+ What can we expect to see from Seven Hills Software in the next year or so?

DH There are currently three products under development, and several more are in the early planning stages. I can't discuss anything specific, but they all will be high quality and exciting. Of course, existing products will continue to be supported.

GS+ What would *you* like to see happen in the IIGS market in the next year or so?

DH I'd love to see the IIGS go mass-market. Current IIGS owners would be reassured that there really is a future with the IIGS, and we would have a larger market to support the development of our long list of new IIGS products we'd like to create.

GS+ If you had 15 minutes of John Sculley's time, what would you say to him?

DH I'd only need a few seconds: Your hardware and software engineers have done a great job. Now support the IIGS with some promotion!

GS+ Is there anything else you would like to add?

DH Yes. We appreciate the tremendous amount of support we receive from Apple, our customers, our Partners (user group representatives), and others in this business. That support lets us continue doing what we enjoy.

For more information on Seven Hills Software's products, contact them at:

Seven Hills Software, Inc.
2310 Oxford Rd.
Tallahassee, FL 32304
(800) 627-3836
(904) 575-0566

GS+

The IIGS world has waited long enough! Are you tired of always having to move your mouse to the Apple menu, pressing the button down and holding it, moving your mouse to your favorite NDA, then finally releasing that button? That's a lot of work, and a lot can go wrong—like letting up on that mouse button too soon! Well I have the solution to your laziness, and it's called *Quick NDA*.

Quick NDA provides key equivalents for your favorite NDAs. I got tired of NDA authors putting in Command (Open-Apple) key equivalents in their NDA definitions because, no matter how obscure the key equivalent was, I can probably find an application that uses the same key sequence for something, and that NDA would mess it up! This really annoys me. So, as an alternative to defining Command key equivalents to NDAs, I present Quick NDA, which defines control-keypad equivalents instead.

Some Installation Required

To install Quick NDA, use the Installer program on your *GS+* Disk. If you need help using the Installer, see the "How to use the *GS+* Disk" article in this issue. After you install Quick NDA, you must reboot for it to be available.

Configuration

Before you can benefit from Quick NDA, you have to configure it. Go to your favorite desktop application which supports NDAs and press *control-keypad-* (period). Note the word "keypad" in that keystroke. That means that you must press the period that's on the *numeric keypad*, not the main keyboard period key. This key sequence will present you with Quick NDA's configuration window.

The window you see has two pop-up menus and three buttons: *About*, *Cancel*, and *Okay*. The controls you should be interested in right now should be the two pop-up menus, labeled *Key* and *NDA*. You use these two pop-up menus to assign keypad-keys to NDAs.

As a standard, I suggest that everyone assign keypad-key 0 (zero) to the Control Panel NDA. To do this, first select 0 from the Key menu, then select Control Panel from the NDA menu. This tells Quick NDA that when you press control-keypad-0, you want to bring up the Control Panel NDA. You can configure other keys to other NDAs in a similar manner. Once you have all your assignments made, you can click on the Okay button to keep the assignments, or you can click on the Cancel button to disregard any changes you may have made. (As you might expect, the About button displays a typical about dialog.)

Now that you have assigned the Control Panel to keypad-key zero, why not test it? Hold down the control key and press the 0 (zero) key on the keypad. Before long you will see the Control Panel!

If you ever have the urge to delete a keypad assignment, simply bring up the configuration window (control-keypad-), select the keypad number you no longer like, and select the "< clear >" item from the NDA menu.

Why Is It Beeping At Me?

Quick NDA will beep at you if it is unable to perform a function that you have requested. For example, if you press control-keypad-0 to try to bring up the NDA associated with it, but the NDA assigned to that key is "< clear >," Quick NDA will beep at you.

If the front window is an alert window (if it has an alert frame) and you press the control-keypad sequence to either open an NDA or configure Quick NDA, Quick NDA will beep at you because no windows can be overlaid on an alert window. Also, since the Quick NDA configuration window is an alert window, you can't open it again when it's open, and you can't open any NDAs either.

Reconfiguration

If you add or remove NDAs from your system, you will probably have to reconfigure Quick NDA. The reason for this is that Quick NDA maintains a table of key

assignments based on NDA *numbers*. Numbers are assigned to NDAs according to the order in which they are loaded by the system. When you remove an NDA from your system (either by deleting it or making it inactive), the relative position of the NDAs in your system will change and a number that was previously valid may disappear. Quick NDA will beep at you if you attempt to open an NDA which has a number that doesn't exist. Also, if you add an NDA, the relative position of NDAs in your system might change (if there was a "hole" in the directory structure). In that case, all the key assignments will be off by one number, and when you use Quick NDA, you *will not* open the NDA which you thought you were going to. By the same token, if you reorder the position of the files in the `*:System:Desk.Accs` directory, the load order will change and Quick NDA will have to be reconfigured.

I want to know what you think of Quick NDA! If you have any comments or suggestions, be sure to send them in. And, as usual, if you find a problem with this program, fill out the problem form supplied on your *GS+* Disk and let me know about it.

Programming Considerations

Quick NDA is a *Permanent Initialization File* (PIF), written in ORCA/M, that installs a task in the RunQ. The RunQ task monitors the keyboard modifier register and looks for a control-keypad combination. If it finds one, it reads the key that was pressed and acts on it.

Quick NDA also uses resources! Before any Resource Manager calls can be made, a little bit of work has to be done. First, the current resource application ID must be saved. Next, the Resource Manager must be started to make Quick NDA the current resource application. After that, all Resource Manager calls can be made. The `OpenFork` procedure does this, and it also opens Quick NDA's resource fork.

When Quick NDA is done, it has to return the Resource Manager to the state it was in

before it took control. This is done in the `CloseFork` procedure by first closing Quick NDA's resource fork, then shutting down the Resource Manager (for every `StartUp` call there must be a `ShutDown` call), and finally setting the current resource application ID back to what it was before Quick NDA took control.

Part I: Installing

There are four distinct parts to Quick NDA. The first part is the installation code. This code is executed when the system boots and executes Quick NDA. First, the current list of key equivalents is loaded from Quick NDA's resource fork, then it is detached so it remains in memory. If it's going to remain in memory and be happy, it has to be unlocked. The final event in the installation procedure is the addition of the monitoring `RunQ` task.

Part II: Monitoring

The second part of Quick NDA is the monitor task. This is called as often as possible by the `RunQ`. It checks the keyboard modifier register for a control-keypad sequence, and if it doesn't find one, nothing happens. If it does find one, it determines what kind of key was pressed (either a number or non-number) and opens an NDA or displays the Quick NDA configuration window.

Part III: Opening An NDA

The third part of Quick NDA opens an NDA. First, the front window is checked to see if it's an alert window. If it is, the speaker is beeped and nothing else happens. If the front window is not an alert window, or if there is no front window (no windows are open), it is safe to proceed and attempt to open an NDA.

The handle to the key equivalence table is then locked so it can be properly checked. The number of the NDA corresponding to the keypad key is then retrieved from the key equivalence table, and the handle is once again unlocked so the Memory Manager can do its job. If the NDA number is zero, the speaker is beeped and nothing else happens. (Zero corresponds to the "< clear >" item.) Otherwise, the NDA is opened with the `OpenNDA` tool call. If, for some reason, the NDA couldn't be opened, the speaker is beeped to signify the error.

Part IV: Configuring

The fourth part of Quick NDA assigns NDAs to keypad numbers. First, the front window is checked to see if it's an alert window. If it is, the speaker is beeped and nothing else happens. If the front window is not an alert window, or if there is no front window, it is safe to proceed and open the configuration window.

Before the window can open, Quick NDA's resource fork must be in the search path, so the `OpenFork` procedure is called. After that, the configuration window (which is initially invisible) is opened with the `NewWindow2` tool call. The handles to the two pop-up menu controls are determined and saved for use throughout the configuration section. Next, the names of all the current NDAs are added to the NDA menu with the `FixAppleMenu` tool call, the current key equivalence handle is locked, and the NDA menu value is set to correspond to the NDA which is equivalent to the Key menu value. Finally, the configuration window is made visible.

A main event loop for Quick NDA is then entered. The task mask for `TaskMaster` is set to do minimal activities—most importantly, *no update events should be handled by TaskMaster!* The reason for this is because if an update event is generated for a window that the host application owns, `TaskMaster` will attempt to redraw the window's contents. If that window has controls that are resources, the machine might crash when it tries to draw them because Quick NDA has control of the Resource Manager, and the host application's resource fork is not in the resource search chain!

The Key Menu

Whenever an item from the Key menu is picked, the NDA menu must be adjusted to reflect the NDA assigned to that key. The technique to do this is simple. The new value of the Key menu is read and then the corresponding NDA is retrieved from the equivalence table. If the value is zero then the NDA menu is set to display the "< clear >" item. If the value is nonzero then the appropriate NDA must be selected. To make the change to the

NDA menu, its value is simply changed to the number of the NDA because `FixAppleMenu` assigns the menu item numbers exactly as it assigns the numbers to the NDAs in the Apple menu.

The NDA Menu

Whenever an item from the NDA menu is picked, the equivalence table must be changed to assign the NDA that was picked to the current value in the Key menu. This process is also simple—the NDA menu value is read, and if the value is "< clear >," it is translated to a zero value and the equivalence table is changed.

The Cancel Button

Since all changes that are made when the configuration window is open only take place to the equivalence table in *memory*, it is easy to get rid of them when the Cancel button is clicked. The equivalence table memory is disposed of, and the old table is loaded from Quick NDA's resource fork again. The quit flag is then set to drop Quick NDA out of the main event loop.

The Okay Button

When the Okay button is selected, the old key equivalence table in Quick NDA's resource fork must be replaced by the new table in memory. Since the table in memory has been detached, a `MarkResourceChange` call cannot be made. Instead, the old table is removed from the resource fork and the new table is added. When the table is added, the Resource Manager takes control of the memory associated with the table. Quick NDA can no longer reliably use it after the Resource Manager shuts down. To get around this, a new copy of the table is loaded and detached. The quit flag is then set to drop Quick NDA out of the main event loop.

As with my other PIF programs, Quick NDA seems small and simple, but the service it provides is so useful you'll soon wonder how you got along without it. It makes just about any desktop application more manageable. I hope this program makes your computing days a bit easier.

GS+

The New Order

By Josef W. Wankerl

Managing a hard drive isn't too much of a hassle, but sometimes you need a little bit of help. A utility that I consider to be vital to hard drive maintenance is something that will sort directories. I've always wanted to have a utility that would sort my directories, so when it was suggested that I write one, I did. *The New Order* is a New Desk Accessory (NDA) that will reorder the contents of your directories.

Using The New Order

To install The New Order, simply use the Installer program provided on your GS+ Disk. For help on using the Installer, see "How to Use the GS+ Disk" in this issue. After The New Order is installed, you can use it by selecting it from under the Apple menu from any desktop application that supports New Desk Accessories.

When you open The New Order, you will see a window with two controls: a pop-up menu and a list control. Initially there are three items active in the pop-up menu: *Open Directory*, *Close*, and *About*. *About* brings up a typical About dialog giving information about the program. *Close*, of course, closes the window. *Open Directory* brings up a Standard File dialog which allows you to navigate through your disks and choose a directory to reorder. To choose a directory, you must be *inside* it (i.e. you must select the directory name from the list, open it, then choose the Accept button.)

Once a directory is chosen to reorder, you will see the contents of the directory in the list control and all of the pop-up menu items will be selectable. Note that the order of the items in the list control will *almost always* be different from the order of the items in the Standard File dialog. This is because Standard File always sorts the items before it displays them. The New Order shows the order of the files as they actually are on the disk.

You can now start to play with how you would like the files arranged within that

directory. The *Sort* item will sort the files in the list control alphabetically. If you would like to have your files in a different order, simply select the files that are in the wrong position and use the *Up* and *Down* items to change the position of your files.

Once you have the files in their final order, select the *Reorder Directory* item. This will make your changes in the directory order permanent. It takes a little while for the reordering to take place, so be patient.

Limitations

Directories that have open files in them cannot be reordered. If you attempt to reorder a directory that contains open files, you will be notified of the situation and the directory will be left alone. Also, you cannot reorder the root (topmost) directory of your disk, but this is just a program limitation and may be corrected in a future version.

Reorder Method

The New Order uses a *file system independent* ordering technique. This means that The New Order can be used to reorder files on volumes formatted with the ProDOS, AppleShare, or High-Sierra file systems. Those are the current file systems that GS/OS can work with. If Apple ever gives GS/OS the ability to work with other file systems, The New Order should automatically be able to sort directories on volumes formatted with the new file system.

Most directory sorting and reordering programs currently available assume the ProDOS file system structure and take advantage of the defined directory file format to quickly rearrange the order of the files. The New Order doesn't make *any* file system assumptions. It creates a temporary work directory, moves the files from the source directory to the work directory, then moves the files back to the source directory in order. When it's done, the files are in order. It takes a little longer to arrange your directory with The

New Order, but the method is *much* safer than directly modifying the innards of the directory file. If anything goes wrong during the rearrangement process of The New Order, the worst result will be that half of the files will be in the source directory and half will be in the work directory. This is easily corrected by moving the files back using the Finder or any other GS/OS file utility. If anything goes wrong during the rearrangement process of a program that directly modifies the directory structure, there's the possibility that the files in that directory will be lost.

It would be faster to just move the files from the source directory to the work directory, then rename the work directory to the source directory's name, but that would change the order of the files in the parent directory of the directory that you are working on. I believe that preserving the parent directory order is worth the time it requires.

But Wait, There's More!

Just kidding. That's about all there is to using The New Order. If you have any problems with The New Order, or you can think of some enhancements, be sure to let me know about it. Fill out the problem form supplied on the GS+ Disk and send it in to help me figure out what's going on when you think you've found a bug.

ORCA/C, Or Not ORCA/C, That's The Question

Even though The New Order is very useful, the source code for it is, perhaps, even more so because of the methods it uses to do various tasks in ORCA/C. Despite all the bugs that version 1.1 of ORCA/C still has lying around, it's a great language to program with. The New Order source code is an example of how to do all kinds of nifty tricks from ORCA/C, such as interfacing with libraries that don't follow ORCA/C's calling conventions (like my OS Library, as well as Apple II DTS's Fake Modal Dialog library), doing compares for the List Manager's sort

routine, and how to code a function entirely in assembly language. In addition to that, this program generically demonstrates how to use resources from a NDA, how to use the List Manager to create and manipulate lists, how to make a custom Standard File dialog that only selects directories (useful for when you just want to know where things are or where things should go), and how to save and restore prefixes around standard file calls.

Assembly From C To Shining C
 Yes, The New Order was written completely with ORCA/C. I did, however, use functions written entirely in assembly language to help me out. Confusing? There are two ways to put assembly language statements in an ORCA/C program: the first way is to code the statements with an inline assembly statement (Example: ASM { statement; statement; }) and the second way is to code an entire function in assembly language.

When the List Manager sorts the list, it can either use a standard compare routine or a custom one that you write yourself. When you use a custom compare routine, the carry flag determines whether an item should be swapped with another (for complete information on custom compare routines, see page 11-24 in the *Apple*

IIGS Toolbox Reference: Volume 1). Writing a function to do a compare for the List Manager using only ORCA/C and nothing else is impossible. But with a little help from assembly language, this procedure can be written with no problems. The reason why the function can't be done entirely in C will become apparent in this next, rather technical, section.

At the start of each C function, ORCA/C normally generates code, which creates a *stack frame*. A stack frame is where your local variables are stored for each function. What happens is that the direct page is mapped onto the stack (see the "Framing Pictures" sidebar for a graphical explanation). Code is generated at the end of each function to remove the stack frame. Specifically, this code subtracts the length of the stack frame from the current stack pointer and then sets the stack pointer to that value. In order to do a correct subtraction, the carry flag must be set before the subtract instruction is executed. As you can see, there is absolutely no way to return a result using the carry flag from a pure ORCA/C function.

By using an assembly language function as a shell, however, the carry flag can be used to return a result reliably. Coding a function in assembly language has the

benefit of not requiring ORCA/C's standard function entry and exit code. The CustomSort function in The New Order is an assembly language function. It calls a custom comparison routine written in C using the standard ORCA/C function calling protocol, and then sets the carry flag based on the C routine's result. With this method, a custom item compare routine can be written using C instead of assembly language.

Library Interfacing

Calling a library routine from ORCA/C is easy if the routine follows the ORCA/C function calling protocol—all you have to do is prototype the routine. Calling a library routine that doesn't follow the ORCA/C calling convention requires a bit more work, though. Why on earth would someone write a library routine that doesn't follow ORCA/C's calling conventions? Flexibility! The Fake Modal Dialog (FMD) tools, as well as my OS Library (OSLib) routines, are called using Toolbox calling conventions (i.e. pushing result space on the stack, pushing input parameters on the stack, calling the routine, and finally removing results from the stack). To call a routine that doesn't follow the ORCA/C calling convention, you have to use inline assembly statements. To use the FMD or OSLib routines, simply push the required parameters on the stack and JSL to the

Framing Pictures - Part 1

A stack frame is created by taking space on the stack and making it the direct page. When a routine is entered, the parameters and the return address are on the stack, as depicted in Figure 1.

Figure 1

Stack Offset	
+06	Previous Contents
+04	Input Parameter
+01	Return Address
+00	----- <- SP DP = \$wxyz

A stack frame is then created with enough space on the stack for all local variables (in this case, 4 bytes), as depicted in Figure 2.

Figure 2

Stack Offset	
+0C	Previous Contents
+0A	Input Parameter
+07	Return Address
+05	Local variable 1
+03	Local variable 2
+01	Old DP Register (\$wxyz)
+00	----- <- SP and DP

function name, then remove any results from the stack. For ORCA/C to recognize the JSL name you have to have it prototyped as extern.

OSLib's TraverseNames And ORCA/C

With the TraverseNames routine in OSLib, you have to specify a routine that will be called for each name in the multi-file reply record. An ORCA/C function can't be called directly, so, similar to the custom sort trick, an assembly function is used to transfer control to the C function and then return a result on the stack.

Managing Lists

Perhaps the sketchiest documentation ever created by Apple is on the List Manager. Once you know how to use the List Manager, however, it all makes a little sense; but for someone who wants to learn how to use it, the documentation doesn't provide any clues into how things are done.

Part A: Creating The Control

The first part of using a list is to create it. I'm only going to discuss using the list as an extended control, as that is probably the most frequent way lists are implemented. You need to set up your control template as described in the *Apple IIGS Toolbox Reference: Volume 3* on page 28-57. This is not too hard to do, but be sure that

you have your moreFlags and listType bits set correctly. Next, create your window and add the control to it (NewWindow2 can do this automatically if you have your window template set up correctly, or you can just use NewControl2 to do it). With The New Order, I don't have any default members, so my listRef field is NULL and my listSize field is zero.

Part B: Adding Members To The List

Adding an entry to a list is also a simple process, but it requires work from your program. The List Manager doesn't have an "AddMember" call. There are many ways in which an entry can be added to a list, but I am going to describe the dynamic way to do it. Apple has sample source code that describes how to do it statically, but, as linked lists are usually superior to arrays, the dynamic method can handle more cases than the static method.

Each entry of a list is referenced by a data structure called, oddly enough, the listRef. The listRef is an array of list entries and each list entry is listMemSize bytes long (this field is defined in the control template). The variable entry size allows you to attach additional information to each list member; however, I'll reserve myself to talking about the standard 5-byte-long

member size and defer the discussion about how to effectively use this field to a later article.

The standard 5-byte-long member entry is divided into two fields: the memPtr field and the memFlag field. The memPtr field is a reference to the string that is associated with that list entry. The memFlag field describes what kind of reference the memPtr field is and the state of the entry (highlighted, unselectable, etc.). Bits 0 and 1 of the memFlag field determine what kind of reference the memPtr field is:

00 = memPtr is a pointer to a string
01 = memPtr is a handle of a string
10 = memPtr is a resource ID of a string
11 = invalid

The type of string is defined in the listType field of the control template. Bits 7 and 6 of memFlag describe the selection state of the entry as being selected, unselected, or disabled. Bit 5 of memFlag describes whether the item can be selected or not.

Your application, not the List Manager, has to maintain the listRef data structure! When you add an entry to the list, you must correctly resize the listRef memory and set the memPtr and memFlag fields for the new entry.

Framing Pictures - Part 2

The code to allocate this stack frame is depicted in Figure 3. Note that after the stack frame is set up, the input parameters are easily accessed by direct page accesses.

Figure 3

```
tsc      ;Transfer stack pointer to
         ;accumulator
sec      ;Prepare for subtraction
sbc     #$0004 ;Subtract enough space
         ;for local variables
tcs      ;Create the local variable
         ;space on the stack
phd      ;Save the old direct page
         ;register
tcd      ;Set the new direct page to
         ;equal the stack pointer
```

After the procedure is done, the stack frame must be removed so only the return address is on the stack and any input parameters have been removed, as depicted in Figure 4.

Figure 4

```
Stack
Offset
+04 | Previous Contents |
+01 | Return Address   |
+00 |-----|<- SP DP = $wxyz
```

The `RealAddToList` function in *The New Order* demonstrates how to do this. Once you have added your entry to the `listRef`, you have to let the List Manager know about it, and the best way to do that is with the `NewList2` call documented on page 35-6 of the *Apple IIGS Toolbox Reference: Volume 3*.

Part C: Removing Members From The List

Although *The New Order* doesn't remove items from its list, the procedure to do so is simple. (A future issue of *GS+* will include an example of this.) Basically, you just delete your entry from the array, move all the members that are after the entry you want to delete up one position, then resize your `listRef` memory to subtract the empty array entry that will then be at the end of the `listRef` array. To let the List Manager know about your deletion, simply make a `NewList2` call.

NDA Prefix Saving

Battery Brain was the first program I wrote that kept the host application's prefixes around Standard File calls. That idea proved so useful that *GS+* has adopted it as a standard for all NDA and CDev's that use the Standard File tool set. With

Battery Brain, I wrote routines in C that would get and set prefixes, but I wanted something that I could use from anywhere, so I included them in my OS library. It's true, it would have been easier to use my C routines in *The New Order*, but I wanted to demonstrate how to use the more generic OSLib routines from C.

Both prefixes 0: and 8: must be saved and restored around Standard File calls. Before the Standard File call is made and after the prefixes are saved, both prefixes 0: and 8: are set to the last path that *The New Order* was in. After the Standard File call is made, only prefix 8: is saved because it holds more information that prefix 0: can. (Prefixes *: and 0: through 7: can only hold 64 characters, while prefixes 8: through 31: can hold about 8000 characters.)

Be sure that the prefix you save is either unlocked or successfully added to your program's resource fork when your NDA or CDev exits. Having locked memory lying around is not very nice to the Memory Manager. Make sure that your handle is locked when your NDA or CDev gets control. Dereferencing and using memory from an unlocked handle might

be even worse than leaving it locked when your program doesn't have control.

Resources

Using resources from a NDA isn't hard at all. The steps are outlined in the IIGS TechNote number 71 which is provided, for your convenience, on the *GS+* Disk. In order to make things a bit easier, I wrote two functions that open and close the NDA's resource fork. They preserve the level and make sure it is zero before any calls are made to open or close. Additionally, when the resource fork is to be opened, the system preferences are set to allow single drive users to insert the disk on which the NDA resides (which will typically be the system disk, but it doesn't have to be).

Order, Order In The Court

Like I said, the source code for this program demonstrates all kinds of nifty techniques. If you have any questions about *The New Order* or suggestions for improvements, please don't hesitate to contact me here at *GS+*. If you have a problem, be sure to fill out the problem form supplied on your *GS+* Disk and then get that information to me. *GS+*

Framing Pictures - Part 3

The code to remove a stack frame is shown in Figure 5. Figure 5 should be viewed in conjunction with the stack frame model in Figure 2, except that all stack references are two less after the direct page register has been restored to its previous value. When moving the return address past the input parameters, first the bank byte and high byte of the return address are moved, then the low byte and a garbage byte (actually the high byte of local variable 1) is moved. When the stack is resized to take off the parameters, the stack pointer is put just below the return address low byte. Moving the extra garbage byte was more efficient than switching to an 8-bit-wide accumulator, moving one byte, then switching back to a 16-bit-wide accumulator. Note that by using a stack frame, a procedure can call itself recursively. Local variables are re-created for each invocation of the procedure. The code below is my own. It is very similar to the code that ORCA/C generates.

Figure 5

```
pld                ;Restore the old direct page register
lda                $06,s    ;Get bank and high byte of return address
sta                $08,s    ;Move it past the input parameters
lda                $04,s    ;Get low byte of return address and some garbage
sta                $06,s    ;Move it past the input parameters
tsc                ;Transfer stack pointer to accumulator
clc                ;Prepare for addition
adc                #$0006   ;Add enough space to recover input and output parms
tcs                ;Set the new stack pointer
rtl                ;Return to caller
```

The main focus of this EGOed update was to add a few new editing features that we needed in order to make our work on the magazine easier. After you install EGOed v1.33 (see "How To Use The GS+ Disk"), the first thing you will notice is the addition of a separate font size menu (font sizes were previously selected from the Style menu) and that the Preferences menu has been renamed "Prefs." After the Size menu was split off from the Style menu, the change to "Prefs" was necessary to make all the menu titles fit in the 320 mode EGOed window.

When you actually pull down a menu, you will notice yet another cosmetic change. This version of EGOed uses real dividers to separate menu items, not just underlines. This makes the menus a bit longer, but it looks more professional and it shows the logical groupings of the menu items more clearly.

Menu Changes

When you first pull down the File menu, the only change you will notice is that the Info and About EGOed items are now grouped together. The other changes in the File menu are operational changes which are explained below in the "Operational Changes" section.

The Edit menu contains two new items, 'Curly Quotes' and 'Straight Quotes'. The 'Curly Quotes' item changes old-fashioned straight quotes (" and ') into their "curly" equivalents (" , ' , and '). The 'Straight Quotes' item returns curly quotes to their straight equivalents. To use either of these items, select some text and select either 'Curly Quotes' or 'Straight Quotes' from the menu. (Note that no checking is done to ensure that the text you select actually contains any quotes.) While converting curly quotes to straight quotes is a straightforward task, changing straight quotes into curly quotes involves a few rules. For information about these rules, refer to Figure 4 of Joe Wankerl's "TeachText Translator v1.1" article in GS+ V2.N3.

The biggest change in the Style menu is that all of the font size selections have been split off into a separate menu. The new Size menu works exactly as you would expect.

To keep the Style menu from appearing too small I added three new style change selections to the menu: ALL CAPS, all lowercase and Title. To use these items, select some text and then pick the appropriate item from the Style menu. Selecting ALL CAPS will shift all of the selected letters to uppercase: JUST LIKE THIS. Selecting all lowercase will shift all of the selected letters to lowercase: just like this. Selecting Title will make the first letter of each word uppercase and all other letters in the word lowercase: Just Like This.

The Preferences menu, now known as the Prefs menu, contains one new item: Zoomed On Open. When you have this preference selected, the EGOed window will zoom to fill the entire screen *when you first open EGOed*. It does *not* affect the window size or position information in any TeachText files that you subsequently open.

Operational Changes

Even though it has not been documented before, you may have noticed that if your file had not been changed and you selected the Save item, EGOed did not save the file

again. This can save lots of time, especially if you are working with a large file and you are storing it on a 3.5-inch disk. In EGOed v1.33 this has been taken one step further. If the file you are editing does not need to be saved, the Save item is disabled. The overall effect is the same, but this way you know for sure that you do not need to save your file.

The biggest change in this version of EGOed is that it now remembers the folder you get your files from and the folder you save your files into. In addition to keeping track of these folders, EGOed also preserves the current folder that your application is working in. Each time you select Open or Save As from the EGOed menu, EGOed loads in the path to the folder you were using and substitutes it for the folder that the application is working with. If the folder that EGOed has saved is not available (i.e. it was on a floppy disk and you ejected the disk), you will be asked to insert the disk containing the folder. When this happens, simply insert the disk and click the OK button, or click on the Cancel button. When you finish with the Open or Save As dialog, EGOed saves the name of the folder you were in and restores the folder that the application was working with. In other words, your application will be right back where it expects to be.

In addition to all this, EGOed lets you "switch-hit" between your Open and Save As folders. To do this, simply hold down the *shift* key as you pick either the Open or Save As item. If you hold down shift while selecting Open, you will be placed in your Save As folder. If you hold down shift while picking Save As, you will be placed your Open folder. If you don't want to use either of these save folders (i.e. you want to use the same folder that the application is currently using) simply hold down the *option* key as you select either Open or Save As.

As I said earlier, EGOed saves the locations of the folders you open files from and save files to. If you don't want

What Is EGOed?

EGOed is a New Desk Accessory (NDA) text editor. When you install EGOed on your startup disk, you can edit and print ASCII text, TeachText, and AppleWorks Classic Word Processor files from inside any desktop program that properly supports NDAs. To use it, you must use IIGS System Software v5.0.2 or later. For information on installing EGOed, refer to "How to Use the GS+ Disk." For more information on using EGOed, read the file EGOed.Docs which is on your GS+ Disk in the EGOed folder.

Preservatives

The first thing to remember is that the current work folder is *always* preserved around Standard File calls!

Open...

When you select the Open menu item, a Standard File dialog will appear letting you select a file to open. The path to the file you select will be kept internally by the program. This path is called the *OpenPath*. The next time you select the Open item, standard file will display the files from the folder specified by *OpenPath*.

If you hold the option key down when you either Cancel or Open a file from the Standard File dialog, the *OpenPath* will not be replaced.

Save As...

When you select the Save As menu item or attempt to save an untitled document, a Standard File dialog will appear letting you specify the folder to save the file in. The path to the file you specify will be kept internally by EGOed. This path is called the *SavePath*. The next time you select the Save As item, standard file will display the files from the folder specified by *SavePath*.

If you hold the option key down when you either Cancel or Save a file from the Standard File dialog, the *SavePath* will not be replaced.

Switch Hitting

If you hold the shift key down when selecting the Open item, the *OpenPath* is

ignored and Standard File will display the files from the folder specified by *SavePath*.

If you hold the shift key down when selecting the Save As item, the *SavePath* is ignored and Standard File will display the files from the folder specified by *OpenPath*.

Using The Current Work Folder

If you hold down the option key while selecting either the Open or Save As item, the *OpenPath* and *SavePath* will be ignored and you will be placed in the current work folder that is being used by the host application. Remember to hold down the option key when exiting the Standard File dialog if you do not want your *OpenPath* or *SavePath* replaced!

EGOed to save that information, simply hold down the *option* key as you dismiss the Open or Save As dialog.

Confusing isn't it? Well, don't worry. All of this is summarized in the "Preservatives" sidebar. If it still confuses you, don't panic! Simply install the new EGOed and spend a few minutes playing around with the Open and Save As items. Once you *use* it, it will all make sense.

That's it for this version of EGOed. These new features have already made my life easier, I hope they do the same for you. If you are interested in some of the technical aspects underlying these changes, read on. If not, you can stop reading now. If you find a problem with this version of EGOed, remember to fill out the problem form on your *GS+* Disk and send it in! If you have a suggestion for future enhancements, send those in too!

Programming Considerations

Let's face it, after a year of enhancements and patches, the EGOed source code was becoming quite a mess. The conversion to ORCA/C helped, but that still left the REZ code with a curious resemblance to Don King's hair—very impressive looking, but still quite a mess. Not only was it tough to read, it was making it very

difficult to add to the program. It took almost 30 minutes just to add the 'Curly Quotes' and 'Straight Quotes' items!

So, I cranked up Genesys and used it to output a new *EGOed.rez* file and a set of equates for each of the items in that file. Now, Genesys does a pretty good job of generating code, but if you are not using Resource Names, the names you end up with are not very descriptive. So, I began to assign more meaningful names to each of the items.

About halfway through the process, it occurred to me that I would have to go to the *EGOed.h* file and make sure that the `#defines` I had in that file matched up with the `#defines` that were in the new *EGOed.rez* file. What a mess. So, what I did was split all of the `#defines` into a new file, *EGOedDef.h*, that would be used by both the C and REZ portions of EGOed.

The obvious problem with this is that C and REZ are two different source code languages. The solution to this is a simple one, create a make file that uses the `change` command to change the language type of *EGOedDef.h* to either CC or REZ depending on the language of the file you will be compiling next.

A less obvious problem is that the REZ compiler knows when to convert a value to a long and when to leave it as a word. In C, you have to do the conversion yourself. To automate this process, you have to be a bit sneaky. A fragment of this sneakiness is shown on the next page in Figure 1. Basically, what happens is this: when the ORCA/C compiler is "active," the symbol `_ORCAC_` is defined, so we define the symbol `cLong` to have the value `"(long)"`. When ORCA/C expands this value into the *EGOedDef.h* source code, the value of `AnEGOedConstant` becomes `"(long) 1"`, which is exactly what we want for the C portion of the program. If the symbol `_ORCAC_` is *not* defined, another compiler must be executing—in this case, the REZ compiler. So, we define `cLong` to simply insert a comment into the source code. Very simple and very effective.

Start 'Em Up

After many months of Joe telling me, "You need to take that `StartUpTools` call out of EGOed. NDAs are not supposed to use that you know." I finally gave in and took it out. Why shouldn't NDAs use `StartUpTools`? Well, the main reason is that `StartUpTools` automatically tries to

open the resource fork of the host application. While it never did EGOed any harm, I've been told that it's very dangerous to do that from an NDA. So, I changed EGOed to start up all of its tools the old fashioned way—one at a time. The routines that handle all of tool stuff are now in the source file **Toolsets.h**.

He's So Resourceful

After making that change, Joe told me that I should also take another look at the way EGOed was using resources. Basically, EGOed was doing a *lot* of unnecessary work in this area. You see, the Desk Manager *automatically* sets things up to allow your NDA to use its resources when your NDA becomes active. Of course, none of this is documented in the Toolbox Reference manuals! For the complete poop on NDAs and resources, you need to check out Apple IIGS TechNote #71: DA Tips and Techniques.

Nice NDA. BIG!

Last issue, I moaned and moaned about how big EGOed had gotten after the switch to ORCA/C. The main problem was that due to some bugs in ORCA/C

Figure 1

```
#ifndef _ORCAC_
#define cLong (long)
#else
#define cLong /* just a comment */
#endif

#define AnEGOedConstant cLong 1
```

v1.1, I simply could not optimize the thing. Well, I'm happy to say that ORCA/C v1.2 is now in beta-testing, and it fixes almost all of these optimization bugs. As a result of this, and some fine-tuning to the code, EGOed v1.33 is about 2K smaller than v1.32c—even with all of the new features.

Staying On The Path

In his OSLib article, Joe included a few routines that are used to work with GS/OS prefixes (**HandleToPrefix** and **PrefixToHandle**). As opposed to using those library routines, I decided to demonstrate the same thing in C. The routines for this were taken from Battery Brain v1.1 and are in the file **PFX.h**. I enhanced them to do a bit more error

checking, but they are basically the same as what you will find in OSLib.

"Quote Conversion"

Last issue, Joe added quote conversion to our TeachText Translator for GraphicWriter III; and this issue, EGOed does quote conversions! Coincidence? Nah, Joe just threw it in while I wasn't looking. Just like v1.1 of the TeachText Translator, EGOed uses the **ConvertQuotes** routine from Joe's **GWLib**.

That's All Folks!

Next time around, I hope to give you an even smaller EGOed and a new file format or two. If you have any problems, don't forget to send them in on your **GS+** problem form. **GS+**

Transfusion v1.1.1

By Josef W. Wankerl

This version of Transfusion is mostly comprised of bug fixes; however, there is one new feature that has been added. Prefixes are preserved around Standard File calls, and internal prefixes are kept in accordance with the new standard proposed in this issue (see "EGOed v1.33" for more information on this).

Them Darn Bugs

The only other enhancements left to talk about are the bug fixes. The most important fix was correcting the key equivalence for the Cancel button during a file transfer. Previously the **moreFlags** field was set incorrectly *and* the modifier key field was also set wrong. However, the transfer did indeed cancel when the Command-. (period) key was pressed! It also stopped whenever *any* Command-key sequence was pressed. This is now fixed.

Please Don't Pollute The C

There were two other bugs that are now fixed, but both of them were caused by ORCA/C v1.1. During my beta-testing of Transfusion, I was using beta versions of ORCA/C v1.1.

What Is Transfusion?

Transfusion is a New Desk Accessory (NDA) Terminal program. When you install Transfusion on your startup disk, you can use it and your *external* modem (Transfusion will not work with internal modems) to go online (and do file transfers) from inside any desktop program that properly supports NDAs. To use it, you must use IIGS System Software v5.0.2 or later. For information on installing Transfusion, refer to "How to Use the **GS+** Disk." For complete information on using Transfusion, read the file **Transfusion.Docs** on your **GS+** Disk in the **Transfusion** folder.

With those beta versions, my **DoInfo** function worked great. However, when I used the final ORCA/C v1.1, the function broke. The reason it broke was because the substitution array for the **AlertWindow** call was off by one in each case (each pointer pointed to the second character of the substitution string instead of the first.) To fix this, I assign the values to each entry *manually* inside the **DoInfo** procedure.

The second ORCA/C v1.1 bug that I circumvented was one that many users complained about: the error \$53 (Parameter out of range) when the text buffer was saved. This is the same bug that is present in Battery Brain v1.0. To correct this bug, I simply turn optimizations off around the **DoSave** function. I have told the Byte Works about both of these bugs and they should be fixed in ORCA/C v1.2. **GS+**

Working With The Toolbox

By Josef W. Wankerl
Part 1: The Tool Locator

With this issue, we begin a new series on programming the IIGS. This series of articles is for anyone who already has the basics of programming down and wants to take advantage of Apple's powerful set of Toolbox routines. I will, eventually, provide code examples of how to effectively use the Toolbox. It's easiest for me to provide examples in either assembly language or C, but I could just as well give them in Pascal. If you have a preference, let me know about it, and I will set my default example language from your responses. This is *not* going to be a tutorial for any particular language, so pick a language that you already know! You have a bit of time to decide, though, because I first have to explain a bit of basic terminology and present some of the underlying mechanics of the Toolbox.

Just What Is the Toolbox?

Making the Apple IIGS do exactly what you want it to is no easy task, no matter which language you choose to program in. Luckily, Apple has built into the IIGS a powerful set of pre-programmed routines that take care of many of the chores that are required to make a IIGS program come to life. Each of these individual routines is called a *tool*. Related tools are divided into groups called *tool sets*. Taken all together, these tool sets are known as the Apple IIGS *Toolbox*, and they are intended to make the programmer's life much easier. That's right, I said *easier*. Believe it or not, next to pizza (of course), *the Toolbox is your very best friend!* In these articles, I will attempt to describe how to use the Toolbox routines in your programs, not just list what each routine does—for that, you need the *Apple IIGS Toolbox Reference*, volumes I, II, and III. (So hurry up and get hold of them if you don't already have them!) I can't stress the point enough that you need *Apple's* Toolbox references, not some third-party Toolbox manual. These articles are meant to *enhance* Apple's reference books, not replace them.

Why Use It?

The reason Apple Computer, Inc. sites for using the Toolbox is the consistent user interface it provides. If you write your program using the facilities of the Toolbox, you already have all the components (windows, scroll bars, etc.) of a tried and true, easy-to-use, Graphical User Interface (GUI—pronounced “goeey”). All you have to do is figure out the best way to put those components together for your particular application.

Another advantage of the Toolbox is that it provides a flexible way to promote *reusable code*. The concept of reusable code is that you have a collection of routines that you commonly use, and you use them in your projects instead of rewriting something similar from scratch. Reusable code has many benefits, some of which are: the time it takes to get a product working is less because you don't have to write as much code, the reliability of your program goes up as the reusable modules have been tested over and over and will be more bug-free than new and untested code, and you can take advantage of experts who write the code for you. There are many other benefits to reusable code, but this isn't the place to discuss them—pick up a software engineering text for that kind of material.

If you use the Toolbox, and Apple enhances it, your program is enhanced also! Consider the Standard File tool set. When Apple released System Software v5.0.3, they added several nifty features to Standard File that made it easier to use. Any program that correctly used the Standard File tools *automatically* got all of these new features! No modifications necessary.

The Toolbox lets you take advantage of many of the advanced features of the Apple IIGS without having to know all of the nasty details of how everything works. For example, you don't need to know how the Ensoniq Digital Oscillator Chip (DOC) works in order to play a digitized sound sample—you just load the sample into

memory and make a Toolbox call. The tools take care of all the dirty work. You don't have to know how the hardware maps memory to the screen—you just issue a Toolbox call to draw boxes, circles, and pictures for you. This lets you concentrate on your program details instead of getting sidetracked on housekeeping chores.

Another advantage of using the Toolbox is that if you use it *correctly*, your program should run without modification on all current and future models of the IIGS. When the ROM 03 IIGS came out, a lot of programs “broke” because they did not use the Toolbox properly.

Many programmers don't like to use the Toolbox because it is “too slow” and they don't think that the code that Apple's programmers write is better than their own. In certain situations, the Toolbox *really is* too slow, such as for animation. But for all other tasks, the Toolbox is more than adequate. I have met many of the people who write the Toolbox and IIGS System Software, and they are some of the best programmers I've ever met. I trust their code and you can too.

Common Ground

There are a few routines that are common to every tool set. They are the *Boot*, *Reset*, *Version*, *StartUp*, *ShutDown*, and *Status* routines. The only routines that you should be worried about right now are the *StartUp* and *ShutDown* routines. The *Boot* and *Reset* routines are handled for you automatically, and you should *never* make those calls yourself. (Boot routines are called when the IIGS is first started up. They put the computer in a state where calls to the other tools in a tool set can be made. Reset routines are called whenever the system is reset.) The *Version* and *Status* routines are purely informational and almost never need to be called, unless you are curious.

In order to use a tool set, it must be started. You do this with the *StartUp* call. Once the tool set is started, you can make calls to the other tools in the tool set. The

ShutDown call tells the computer that no more calls will be made to the tools in that tool set. The Status call returns a value (True or False—where False is a zero value and True is a nonzero value) to let you know whether the tool set is started or not. The Version routine returns back the version number of the tool set.

The final word for all Toolbox information is the *Apple IIGS Toolbox Reference Manual*, volumes I, II, and III, published by Addison-Wesley. If you intend to do Toolbox programming of any kind, you **NEED** these manuals. (Did I say that already? Good.) The TechNotes published by Apple are also great reference materials. They contain tips and techniques for programming the Apple IIGS as well as corrections for the Toolbox Reference Manuals. Apple Developer and Technical Support produces sample source code packages as well. I find that learning by example is the best way to understand what's going on, so I really love the sample source code. You can get the Toolbox Reference Manuals at almost any bookstore (if they don't have them in stock, they can order them for you), and you can get the sample source code and the TechNotes either from APDA or from almost any online service, such as America Online.

Finding Yourself

There are many modules that compose the Toolbox system. Each module is called a *tool set*. The tool set that ties everything together is called the *Tool Locator*. The Tool Locator is responsible for loading in new tool sets and providing an interface between your program and the Toolbox. There are some more advanced features of the Tool Locator, but its main job is to keep track of all the tool sets.

The main interface between the Tool Locator and your program is a vector at \$E10000. The way to make a Toolbox call is to put the number of the Toolbox routine you want to call in the X register, and then JSL to the \$E10000 vector. The vector will then call the Tool Locator, which will interpret the X register value and call the appropriate tool set routine. If you program in assembly language, there are macro files which let you reference

Toolbox calls by name rather than by number. The macro will automatically load the X register and make the JSL for you. Using macros makes Toolbox programming infinitely easier. Of course if you work with a high-level language (C, Pascal, BASIC), you won't have to worry about any of this—Toolbox calls will be translated for you by the compiler.

The Tool Locator Routines

The Tool Locator's main job is to provide a way to load tool sets from disk and install them. There are two types of tool sets: *system* tool sets and *user* tool sets. System tool sets are written by Apple and can be used by any application. User tool sets are written by individuals and are used by applications that need the extra power. The system tool sets are guaranteed to be available to everyone who has the latest system disk. User tool sets are guaranteed to be available to only those people who have them. In other words, when you are writing a program, you can be sure that people will have system tool sets, but you cannot be certain that a user tool set will be available, unless it is contained within your program and your program installs it every time it is run.

In order not to complicate matters, I'll concentrate on system tool sets. User tool sets are rare. I'll also concentrate on the latest versions of the tool sets, as they are on System Disk v5.0.4. Through these Toolbox articles, I hope to bring forth a new generation of Apple IIGS programmers—programmers who aren't afraid to use the Toolbox and, more importantly, who use the Toolbox *correctly*.

How To Use the Tool Locator

The correct procedure to follow when starting up a tool is to first load it from the System Disk, then call the StartUp routine. There are two ways to load a tool from disk: LoadOneTool and LoadTools. The LoadOneTool call loads in one tool at a time, while the LoadTools call loads in a bunch of tools. Usually an application will issue the LoadTools call to load in all the tools it needs at the start of the program. The LoadOneTool call is used mostly by desk accessories. The desk accessory

first checks the status of each tool it needs (with the Status call), then issues the LoadOneTool call for each tool that it needs and is not already started. After the tool is loaded, a StartUp call is then issued. When the program is done with the tool, a ShutDown call is issued for each tool that was started. If the program that is shutting down is a desk accessory, it is proper to issue the UnloadOneTool call for each tool that was loaded. This leaves the computer in the same state that it was in before the desk accessory was started. An application, however, does not need to unload the tools when it shuts down.

An application has one additional way to start up and shut down tools: the StartUpTools and ShutDownTools calls. The StartUpTools call takes a list of tools to load and start, similar to the LoadTools call, except that this call loads the tools and then actually starts them. Using this call is a boon in many ways. First, it hides all the StartUp calls, which can get rather complex from the parameters needed and direct page memory allocations. It also ensures that the tools will be started correctly—if something goes wrong with your program, you can be sure that it isn't because your tools were started incorrectly. The ShutDownTools call shuts down the tools that were started with the StartUpTools call. Notice that the StartUpTools call is only available for applications. Desk accessories should not use the call because of some dire consequences that are out of the scope of this article.

Is That All?

Believe it or not, it is! In these few paragraphs, we have gone over what the Toolbox is, why you should use it, and how to use the most fundamental of all the tool sets—the Tool Locator!

For the next article in this series, I'll need your help. Write in and tell me which language you would like to see the examples given in and which tool sets you are having the most trouble with. Until next time, remember, the Toolbox is your friend!

GS+

[Editor's Note: OS Library is not a stand-alone program! It is a programming tool that we feel our advanced readers will find very useful. It is intended for those doing advanced IIGS programming.]

OS Library is a collection of routines designed to make interacting with some commonly used GS/OS and Standard File calls easier. OS Library (OSLib) can be used from any language that supports linking to standard libraries, just make sure that when you link, the OSLib file is in your 2/ directory. However, some additional work may be required to correctly access OSLib routines if the language you use can't call libraries that follow the Toolbox parameter passing protocol.

To install OSLib on your system, see "How to use the GS+ Disk" in this issue. For demonstrations of some of OSLib's functions, see the program "The New Order" in this issue.

Currently, OSLib is comprised of five procedures: *FillReplyRec*, *GetDirectory*, *PrefixToHandle*, *HandleToPrefix*, and *TraverseNames*. Each procedure is called using the Toolbox parameter passing protocol. First, result space is pushed on the stack. Next, input parameters are pushed on the stack. Then, the OSLib procedure is called. When the OSLib procedure returns, the calling program must pull any results from the stack.

The following paragraphs give a brief description of each OSLib procedure. The stack diagrams accompanying each OSLib routine are on the next page.

FillReplyRec

If you've ever wanted to use a Standard File multi-get routine, you know how difficult it can be to work with the multi-file reply record. This routine takes away all that pain. This procedure takes, as input, the multi-file reply *namesHandle*, a pointer to a new-style

reply record, and an entry number. Figure 1 displays the parameters for this call. *FillReplyRec* fills the new-style reply record with the entry (specified by the entry number) from the multi-file *namesHandle* structure. In addition to that, it returns a pointer to the entry within the multi-file *namesHandle* structure. Normally this result can be discarded because all the information from the entry is in the new-style reply record, but if you have the need to access the entry directly, the ability is there.

GetDirectory

This routine takes, as input, a pointer to a Class 1 *output* string containing the pathname of a directory. It returns a handle to a multi-file *namesHandle* structure containing the information for all the files in the directory. This handle can then be used as input to the *FillReplyRec* procedure. Figure 2 displays the parameters for *GetDirectory*. The reasons that a Class 1 output string is used instead of an input string are because it's simply easier on the programmer to work with only one kind of structure, and the Class 1 string is the most flexible because it contains a Class 1 input string. Only the input string fields are used within *GetDirectory*, the buffer size field of the output string is ignored.

PrefixToHandle

This routine takes, as input, a valid memory handle (locked or unlocked) and the number of a prefix. When it returns, the handle has been resized and *locked* and it contains a Class 1 output string with the prefix name in it. Figure 3 displays the parameters for this routine.

HandleToPrefix

This routine takes, as input, a handle to a Class 1 *output* string and the number of a prefix. When it returns, the prefix has been set to the name in the Class 1 *input* string field from within the passed output string. Here again, an output string is used as input to the procedure simply because it is more flexible. The buffer size field of the output string is ignored.

Figure 4 displays the parameters on the stack.

TraverseNames

The *TraverseNames* procedure is the most complex of all the OSLib routines. *TraverseNames* takes, as input, a word containing flags to control the operation of *TraverseNames*, a pointer to a routine of your own that will handle each name in a *namesHandle* structure, a pointer to a new-style reply record, and a handle to the aforementioned multi-file *namesHandle* structure you wish to traverse. Figure 5 displays these parameters on the stack.

TraverseNames then makes a call to *FillReplyRec* for each entry in the *namesHandle*, to copy that entry to the new-style reply record you passed. *TraverseNames* then calls your name handling routine. Your routine can then take action on whatever is in the new-style reply record.

TraverseNames calls your name handling routine with a JSL. When your routine is called, there is a word of result space on the stack, as shown in Figure 6. Your routine *must* set this result word to either a zero, one, or two before it exits. A zero value means that the entry was processed correctly. A value of one means that the entry was skipped. A value of two means that the entry was skipped and the traversal should stop.

The flags parameter controls the operation of the *TraverseNames* routine. Bits 0 and 1 of the flags parameter are used to tell *TraverseNames* how to handle *directory* entries. The combinations are:

00 = Skip Directories - your routine passed is not called for directory entries.

01 = Contents First - the contents of the directory are read with *GetDirectory* and the resulting *namesHandle* is traversed, then your routine is called with the directory name.

10 = Contents Last - your routine is called with the directory name, then the contents of the directory are read with GetDirectory and the resulting namesHandle is traversed.

11 = Directory Only - the directory entry is treated just as any other entry in the namesHandle. Any files that are in the directory are ignored.

Bit 2, called the SkipOnBad flag, is only valid when the contents of directories are being processed (bits 0 and 1 equal 01 or 10).

If the SkipOnBad flag is set, and the contents of directories are being processed *last*, and your routine returns a value of one to TraverseNames for a directory (which tells TraverseNames that your routine skipped the directory), the *contents* of the skipped directory will also be skipped.

If the SkipOnBad flag is set, and the contents of directories are being processed *first*, and your routine returns a value of one to TraverseNames for a file in the directory (which tells TraverseNames that your routine skipped the file), the directory entry will also be skipped after any remaining files in the directory have been processed.

If the SkipOnBad flag is clear and you skip either a directory or a file within a directory, nothing special happens.

Confused?

Granted, these routines may seem a bit confusing when you first read the descriptions! If you are confused about how these routines work, be sure to break out the source code for The New Order. It's on this issue's GS+ Disk and it contains working examples of almost all of these routines. With a bit of study, I think you will begin to see just how powerful and useful the OSLib routines are.

If you have any questions about OSLib, let me hear them. I especially want to hear any suggestions you might have for additions to OSLib. Putting all of these routines in one place has already made my IIGS programming easier, I hope it does the same for you.

GS+

Figure 1 - The Parameters Of The FillReplyRec Call

```
Stack Before Call
| previous contents |
| longspace        | WORD-Space for result
| ReplyRecPtr      | LONG-Pointer to the new-style reply record
| namesHandle      | LONG-Handle to the multi-file get namesHandle
| entryNum         | WORD-Entry number to copy to the reply record
|-----|<-- SP
```

```
Stack After Call
| previous contents |
| entryPointer     | LONG-Pointer the the entry in the namesHandle
|-----|<-- SP
```

Figure 2 - The Parameters Of The GetDirectory Call

```
Stack Before Call
| previous contents |
| longspace        | WORD-Space for result
| pathHandle       | LONG-Handle of C1OutputString with directory pathname
|-----|<-- SP
```

```
Stack After Call
| previous contents |
| namesHandle      | LONG-Handle to directory entries in namesHandle format
|-----|<-- SP
```

Figure 3 - The Parameters Of The PrefixToHandle Call

```
Stack Before Call
| previous contents |
| longspace        | WORD-Space for result
| pathHandle       | LONG-Handle to receive the C1OutputString
| prefixNum        | WORD-Number (0 to 31) of the prefix to get
|-----|<-- SP
```

```
Stack After Call
| previous contents |
|-----|<-- SP
```

Figure 4 - The Parameters Of The HandleToPrefix Call

```
Stack Before Call
| previous contents |
| pathHandle       | LONG-Handle of C1OutputString containing new prefix
| prefixNum        | WORD-Number (0 to 31) of the prefix to set
|-----|<-- SP
```

```
Stack After Call
| previous contents |
|-----|<-- SP
```

Figure 5 - The Parameters Of The TraverseNames Call

```
Stack Before Call
| previous contents |
| longspace        | WORD-Space for result
| TraverseFlags    | WORD-Flags controlling TraverseNames features
| RoutinePtr      | LONG-Pointer to routine to handle each entry
| ReplyRecPtr     | LONG-Pointer to the new-style reply record
| namesHandle     | LONG-Handle to the multi-file get namesHandle
|-----|<-- SP
```

```
Stack After Call
| previous contents |
|-----|<-- SP
```

Figure 6 - The Parameters For The Traversenames Entry Handler Routine

```
Stack Before Call
| previous contents |
| longspace        | WORD-Space for result
| ReturnAddress    | 3 Bytes-Address to RTL to
|-----|<-- SP
```

The HyperStuff Collection

First in this innovative series is ClipTunes™

ClipTunes™ is a collection of MidiSynth™ format songs. Included in the package is an Xcmd player for HyperStudio™ and HyperCard IIGS™. The "Synth" Xcmd allows the HyperCard™ and HyperStudio user to easily play 7 voice stereo music from within their own stacks. The songs were specially arranged for the Hypermedia environment and play in the background so you can open menus, click buttons, switch cards, animate Icons all while the music continues to play. Selections range from Holiday music, marches, songs for special occasions, and music to switch cards by. ClipTunes comes complete with tunes, sample stacks, Xcmd player, and MidiSynth tool (Tool 35). Suggested Retail Price\$39.95

Second in the HyperStuff Collection™ is ClipArt Plus™

ClipArt Plus contains 20 screens of the finest ClipArt available for the Apple IIGS. Categories include: Animals, trains, food, boats, construction, office, tools, and just about everything else including a kitchen sink. To compliment this fine ClipArt, included are 4 New Desk Accessories and as a bonus to HyperCard IIGS users, 4 Xcmds in a small demo stack.

Icon Button NDA makes creating HyperCard format Icons a breeze. Icon Button NDA lets you "Clip" out sections of any SHR screen and saves it right to the stack of your choice. GetPic Xcmd lets you display ClipArt Plus pictures, or pictures from most graphic formats including PrintShop IIGS™. Once displayed, selecting a portion of the picture for use in backgrounds, cards, or buttons becomes child's play. Simply draw a box around the image you want and click the mouse. Icon Mover Xcmd lets you copy Icons from stack to stack. With Vclip Xcmd you can view (or print) the contents of the Clipboard. Once you use ClipIt NDA you will wonder how you ever managed without it. ClipIt works great with any program that uses the standard Clipboard.

The complete ClipArt Plus package includes 20 screens of ClipArt, 4 New Desk Accessories, and 4 Xcmds for HyperCard IIGS. Suggested Retail Price ...\$39.95.

TM *GDL 2.0 Graphic Disk Labeler*

Create Colorful labels for your 3.5" disks. *GDL* imports popular graphic formats including SHR pictures, Icons, and PrintShop IIGS graphics. *GDL* uses Apple IIGS fonts, and includes provisions for changing styles & colors. Text can be superimposed over the graphics to achieve special effects. *GDL* includes Paint Tools for editing artwork and includes 320 and 640 mode versions. Complete with 50 labels and label grabber NDA. (1 MEG. required. for Imagewriter II.) Suggested Retail Price.....\$39.95

Also available.....*GDL* Graphic Disk with over 100 label files. Also included are Fonts, Icons, & PrintShop IIGS Graphics....\$9.50

.....and extra labels (250 continuous).....\$11.95 per package, plus \$2.50 shipping in U.S.

Available from most major software mail vendors or contact:
(516) 732-3771 (516) 360-0797
or America-On-Line Direct Connect Area

TRIAD VENTURE INC.
P.O.Box 12201
Hauppauge, New York 11788

PrintShop is a registered TradeMark of Broderbund Software.
Hyperstuff, Cliptunes, and ClipArt Plus and GDL are registered TradeMarks of Triad Venture Inc.
HyperStudio is a registered TradeMark of Roger Wagner Publishing.
MidiSynth, Hypercard IIGS, and APPLE IIGS are registered Trade Marks of Apple Computer, Inc.

GS+ User Group Connection

By Steven W. Disbrow

Beginning this issue, the *GS+* User Group Connection gets its own little column here in the magazine. In this column, we will discuss some of the more interesting things we have seen in the various user group newsletters that we receive here, and we will let you know which articles from *GS+* Magazine you can reprint in your user group newsletter. The objective here is to get a few more *GS+* subscribers (of course) and to help newsletter editors overcome the "deadbeat membership" blues. If you edit a newsletter, you know what I'm talking about. All we ask in return is that:

- Your user group must be involved in our newsletter exchange program.
- You do not change the content of the article in *any* way.
- You do not distribute the article any further.
- You reprint no more than one article per issue of your newsletter.
- You print the following disclaimer with each article you reprint:

"Reprinted with the permission of GS+ Magazine. This article is copyright 1991 by EGO Systems. For more information on GS+ Magazine, write to: GS+ Magazine, P. O. Box 15366, Chattanooga TN, 37415-0366."

How Do I Get It Typed In?

As far as I know no other magazine has tried anything like this before. So, like everything else here at *GS+*, we get to make it up as we go along! Basically, what we will do is maintain a disk (and, eventually, several disks) that contain the articles available to user groups. (In other words, you must get the articles from us. You should *not* type or scan them in.) To get the latest copy of this disk, simply send us a *blank, formatted, 3.5-inch* disk with a return envelope and return postage and we'll send you the latest disk of *GS+* User Group Connection files (in both plain ASCII and TeachText formats). We'll even type in all of the disclaimer stuff for you and throw in demo versions of some of our programs that you can put on your club's Disk Of The Month. What could be easier?

What's The Catch?

There is no catch (unless you consider having to pay for the postage and disk a catch). We want to help IIGS user groups help their members and, yes, we want to drum up some new business.

What's On The First Disk

To kick things off, we will make available the following articles:

Beginner's Guide To The Finder - Parts 1, 2, and 3
Beginner's Guide To System Disks - Parts 1, 2, and 3
Trash Can Award: *InnerExpress*

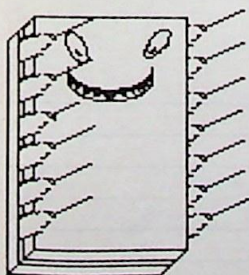
We will also include on the disk the EGOed Demo, the Transfusion Demo and a few other goodies. If there is anything else you want to see included on future disks, or if you have a suggestion for improvements in the *GS+* User Group Connection, let us know!

We Want You!

If your IIGS user group or IIGS special interest group (SIG) is not a member of the *GS+* User Group Connection, we want to change that. To become a member, simply have your president contact us. All we need is the name and address of your group. However, if you give us a free subscription to your group's newsletter, we'll give your group a free magazine-only subscription to *GS+* and access to the *GS+* User Group Connection reprints! Send that information and/or newsletter subscription to:

GS+ User Group Connection
P. O. Box 15366
Chattanooga, TN 37415-0366

Free Memory!



Free Memory is a memory management system that, by reorganizing and compressing blocks of memory, can increase your free memory by up to 20%. This extra memory could allow you to have longer documents in RAM. No matter what you use it for, it will be a valuable addition to your software library. Checks and money orders only. Sorry, no credit cards!

Free Memory™ © 1990-1 by GDCS.

Mail \$15.00 to:

GDCS
12810 Gaffney Road
Colesville, MD
20904-3517

**Golden
Delicious
Computer
Services**

Product Updates

Zippin' Along

Richard Stivers, President of Zip Technology, called us up the other day to let us know three things. First, he really liked Brian Winn's review of the ZipGSX (published in *GS+* V2.N3). Second, the problem that Brian found when using his ZipGSX with ProTerm had been corrected, and all new ZipGSX cards have the fix. Third, an upgrade to 10MHz is now available for the ZipGSX. The cost is \$175 and you have to send your ZipGSX back to Zip Technology for a few days. For more information on this upgrade offer, contact:

Zip Technology
5601 West Slauson Ave, Suite 190
Culver City, CA 90230
(800) 955-5520

Pak It In

Simple Software Systems International, Inc. (SSSI) has just announced Deskpak version 2.0. Although the press release they sent us does not say exactly *what* NDAs are included, the screen shots seem to indicate that the new Deskpak includes an NDA Word Processor, a calculator, a data base, a *macro recorder*, a file utility NDA, a "Super-clipboard," an appointment calendar, and an NDA that looks a bit like Suitcase for the Macintosh! The price? Well, it's either \$59.95 or \$89.95—there are two different prices on the two different sheets they sent us. To get the right price, contact SSSI at:

Simple Software Systems International, Inc.
4612 North Landing Drive NE
Marietta, GA 30066
(404) 928-4388

It's All In The Cards

In his review of Dragon Wars GS (*GS+* V2.N3) Dave Adams mentioned that Dragon Wars GS did not include a IIGS-specific reference card. According to an Interplay spokesman, the first several dozen copies of Dragon Wars GS accidentally shipped with the IIe/IIc reference cards. The problem has been corrected. If you got a IIe/IIc reference card with your copy of Dragon Wars GS, send it to Interplay (along with a short note explaining the situation) and they will send you a IIGS reference card. Send those cards to:

Interplay Productions
3710 S. Susan, #100
Santa Ana, CA 92704

GS+

GS+ Classifieds

1/2 PRICE SOFTWARE

Hostage - \$20
The Hunt For Red October
(novel not included) - \$ 20
Silent Service - \$20
Sub Battle - \$5
Questmaster - \$20
Neuromancer - \$20

Contact: *GS+* Magazine
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366
(615) 870-4960

Readers can place an ad in the **GS+** Classifieds for only \$5. This cost buys 25 words in one issue of **GS+**. Additional words are just 10 cents each. The **GS+** Classifieds are the perfect way to contact all of the other IIGS owners out there. The deadline for inclusion of a classified ad in the next issue (Volume 2, Number 5) of **GS+** is May 1, 1991. Simply fill out a photocopy of the coupon below; or send your ad along with your name, address, phone number, number of issues to run, and payment (**made payable to EGO Systems**) to us here at **GS+**; or call us at (615) 870-4960, Monday through Friday between 9 a.m. and 6 p.m. EST, to place an ad with your MasterCard or VISA.

GS+ Classified Ad Order Form

Ad copy: _____

Number of issues to run: _____ Number of words: _____ Total enclosed:\$ _____
Name: _____ Phone: _____
Address: _____
City: _____ State: _____ Zip: _____

Advertisers Index

By Steven W. Disbrow

Golden Delicious Computer Services - Page 19

Free Memory? Really? Well, yes and no. It's not free memory *chips*—it's a Classic Desk Accessory that compacts the memory in your IIGS. Ahem. That's all it does. \$15? This isn't a *bad* piece of software—it seems to be quite well done in fact—it just does not do \$15 worth of stuff. Buy this only if you can't scrounge up one of the dozen or so public domain or shareware utilities that do the same thing.

Gravenstein Apple IIGS Users Group - Page 44

What's this? An ad for a *user group*? Sure, why not? Even if you don't live in the greater Petaluma area, I highly recommend that you drop Russ Messana a line to see about getting a copy of their newsletter. It's a very slick (and informative!) piece of work.

TMS Peripherals - Back Cover

If you need a hard drive, I *highly* recommend that you buy it from TMS. Several months ago, I desperately needed a 45 MB removable to do backups with. TMS got it too me in less than 24 hours and I paid about \$100 less (shipping included) than the "base sticker price" of just about any of the big name 45 MB removable drives advertised in the MacRags. Since then, the drive has not given me a single problem. Even if the drive *had* been bad, TMS has a toll-free

support line, a 30-day Money Back Guarantee and a 2-year Warranty!

Part of our job here at *GS+* is to find the best people in the business and point them out to our readers. I was so impressed with TMS's products and service that I practically *begged* them to advertise with us. These guys are some of the best and they truly deserve your business.

ToolBox - Pages 46, 47

Imagine my surprise when the FedEx man brought this package! Inside was the ad you see on pages 46 and 47 and review copies of *Photonix II*, *Bouncing Bluster II* and *Space Shark*! Not only that, but the ToolBox people also publish "The Second Apple IIGS-Specific Magazine + Disk" in France! It's called "ToolBox Mag" and they sent us a complete set of back issues (3 issues). Even though I don't read French (the articles are in French, the programs are in French and English [the program keywords are in English and the variable names are mostly French]) it looks like there is a lot of neat stuff inside! If we can find a translator, we'll have more info next issue.

As for the programs they sent: *Photonix II* is the commercial descendant of the shareware disk-copy utility *Photonix*. Unlike the original, *Photonix II* can be installed on a hard disk and can be run from the Finder. Unfortunately, the only way to quit is to reboot. If you can live

with that, and you thought the original *Photonix* was the cat's pajamas, you should get *Photonix II*.

Space Shark is billed as "the fastest IIGS game ever." It's loosely based on the arcade classic "Defender" and it *is* dang fast. It's so fast I could not last more than about 3 minutes.

Bouncing Bluster II is the commercial descendant of the shareware game, *Bouncing Bluster*. *Bouncing Bluster II* is an Arkanoid-like game complete with a construction set for building your own levels. Unlike the original, *Bouncing Bluster II* works with System Software v5.0.2 and later and can supposedly be run from a hard disk (I could not get it to work from mine).

Look for full reviews of these products in upcoming issues of *GS+* Magazine.

Triad Venture - Page 18

Of the three products shown in this ad, we have only reviewed *Graphic Disk Labeler (GDL)*. This review appeared in *GS+* V1.N3. The reviewer, Wayne Packard found the program to be quite good and recommended it. In the year that has passed, GDL has been updated to version 2.0 and Triad Venture has shown their firm commitment to the Apple IIGS. The other products featured in the ad have just been released and we are in the process of reviewing them. **GS+**

MOVING?

Well, don't forget to tell us!

Simply remove your mailing label from a previous issue of *GS+*, affix it to a change of address form (available at your local post office), fill in your new address, and send it to us at:

GS+ Subscription Services
P.O. Box 15366
Chattanooga, TN 37415-0366

GS+ Back Issue Information

September-October 1989 (Volume 1, Number 1)
\$4.50 magazine \$5.50 disk \$6.50 magazine + disk

- System Software 5.0 Compatibility Chart
- NoDOS - A file utility New Desk Accessory complete with ORCA/C source code on disk
- Graphics Galore - Drawing "how-to" with 3 pictures on disk
- Reviews of Arkanoid II (new custom levels on disk), Crystal Quest, ORCA/C, Rocket Ranger, Silpheed, Test Drive II, TransWarp GS, Turbo Mouse ADB
- PLUS: Graphics, rumors, and the most over-hyped product of the year!

November-December 1989 (Volume 1, Number 2)
\$6.50 disk only (we are SOLD OUT of the magazine!)

- EGOed - An NDA text editor (TML Pascal II source code on disk)
- Brush with Greatness - Tips on drawing faces (pictures on disk)
- PLUS: Original icons and new levels for Laser Force on disk

January-February 1990 (Volume 1, Number 3)
\$6.50 disk only (we are SOLD OUT of the magazine!)

- Rotator - A beginner's desktop programming tutorial and program w/source code written in ORCA/C
- Winning Arkanoid II Levels
- Brush with Greatness - Space graphics (pictures on disk)
- HyperStudio stack version of GS+ V1.N2 on disk.

March-April 1990 (Volume 1, Number 4)
\$6.50 disk only (we are SOLD OUT of the magazine!)

- All About Control Panel Devices - with Desk Color CDev and ORCA/C source code on disk
- Random IIGS Programming Notes - An EGOed update
- Brush with Greatness - Architecture on your IIGS with pictures of the CitiCorp building and Frank Lloyd Wright's house on disk

May-June 1990 (Volume 1, Number 5)
\$4.50 magazine \$6.50 disk \$9.50 magazine + disk

- AppleFest Report
- Beginner's Guide to System Disks - Part 1
- GS/OS prefixes - PreFixer CDev and ORCA/Pascal source code on disk
- Brush with Greatness - How your IIGS makes colors
- Reviews of CMS SDRM 45 Megabyte Removable Hard Drive, S&S-RAMCard, DataLink Express modem, Visionary GS digitizer, GraphicWriter III, ZapLink, McGee, Math Blaster Plus IIGS, The New Talking Stickybear Alphabet, a sneak peek at the Zip GS

July-August 1990 (Volume 1, Number 6)
\$4.50 magazine \$6.50 disk \$9.50 magazine + disk

- KansasFest Report
- Beginner's Guide to System Disks - Part 2
- Transfusion - An NDA terminal program
- Reviews of AMR AS800K 3.5-inch drive, Salvation: The Exorciser, Disk Access, MD-BASIC, Katie's Farm, Task Force, BLOCKOUT, OMEGA, 2088: The Cryllan Mission, Hunt for Red October, Revolution '76, Where in the U.S.A. is Carmen Sandiego?

September-October 1990 (Volume 2, Number 1)
\$4.50 magazine \$6.50 disk \$9.50 magazine + disk

- Brush With Greatness - making the most of your digitizer
- Interview with Brian Greenstone (programmer of Xenocide)
- PING - video table tennis program (Merlin assembly)
- Shuffle - an Init file that allows you to move desktop windows from the foreground to the background (ORCA/M)
- Battery Brain - CDev saves BRAM parameters to disk (ORCA/C)
- Reviews of GS Sauce memory card, Salvation: Wings, World GeoGraph, Orange Cherry Talking Schoolhouse series, QIX, Solitaire Royale, InnerExpress

November-December 1990 (Volume 2, Number 2)
\$4.50 magazine \$6.50 disk \$9.50 magazine + disk

- Interview with Bill Heineman (programmer of Dragon Wars)
- Beginner's Guide to System Disks - Part 3
- LaserWriting - a guide to using an Apple LaserWriter with the IIGS
- Christmas Buyer's Guide
- TeachText Translator - import and export TeachText files in GWIII
- Reviews of Quickie Hand Scanner, AE 3.5" Disk Drive, Salvation: Renaissance, USA GeoGraph, Rastan, Captain Blood, HOSTAGE, Questmaster, Pipe Dream, The Immortal, PIRATES!

January-February 1991 (Volume 2, Number 3)
\$4.50 magazine \$6.50 disk \$9.50 magazine + disk

- AppleFest/Long Beach '90 and the Apple II Achievement Awards
- Interview with Jim Carson of Vitesse, Inc.
- Introduction to System Software v5.0.4
- RAM Namer - a CDEV that can rename your RAM disk at boot time, with ORCA/C source code on disk
- GS+ program updates - Battery Brain v1.1, EGOed v1.32c (now written in ORCA/C), GWIII TeachText Translator v1.1
- Reviews of ZipGSX, LightningScan, Design Your Own Home, Print Shop Companion IIGS, *Your IIGS Guide*, Dragon Wars, 2088: The Cryllan Mission - Second Scenario, Space Ace, Sinbad and the Throne of the Falcon

All prices include \$1.50 postage and handling (orders will be sent First-Class to the U.S., Air Mail to Canada and Mexico, and surface to all other countries. For Air Mail to all other countries, add \$5 per issue). Tennessee residents add 7.25% sales tax. Mail back issue requests to: GS+ Back Issues, c/o EGO Systems, PO BOX 15366, Chattanooga, TN, 37415-0366; or call (615) 870-4960, Monday through Friday between 9 a.m. and 6 p.m. EST, to verify availability. Please include your phone number on all orders placed by mail (in case we are sold out of an issue)! For MasterCard or VISA orders placed by mail, also include your card number, expiration date, and signature.

Rumors, Wishes & Blatant Lies

By Prof. G. S. Gumby

CinemaWhere?

Well, it had been rumored for months, but now it's a fact: Cinemaware is out of business. What happened? Well, Electronic Arts was supposed to buy the financially troubled company, but at the last minute (and we mean *last* minute) the deal fell through. Rumor has it that the night Cinemaware closed its doors, several ex-employees of the company got together at a local bar and discussed plans for a new company that would produce even better software and not suffer from being "run by pinheads." They also discussed plans for the waitress on duty and were ejected from the bar.

The New Finder

As our fearless leader mentioned in his editorial, there have been several signs that a new Finder will soon be released. What are these signs? Well, for one thing, Apple has recently defined formats for Version Resources and Comment Resources. Taken together, this would indicate that the new Finder will allow us to actually store info in those little Get Info windows—just like the Macintosh Finder. Another clue about the new Finder came from our own Big Dave Adams. He wrote to a member of the Finder team about the possibility of changing the Finder so that if an NDA, CDA, Init, or CDev was inactive, it would display a special "inactive" icon. The Apple rep wrote back saying he liked the idea, but it was too late to get it into the next Finder.

A New Name For The Magazine

Personally, I like "Dr. Gumby's Journal." But our idiot publisher won't let employees enter the contest. Jerk.

VGA Anyone?

Remember the Turbo-REZ board that was supposed to revolutionize IIGS video? Well, it's been cancelled. However, a completely different group of people are supposedly working on a new board that would allow you connect any standard VGA monitor to your IIGS. The main goal of the project is to "give IIGS users

access to low-cost *square* pixels." Amen to that!

Were You Enticed?

Last issue's rumor about the "Apple Enticer" certainly stirred up a lot of inquiries. The sad truth is that it was simply the fevered imagining of Brian M. Winn. (He's much better now thank you.) However, you might like to know that there *will* be a home video game system that uses the same 65816 CPU that the IIGS uses. It's currently available in Japan where it goes by the name "Super-Famicom." The maker of this product is a little company called Nintendo. We might just see a new interest in programming the 65816. Now, what would be a good machine to learn on . . . ?

Truly Outrageous!

Has there ever been a more annoying word than "Synergy?" Have you ever wondered where Apple came up with that word? After an exhaustive search, we think we may have found the answer: Jem. You know, the syndicated cartoon about the girl with the alien earring (Holy Gasseé Batman@!) that allows her to change her appearance. The earring's name is "Synergy." Seriously! Could we make this kind of stuff up?

Of course, we would never have suspected that so many of the policymakers at Apple watch cartoons if we had not caught word of Apple's soon-to-be-launched "Coo-Coo for Macintosh!" ad campaign. Apparently, they wander around the halls screaming, "Coo-Coo for Macintosh! I'm Coo-Coo for Macintosh!" Scary stuff!

McGee Goes Nuts

"McGee," "McGee Visits Katie's Farm," "McGee at the Fun Fair" . . . where will the computing industry's favorite yard-ape turn up next? Well, sources close to McGee (he is a real kid you know) say that the next few games in the McGee line will reflect the fact that McGee is a bit older than when he first started all this. Potential titles include, "McGee at the Free Clinic," "McGee Visits the Mustang

Ranch," and "McGee at the Robert Maplethorpe Exhibit." McGee's dad—who is also his manager—was quoted as saying, "I do all the research for these programs, so I'm really looking forward to this new line of games!"

The Next Thing You C

In the past few issues of *GS+*, we have made vague references to bugs in ORCA/C v1.1. None are very serious, but when added together they can make programming with ORCA/C difficult. Fortunately, The Byte Works is still one of the most responsive companies around. ORCA/C v1.2 is currently undergoing beta testing and should be available by the time you get this magazine. Even as I write this, we have received our update to ORCA/Pascal (v1.3) and the Prizm desktop development environment (v1.1). According to Patty Westerfield of The Byte Works, a general mailing will go out before 3/15/91 containing more information about the updates. Why didn't we put this info in "Product Updates?" Well, when I started typing, it was all still a big rumor!

Well Done!

Officially, *GS+* Magazine's position on the Gulf War is one of neutrality. However, a great many of our friends and subscribers have been serving in the Gulf, and it would be silly to ignore the situation. So, we just wanted to say, "**WELL DONE!**" In the time it takes us to get out a single magazine, you boys and girls won a war! (Of course, there are 500,000 of you and only 4 of us.) You did an absolutely amazing job. Come home soon.

You Can Do Better?

Sure you can! Send those rumors, wishes and blatant lies to:

America Online: GSPlusDiz

US Mail:

GS+ Rumors

P.O. Box 15366

Chattanooga, TN 37415-0366

GS+

How To Use The GS+ Disk

Back It Up!

The first thing you need to do is **MAKE A BACKUP COPY OF YOUR GS+ DISK WITH THE FINDER!!!** Next, put the original in a safe place. If you are having a problem making a backup copy, give us a call at (615) 870-4960. If your disk is damaged, let us know and we'll get a new one to you as soon as possible.

Installing The Software

To install the software on this issue's GS+ Disk, start up your computer using System Software v5.0.2 or later (preferably v5.0.4), and then place your *backup copy* of the GS+ Disk in a drive. (You *did* make a backup didn't you?) Now, run the Installer program that is on your GS+ Disk. (From the Finder, you would double-click on the Installer icon.) When the Installer window appears, select the update you want to install from the left-hand window, and the disk you want to install it on in the right-hand window. Then click on the Install button. For more information on how to use the Installer, refer to your IIGS owner's manual.

For More Help . . .

Remember, each folder on the GS+ Disk contains its own *a.Read.Me* file that describes the contents of the folder in detail.

What's On The Disk?

There are 12 items in the root directory of this issue's disk. They are:

a.Read.Me

A lot can happen from the time we send this magazine to the printer and the time we get ready to mail them out. If anything does happen, we will put everything we can find out about it in this file. This is a plain text file.

Allison.Rev

This folder contains the two digitized pictures that Greg Zimmerman mentioned in his review of the Allison digitizing software. These are uncompressed pictures. Use any paint program to view them.

EGOed

This folder contains EGOed v1.33. This folder also contains complete documentation for EGOed v1.33 in the file *EGOed.Docs*. EGOed must be installed on a startup disk.

HOM

In this folder is a new scenario for Halls of Montezuma (see review this issue) designed by our own Big Dave Adams.

Icons

This folder contains the Finder icons discussed in the "Icons" article.

Installer

This is the Apple IIGS Installer. Run it to install the other programs on this issue's disk. For more information on the Installer, refer to your IIGS owner's manual.

OSLib

This folder contains the OS Library described in the "OS Library" article. OS Library should be installed in your *2/Libraries* folder.

Problem.Form

This is the standard GS+ bug report form. If you have a problem with one of our programs, fill out this form and send it to us. This is a TeachText file. You may use EGOed to view it.

QuickNDA

This folder contains the Quick NDA init described in the "Quick NDA" article. Quick NDA must be installed on a startup disk.

Scripts

This folder contains all the scripts that are used by the Installer in order to automate the installation process.

TheNewOrder

This folder contains The New Order NDA as described in the "The New Order" article. The New Order must be installed on a startup disk.

Transfusion

This folder contains the Transfusion NDA version 1.1.1 as described in the "Transfusion v1.1.1" article. This folder also contains complete documentation for Transfusion v1.1.1 in the file *Transfusion.Docs*. Transfusion must be installed on a startup disk.

Writers.Guide

This is a TeachText file that explains what you need to do in order to write reviews, articles, programs, etc. for GS+. You may use EGOed to view it.

Please Remember . . .

The contents of the GS+ Disk are *not* public domain or shareware! We depend on *your* honesty to stay in business. Please do not give away copies of the GS+ Disk or any of the files on it. If you do, we will not be able to stay in business. It really is that simple! **GS+**

This issue, we have several nifty Finder icons that were sent to us by Keith Passmore of Phoenix, AZ. Keith used D!cEd v1.2 to create these icons. All of these icons are in the **Icons** folder of your **GS+** Disk. To use them, simply copy the appropriate icon file off of your **GS+** Disk into the **Icons** folder of your startup disk or the disk that you run these programs from.

L.Ninja.Icon

This is a really sharp icon for the old game "The Last Ninja."

Mancala.Icon

An icon for the game "Mancala."

Life.Icon

An icon that you can use with any of the dozens of "Life" games that are available for the IIGS.

GS.Plus.Icons

This file contains icons that are for items that are on each **GS+** disk. There are icons for EGOed, the **GS+** disk, TeachText files, and a special folder icon. To have these icons available at all times, copy this file into the **Icons** folder of your boot volume. If you don't like the yellow folder icon contained in this file, delete the **GS.Plus.Icons** file and it will go away.

Installer.Icon

This is an icon for Apple's Installer program. It was taken from the **Finder.Icons.X** file that comes on the IIGS System Disk.

XFusion.Icon

This is an icon for the Transfusion NDA.

That's it for this issue. Remember: we need your help! If you have any Finder icons that you would like to share with the rest of our readers, please send them on in! **GS+**

IMPORTANT!

**Use scissors or a knife to open disk bag!
Do not attempt to pull bag away from magazine!**

DISKLESS?

If you did not receive the disk with this magazine and have decided you would like to have it, just send a check or money order for \$6.50 to:

**GS+ V2N4 Disk Offer
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366**

Or call us at (615) 870-4960, Monday through Friday between 9 a.m. and 6 p.m. EST, to bill it to your MasterCard or VISA.

Tennessee residents add 7.25% sales tax.
Prices include \$1.50 for First-Class delivery to the U.S., air mail to Canada and Mexico, or surface to all other countries. Add an extra \$3.50 (\$10 total) for air mail to all other foreign countries.

The Molehill

By Josef W. Wankerl
continues on the next page . . .

Due to the space constraints on the last *GS+ Magazine*, *The Molehill* was, unfortunately, left out. This was, perhaps, good in the sense that I can now bring you even more up to date with Nifty List! When the last *GS+* came out, Nifty List was at version 3.0, but now it's at v3.2 (v3.1 was never released.) Included in the v3.2 package is a freeware external command module that I wrote and gave Dave Lyons the permission to include with his newest Nifty List release, so hurry up and get it! But enough of my yakking, let's rock 'n' roll.

Writing programs has always been an exciting adventure for me. Tracking down bugs is fun personified. However, there is always the occasional bug that just can't be found by normal means. When that happens, what do you do? Who do you call? No, not anything with a buster suffix, you silly human. You call on Nifty List and GSBug!

Nifty List

Nifty List is, perhaps, one of the greatest Classic Desk Accessories (CDA) ever written. Dave Lyons, now working for Apple Computer, is the author of this beautiful program. Originally, Nifty List was just supposed to be able to display a monitor listing, but features kept getting added, and now it's a full-blown programmer aid. The functions that I use most often are seeing what memory handles have been allocated and who they belong to, checking out memory handles that my program has allocated and making sure they are correct, and seeing which resource forks are open.

A new feature that appeared in Nifty List version 3.0 was the ability to make Toolbox calls! Did you forget to do a `SetPort` call in your program? Don't want to re-compile to see if that's all that's wrong? Issue a `SetPort` call from Nifty List and continue! Do you want to start up an additional tool? Issue a `LoadOneTool` call and then the appropriate startup call. You can then make sure that the tool is started by using

the "v" command (this command shows the version numbers of all tool sets and whether or not a tool set is started).

Groovy Tips

Because I use Nifty List so much, I've compiled a few tips for using it to help you get your work done much faster:

- Don't forget the "^" (3-byte indirection or dereferencing) operator! If you have handle `E0/65C8` and you want to see what's at that location, just issue a `E0/65C8^;h` and all the work is done for you. This saves *lots* of time.
- Use the command history to your advantage by storing comments about your work—using the "!" as the first character on the line designates the line as a comment line.
- Use the "s" command to see all the current prefixes—when you're working with changing paths, this helps immensely.
- "=" explains things. Use it. Half my commands in Nifty List are = commands.
- The "\err" command gives you what error occurred and which tool set it refers to. It sure beats leafing through books.
- Ever forget the full name of a Toolbox call? Use "" to find all matches. (I use this command lots.)
- If you need to find out what a resource number means, use the "\rtype" command.
- Use the "I" command to track down all memory referenced by your program. Also, use it to see all the memory used by desk accessories (5000I), tools (4000I), the operating system (3000I), and init files (A000I) if you're curious. Use "H" and "W" for information on specific handles.
- ";C" shows control records. Use it to make sure your controls are defined properly.
- ";N" and ";C" dump NDA and CDA headers so you can find the entry points, if you're curious. This helped me to track down a bug with ORCA/C 1.1 NDA init procedures.

GSBug

GSBug is *the* debugging tool to have, especially if you're an assembly language programmer. With GSBug, you can step

and trace through your program to see exactly what's going on. Some great features are the ability to set tool breaks and operating system breaks which will drop you into GSBug whenever a specific tool call or GS/OS call is made.

One of the many features of GSBug is base conversions and simple expression evaluation (with +, -, *, and /). This isn't the main purpose of GSBug, though, but it sure beats doing math in your head when you can't find your calculator because it's buried under printouts and other junk that may be cluttering your workspace beyond all belief. And to stray a bit off the point, clean up your work area! A fresh, clean desk is wonderful!

Here are some tricks for using GSBug that I've compiled:

- Put in a useless tool call (such as `SysBeep`) at the start of your program (or anywhere else that you need to trace) so you can set a tool break on the call to gain control whenever your program starts into some critical section.
- Be sure to use the "X" command when stepping through a piece of code so you don't have to step through irrelevant subroutines. In fact, you can use the "X" key to advance your code normally so you don't have to watch for `jsr`'s and `jsl`'s.
- Use the templates! Break on a suspicious tool call or OS call and check the data structures before and after the call to make sure they are correct. Whenever you get a bad parameter count GS/OS error or an invalid pathname, check the template to see what things really are. (Don't forget that `_templates 0` will list all the major template groups and you can list the individual templates from there.)

There are so many features of GSBug that it's hard to describe, but believe me, you'll recover your investment in the program after your first night of debugging with it.

Talking About Templates . . .
When you get the Nifty List 3.2 package, you might notice that there is an external

Contest #4 Update

By Steven W. Disbrow

A Skunk By Any Other Name

Well, we've already gotten more response on this contest than *all* of our other contests combined. (A special honorable mention goes to David R. Hopkins who has sent in no less than 31 entries.) For those of you that are just joining us, the purpose of this contest is to find a new name for *GS+* Magazine. (If you want to know *why* we are doing this, pick up a copy of *GS+ V2.N3*, it explains all.) There is only one prize in this contest: a lifetime subscription (Magazine and Disk).

Here's how the contest works:

- 1) Come up with a new name for *GS+* Magazine.
- 2) Send it in.
- 3) Win a lifetime subscription.

Our Current Favorites

Granny Smith's Almanac
(Suggested by Keith Bekofske . . . sort of. Keith sent in the "Granny Smith" part and I added the "Almanac." A certain person here in the office did not realize that a "Granny Smith" is a type of apple. When pressed, they also admitted that they did not know that a McIntosh was a type of apple.)

G/S+ Magazine

(Suggested by both Charles Bartley and Ron DeCostro. Both entries had the same postmark date.)

Honorable Mentions

GutS+ Magazine

(Suggested by Brian Winn)

Gee yeS

(Lots of folks suggested this. It's probably the single most suggested title.)

Downright Wierd Suggestions

Dave's Digest

(Suggested by our own Big Dave Adams. Dave also came up with the ideas for RAM Namer and The New Order. Go figure.)

GuS+

(Richard Cain calls his IIGS "Gus" and suggested we name our magazine after his computer.)

To Hell With Apple!

(Suggested by almost as many people as Gee yeS. The only problem here is that *two* of these words are trademarked company names. Guess which two.)

Personally, I like the "Granny Smith" one best. But I want to know what you think! Remember, if none of the names we get are good enough, there will be *no* winner and we will probably end up naming the magazine "G+S Magazine."

However, if you come up with a great name (and send it in to us—that's the important part), and we use it, we'll give you a lifetime subscription with diskette! Of course, if two of you send in the same name, we will take the one with the earliest postmark. If we get two or more identical entries with the same postmark, we will have a drawing to determine the final winner.

The deadline for this contest is April 15, 1991. The winner, if there is one, will be presented in the May/June issue (V2.N5).

Send those entries to:

GS+ Contest #4
P. O. Box 15366
Chattanooga, TN 37415-0366

No e-mail or phone entries please! Make *sure* you include your **name**, **address** and **phone number** with your entry! **GS+**

The Molehill

. . . continued from the previous page

command module called Templates 1.0 which allows you to use GSBug style templates from Nifty List. I got really tired of tracking down memory to look at with Nifty List and then dropping into GSBug to see it with a template, so I wrote the command module to let me use templates right from Nifty List. Even if you don't have GSBug you can still use templates, provided that you make up your own template file.

Together, At Last!

The combination of Nifty List and GSBug is nothing short of phenomenal. For example, you can drop into GSBug after a *NewHandle* call and then use Nifty List to find out everything relevant on that new

memory block. You can step through some code that adds an item to a list control with GSBug and then test what additional tool calls will do to the list control with Nifty List. You can also use Nifty List to find the call number for a GS/OS call (for example, issue "Open") and then set an OS break with GSBug since OS breaks can only be set by number. The possibilities are endless.

Send It In

If you use any debugging techniques that aren't described here, send them to me! I'd love to hear about them, and even throw them together in a future Molehill column. They don't have to be specific to a particular debugging aid such as GSBug,

they can be along the lines of "put a print statement in to track your variables" just as long as it is useful.

Nifty List version 3.2 is a \$15 shareware product. You can download it off of virtually any online service—including our own pro-gsplus BBS (615-875-4607). To buy Nifty List, send your \$15 (along with who you are and how you found out about Nifty List) to DAL Systems, P.O. Box 875, Cupertino, CA 95015-0875.

GSBug is available from APDA (1-800-282-2732), the part number is A0037LL/A and it's \$30. You must be a member of APDA to order this product. **GS+**

Harmonie vs. Independence

By Steven W. Disbrow

Harmonie v1.02

Software by Bill Heineman

Retail price: \$49.95

Typical mail-order price: \$29

Vitesse, Inc.

P. O. Box 929

La Puente, CA 91746-0929

(800) 777-7344

Independence v1.0

Software by Steve Stephenson

Retail price: \$39.95

Typical mail-order price: \$22

Seven Hills Software, Inc.

2310 Oxford Rd.

Tallahassee, FL 32304

(800) 627-3836

Both packages require System Software v5.0.2 or later and at least 512K RAM. Some printers may also require additional memory.

Harmonie? Independence? What Are They?

Well, aside from being a couple of philosophical concepts, Harmonie and Independence are two sets of GS/OS printer drivers. When you install a set of these drivers on your IIGS, you can use your IIGS desktop applications (AppleWorks GS, GraphicWriter III, etc.) to print on several different types of printers that are not supported in the generic Apple IIGS System Software release. Most notably, you can purchase and use either a Hewlett-Packard (HP) LaserJet or DeskJet printer. (This is why Vitesse called their package Harmonie—it helps your IIGS live peaceably with non-Apple printers.) Another benefit of these packages is that you can get 300 dots-per-inch (DPI) output without having to cough up \$3,000+ for Uncle Sculley's retirement fund. (Which is why Seven Hills calls their package Independence—it helps your IIGS live peaceably with non-Apple printers.)

Figure 1 Printers Supported By Harmonie

HP DeskJet, DeskJet Plus, DeskJet 500
HP PaintJet, PaintJet Plus
HP LaserJet II, LaserJet IIP, LaserJet III
Epson LQ 24-pin printers
Epson MX-80
ImageWriter II, ImageWriter LQ
Nec Pinwriter P-2200
Canon 1080

The Big Difference

Harmonie and Independence are two products that are meant to do *exactly* the same thing: give IIGS users the ability to print high-quality, 300 DPI text and graphics on low-cost, non-Apple printers. So, the overall question is: who does the better job?

Unfortunately, that question has two answers. You see, Harmonie supports a *lot* more printers than Independence does. If you don't believe me, take a look at Figure 1 and Figure 2. (The printers shown in *italics* in Figure 1 are the printers that Independence supports.) Quite a difference eh?

So, if you have your eye on a HP PaintJet, or if you have an old MX-80 in the attic, the choice is pretty obvious: you should get Harmonie.

Common Ground

However, if you have one of those printers that both products support, the choice becomes much less clear. (At this point, dear reader, I must warn you that the only printer I have used Harmonie and Independence with is the HP DeskJet. I have *seen* output that each program has generated on other printers, but I have not used those printers myself. I must also warn you that *this* review is based on version 1.02 of Harmonie and version 1.0 of Independence. We received Harmonie v2.0 about three days before we went to the printer and did not have time to test it extensively. See the "Harmonie v2.0" sidebar for more info on this product.) To

sort things out, we need to make some fairly direct comparisons between the products. For me, there are only three areas where direct comparisons will make any sense for this type of product: Documentation, quality of the printed output, and features.

Notice that I did not include "speed" in that list. To me, speed is somewhat irrelevant—it's the final product that matters. Furthermore, each package has many different features that affect that final product and the time it takes to achieve it. So, as far as the speed issue is concerned, let me just say that when using a DeskJet, both products are slower than an ImageWriter. Of, course, your output *will* be 300 DPI. Also, if you are using a LaserJet, the speed will improve dramatically.

Documentation

The documentation for Independence is approximately 50 pages long. Thirty-one of those pages are documentation for the individual printer drivers and the rest are appendices and an *index*. There are numerous figures and tables detailing the operation of the drivers and the different HP printers. The Independence documentation is written with the novice in mind and covers everything from connecting your printer to the IIGS to actually using the drivers.

The Harmonie documentation is 10 pages long. It covers installation of the drivers and gives an explanation of the various options available from the LaserJet IIP driver. The last page and a half of the manual gives tips on how to use several of the features contained in the various drivers.

Figure 2 Printers Supported By Independence

HP DeskJet, DeskJet Plus, DeskJet 500
HP LaserJet IIP, LaserJet III

Figure 3

Text Quality

To paraphrase an old cliché, "printouts speak louder than words."

As you might guess, Independence is the hands-down winner when it comes to documentation. While it is not surprising to me that Seven Hills did such a good job with the Independence documentation, it is very surprising that Vitesse did such a shoddy job with the Harmonie documentation—in the past their documentation has been absolutely top-notch. Hopefully, this will be fixed in a future version of Harmonie.

Text Quality

To paraphrase an old cliché, "printouts speak louder than words." Figure 3 contains a block of text printed out at 150 DPI by Harmonie and Figure 4 contains the same block of text at 300 DPI printed by Independence. Both blocks of text were printed from EGOed v1.33. Why didn't I print both samples at 300 DPI? Well, I could not get Harmonie to print at 300 DPI—it crashed my computer (a ROM 03 IIGS with 2 MB RAM). Independence had no problems at 300 DPI.

Graphics Quality

Figure 5 contains a simple graphic (scanned with the Quickie Hand scanner) printed at 150 DPI by Harmonie. Figure 6 is the same graphic printed at 300 DPI by Independence. Both graphics were printed from Platinum Paint. Again, Harmonie would not print at 300 DPI. Independence had no problems here either.

To be fair, I have seen some extremely good 300 DPI printouts done from

Figure 5



Harmonie v1.02 on other printers and I have seen beta versions of the v2.0 DeskJet driver that fix this 300 DPI bug. However, I can only write about what I have and what is commercially available now.

Features

Both Harmonie and Independence have a wealth of features to help you "fine-tune" your printouts. As you would expect, both Harmonie and Independence allow you to print in either portrait or landscape mode and, of course, you can select a range of pages to print.

As for special features—well, Independence let's you "Save Ink" if you are using a DeskJet printer, and you can easily set the vertical and horizontal scaling using simple percentages. For example: the graphic in Figure 6 was printed at 25%, both vertically and horizontally, of its normal size. Independence also allows you to print only odd or even numbered pages, multiple copy printouts can be collated and, as a default (which you can turn off), multiple page printouts come out "last page first" so that when you pick them up, they are already in the proper order.

Harmonie also lets you specify horizontal and vertical scaling, but it's not as easy to use as Independence. Scaling factors are set via a pair of scroll bars and are expressed in dots-per-inch as opposed to percentages. To approximate the 25% scaling factor that I used in Independence, I set the horizontal DPI to about 280 and vertical DPI to about 120. Really intuitive, right? Actually, it took several tries to come up with those numbers, and after a while I just said "good enough for government work" and gave up. Fortunately, there is a pop-up menu containing several of the more useful settings for these scaling factors. The choices include ImageWriter Normal, ImageWriter Condensed, DeskJet Normal, DeskJet Condensed, DeskJet Best, and Macintosh Normal. Simply select one of these items from the menu, and the horizontal and vertical scaling factors change automatically. This is handy, but it would be even better if the scaling factors were expressed as percentages.

When you get past that, however, Harmonie really begins to rack up on features. For

Figure 4

Text Quality

To paraphrase an old cliché, "printouts speak louder than words."

instance, you can specify an additional half-inch margin at the top of the paper (I think this is for the DeskJet only). You can specify four levels of contrast for your printouts. You can tell Harmonie how many shades of grey to use (2, 3, 5, 9 or 16) in your printout. You can specify either automatic or manual paper feed. You can tell Harmonie to print bi-directionally (i.e. the printer prints as the print-head moves each way across the paper). You can even specify the level of data-compression to use for sending information to the printer.

But, the best feature of Harmonie is its ability to use the fonts that are built into your printer. For example, the Courier font is built into every DeskJet. Using this capability (which is called "External Rendering"), Harmonie can print text-only rough-drafts in a fraction of the time it takes Independence. The reason for this is that, using External Rendering, Harmonie sends only raw text to the printer. The printer gets the text and prints it immediately. Independence has to generate a bit-map image of the text and send that to the printer. The printer treats it just like any other graphic and it takes a while to print.

Earlier I said that speed was not that important a criteria to me. That's true when I'm printing things with graphics and text—which is 99% of the time. In those cases, I just want the output to look good. However, when all I need is a quick

Figure 6



printout of some text, the time difference between Harmonie and Independence is just too great to ignore—we're talking several minutes here!

As great as Harmonie's External Rendering feature is, it's not complete. Harmonie can only use the fonts that are built into the printer. There is no support for downloadable "soft" fonts or font cartridges. This is really a shame—there are literally thousands of high-quality soft fonts and fonts on cartridges available for HP printers. This is one of the main reasons HP printers are so popular in the MS-DOS world.

With the exception of Harmonie's External Rendering, Harmonie and Independence both use the fonts that they find in the **System:Fonts:** folder of your startup disk to draw text on your printer. Since these are bit-mapped fonts, it is a Herculean task to make them look good on paper. So, to make the text look *really* good on paper, both products use a technique which I like to call "quadruple-size font squashing." Basically, what happens is this: when you try to print some text that is set in, say, 12-point Times, Harmonie and Independence look in your

System:Fonts: folder for a 48-point Times font. If they find this quadruple-sized font, they use a special font squashing algorithm to reduce it down to 12-point Times. Because the original font was so much bigger, it contains much more detail. The trick, then, is to retain as much of this detail as possible after squashing the large font. While this extra detail would be lost on the computer screen, it translates quite nicely to the printer—which has a much higher resolution than the IIGS screen.

So, having said that, where do you get these monster-sized fonts? Well, the Independence package contains a separate disk packed full of these fonts—all the way up to 96-points! Harmonie does not come with *any* large-size fonts. Instead, the Harmonie manual suggests you check your local user group or BBS.

Other Stuff

When I first bought my DeskJet, the friendly salesperson informed me that I would have to have a parallel-printer card in order to use it with my IIGS. So, I put down an extra \$80 and got one. When I got home, I found a section in the DeskJet

manual that was actually devoted to the IIGS! It said I could simply plug the DeskJet into one of the serial ports. The DeskJet comes with both a serial and parallel port, so you can do it whichever way you want. I wanted to do it the way that was \$80 cheaper. So I took the parallel card back, informed the salesperson of his "error" and, when I got back home, plugged the DeskJet into my modem port.

Harmonie comes with several drivers for some of the more popular parallel cards, so, if you have one, you can keep right on using it. In fact, the Harmonie documentation *recommends* that you use a parallel card. The reason for this is that printing through a parallel card is usually faster than printing through the serial port. Of course, that really depends on the speed of the serial port. One of the reasons people consider serial communications so slow is that most folks think serial speeds top out around 2400 baud (2,400 characters per second). Actually the IIGS serial ports can handle speeds up to 19200 baud! By a neat coincidence, so can most printers!

Harmonie v2.0

Ah! The reviewer's life! No sooner do you finish reviewing a product than they bring out an update to it! Unfortunately, Vitesse did not get it out in enough time for us to thoroughly test it for use in this review. However, I can tell you a few things that I did notice from the few minutes I got to play with the DeskJet drivers in Harmonie v2.0.

Most importantly, the 300 DPI bug has been fixed! At the left and right of this sidebar are Figures 7 and 8 showing

Figure 7

Text Quality

To paraphrase an old cliché, "printouts speak louder than words."

Harmonie v2.0's 300 DPI printouts of the same text and graphics that were presented in Figures 3 through 6.

Second, several of the features that were unique to Independence are now in Harmonie. For example, Harmonie can now print only odd or even pages, and it can print "last page first." A "Save Ink" option has also been added. Still missing, however, is the ability to collate pages and the ability to set scaling factors via simple percentages. Harmonie v2.0 also adds several new features that Independence does not have—such as the ability to control the amount of Red, Green and Blue that appear in color printouts. Harmonie v2.0 also saves separate page setup information for each program that you use it with. Very handy.

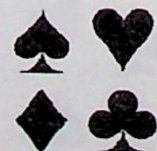
Third, Harmonie v2.0 ships with several large-size fonts on the disk. There are

nowhere near as many as you get with Independence and the largest size is only 72-points, but it's better than nothing!

Fourth, Harmonie v2.0 comes with a documentation file on disk that looks to be about twice as big as the printed manual!

And, finally, the retail price of Harmonie has been raised to \$54.95. Registered owners of Harmonie v1.02 will be receiving the update to v2.0 free of charge.

Figure 8



Which brings us to Independence. In the Independence manual, it is recommended that you connect your printer to the serial port and you set your serial port and printer to communicate at 19200 baud. Now, this is *probably* still slower than a parallel connection, but it's not slower by much, *and* it's \$80 that you will be able to spend on something else. Another advantage to using the serial port is that you only need a simple IIGS to ImageWriter I cable to make the connection. These are available under the name "Mac Plus to ImageWriter I cable" for about \$15 from just about any mail-order company. If you are a soldering-iron kind of guy (or

gal) you can easily build your own. Mine's about 30 feet long and has not failed me yet.

Fortunately, Harmonie works quite well through the serial port at 19200 baud—even if the documentation does not say so.

So, Who Does The Better Job?

These are both great products and it would be really hard to go wrong with either one. Both Vitesse and Seven Hills are major players in the IIGS market and both have shown their firm commitment to enhance their products.

However, at this point, superior documentation, a nice collection of large-size fonts, and the ability to actually print at 300 DPI gives Independence a clear edge. It's superior documentation makes it especially well suited for "printer novices."

Of course, if you have a printer that only Harmonie supports, you should get Harmonie.

The best solution, of course, would be to get both packages. The combination of features is well worth the extra (and relatively low) cost. After all, you probably saved about \$2,500 dollars by not buying a LaserWriter, surely you can spare another \$50. **GS+**

Noreen's IIGS Tips

By Noreen Ribaric

If you have ever had to wait on that "Saving the clipboard..." dialog when quitting an application, there is a solution to this. (I once had to wait over a minute to return to the Finder from AppleWorks GS because the last thing I had copied was a block of about 100 cells from a spreadsheet!) Just copy a blank line (or cell or part of your graphic, depending on what application you're using) right before quitting so that there won't be that much of a clipboard to "save." Thanks to Joe Wanker! for pointing this out to me!

If you use lookup tables in AppleWorks GS spreadsheets, you should use absolute referencing (add a dollar sign before the row and column of the cell) when entering the cell range in the VLookup or HLookup formulas. That way, if you have to copy the formula, say five rows down, the cell ranges won't be adjusted by 5 cells (which could either result in an error or an incorrect value).

There is a problem with exporting text with multiple styles using the Universal translator in GraphicWriter III (GWIII) v1.0—it starts repeating itself when it gets to the first change in style. A solution to this would be to use a different export translator (like our TeachText translator), and then load that document into a word processor or text editor (like

EGOed) to convert it to plain text (if that is the format you wanted).

If you copy (or cut) and paste several typing frames that are linked together in GWIII, after pasting, text will only reappear in the *first* typing frame. The links are apparently lost, but the text is still all there. All you need to do is re-link the typing frames (I almost panicked when I thought I had lost most of a pretty big review, before I tried re-linking the frames)!

This last tip is a biggie. It has to do with deleting stuff from GWIII page layouts. We have discovered that if you copy a page layout, and then delete everything and resave it, the page layout still takes up as much disk space as it did before, even though there is nowhere near as much stuff in it! Even worse, it can only get bigger! We use GWIII to lay out our magazine in several layouts of 3-5 pages each. We used to open old layouts from the last issue (since a lot of stuff stays the same), delete some stuff, add other stuff, then save them with a new name (or use the Open As Copy option). Then when we were ready to do the next few pages, we would open the new file, delete and add to it, and save it as a new file (and so on). Well the first, 5-page layout would be only about 25K or so, but by the time we got to the last pages in the magazine, the layout was pushing

250K (and the last layout usually only had one page in it)! Apparently, when stuff is deleted, the "links" to it are erased, but the actual data is still present somewhere in the file. Using Disk Access to "show" the file verified this. Maybe this is why we were getting more and more "corrupted" files as we got closer to finishing the magazine. The way we solved this was to create a layout with only the master page info. Now we *always* use this "master" file when creating subsequent layouts. Even if you want to create a second copy of a page layout in which you want to change only a few things, it would probably be safer to open a new page layout and copy only the parts you want from the old one to the new one. This way, you will be sure that only what appears on your page layout is actually in the file. It took us almost a year of frustration (and many huge, and sometimes "corrupted," files) before we realized what was happening. We have notified Seven Hills of this problem.

If you have any tips on using any application for the IIGS that you would like to share with other IIGS users, you can send them to:

Noreen's IIGS Tips
c/o GS+ Magazine
P.O. Box 15366
Chattanooga, TN 37415-0366

GS+

InWords

Programmed by Alan Bird

Retail price: \$129

Typical mail-order price: \$80

Not copy-protected

Requires 512K RAM

WestCode Software

11835 Carmel Mountain Road, Suite 1304

San Diego, CA 92128

(619) 679-9200

FAX: (619) 451-0276

Reviewed by Jonah Stich

[Editor's Note: InWords is *not* a IIGS-specific piece of software. However, it is an extremely important piece of software that a great many of you have asked us to review. So, we bow to your request. Now, contact WestCode and tell them you want a IIGS-specific version. Perhaps *they* will bow to your request also.]

InWords is the very first Optical Character Recognition (OCR) program for the Apple II series of computers. OCR is the process wherein printed matter is converted into text that can then be edited with any standard word processor. Keep in mind that InWords can only recognize printed matter—hand-written text varies too much to make it recognizable. At this time, InWords requires a Vitesse Quickie hand scanner (reviewed in *GS+* V2.N2) for text input.

InWords is very easy to use. The basic operation consists of selecting the Scan item from the main menu, scanning the text, and letting InWords get to work converting it. At each step of the process there are numerous parameters that can be altered to affect InWords' operation.

At the Scan menu, you can select between "Standard," "Merge," and "Column" scan methods. The standard scan is somewhat mislabeled, as it's probably not the mode you'll use the most. However, it is the mode with the fewest frills, so it was dubbed standard. In standard mode, you simply pull the scanner down the page and any characters InWords detects are converted to text. Merge mode is used when the text to be converted is wider than the 4" scanner

head. In this mode, you are prompted to scan the left half of your wide page. InWords converts that swath and then prompts you to scan the right section. The program then converts this section and automatically pastes the two sections together. To get the two sections properly aligned, you must leave a bit of overlap between them, and because you are limited to two passes, your page better not be wider than about seven inches. That's not as much of a problem as it would appear, as most printed matter that is wider than about six inches comes in columns. InWords' third mode, the Column scan, can be used in these situations. In this mode, after you have scanned a section of the page, InWords converts only the text in what it deems the "middle column" and ignores all of the other text on the page. For example, if you'd like to scan an article out of the *New York Times*, you make several passes, centering a separate column in each scan, and InWords ignores the parts of other columns that you may have scanned. You can then join all of these scans together and have a copy of the original article. In all three scan modes, InWords does its best to ignore graphics, column lines, and other non-text parts of the scan, and although it sometimes gets confused, it is, for the most part, quite successful.

Another group of choices that can be made from the Scan menu deal with the current Font table. The font table is what InWords uses to recognize characters. If it is not sufficiently well-mapped to the font of the printed matter, InWords will not be able to accurately recognize the characters and the converted text will be extremely garbled. The Scan menu allows you to choose the current font table, as well as the "Recognition precision," or how close a match must be made for the character to be recognized. You can also turn "Font training" on and off. If this option is on, InWords will stop every time it doesn't recognize a character and ask you what it is. This character will then be added to the current font table, so that next time it is encountered, InWords won't have to ask you again. This option is most useful when creating a completely new font table, either from scratch or by modifying an existing table. The user may create as many "custom" font tables as they want.

The final two options on the Scan menu allow you to dictate whether the newly scanned text should be appended to the previous scan's text or replace it, and whether or not to show you all of the characters as they are recognized, so that you can see how the character patterns are being converted to text. Usually this option is left off because it greatly decreases the speed of conversion.

After you have scanned the image, it is shown in graphic form, greatly magnified from the original, so that you can make sure that all of the characters came out well and weren't too faded or too dark. If the scan is messed up, you can now abort the conversion and go back to rescan the material. If you feel that the scan is okay, you simply press return and InWords sets to work converting the text. You now go to InWords' built-in text editor, where the text is placed when it has been converted. When the entire scan has been converted to text, you are left in control of the editor. You can now edit out any mistakes that InWords made in converting the characters. When you are satisfied with the conversion, you may save the text as either a text file with line breaks only between paragraphs, a text file with line breaks at the end of every line (for when you would like to retain the formatting of the text as it originally appeared), or an AppleWorks Classic file. If you don't like the text, or if you would like to append more text to it, you can go back to the Scan menu and do more scanning before saving the text.

Recognition accuracy, while seldom 100%, is quite acceptable. When InWords cannot decide what a character is, it simply replaces it with an "unknown" character. The default for this character is a tilde (~), but the user can specify another character if the text to be scanned already contains a large number of tildes. Thanks to this substitution, it is easy enough to load the output file into your word processor and use the find command to quickly fix all of the unknown characters. Unfortunately, sometimes InWords is just flat out wrong. For instance, sometimes it'll replace 'm' with 'm'. An easy mistake to make (I do it myself sometimes when reading late at night . . .), but a bit harder to find. For the most part, text imported by InWords

will need to be run through a spelling checker and then manually scanned for errors.

To give you some idea of the accuracy of the program, I used the column scan method to scan the second column on the first page of Diz's review of the Quickie scanner. I set my Quickie to 400 dpi (dots per inch) and used the "Standard" font table, one of the dozen or so that comes with the InWords package. The paragraph in the middle of that column, set in Times 10-point, read:

"The Quickie software is even easier to install. If you have a hard disk, just create a new folder and copy the software into the folder, or make a copy of the Quickie disk and run the software from there. There is also a New Desk Accessory (NDA) version of Quickie that you can install so you can use the scanner from within any desktop program."

and InWords converted it to:

"The Quic~e sof~ware is even easier ~o ~stall. If you have a hard d~k, just create a new folder and copy the ~ftw~e in~o the folder, or make a copy of the Quic~e disk and tun the ~ftware from there. There is also a New Desk Accessory (NOA) version of Qu~c~e ~at you can instal~ so you can use the sc~er from with~ any desktop ~~am."

If I had spent a few minutes retraining this table to be more in sync with the font used in GS+, the constant replacement of the 'ki' in Quickie with a tilde could have been fixed. Problems like the replacement of 'run' with 'tun' and 'NDA' with 'NOA' probably would not have been avoided.

On my IIGS with a ZipGS accelerator running at 8 MHz, it took InWords 31 seconds to completely convert the entire middle column to text. With the ZipGS turned off, it took about 65 seconds. According to InWords' "recognition summary" command, there were 1586 characters, in 384 words on 53 lines, in the scanned text. Of those 1586, InWords replaced 110 of them with tildes. Counting the number of improperly converted characters, there were actually 136 mistakes, or about 86% accuracy on this scan. The

first scan I ever attempted had lower accuracy because of improper settings on the scanner, and a scan I did from the manual with a specially trained font table had 100% accuracy. If you're willing to spend the time training the table, most scans can get into the 90%+ accuracy range.

For all of its power and innovation, InWords suffers from some major flaws. Chief among these, in my view, is the fact that it is a ProDOS 8 program, running in text mode, using an AppleWorks Classic filecard menu system. I think the reason for this is that the program is designed to work on any Apple II. Well, at least those Apple IIs with MouseText support and 512K of RAM. Because this describes every IIGS, but only a handful of IIs, I feel that a Desktop version is warranted.

Because of the use of ProDOS 8, I can only access the first two partitions on my six-partition hard drive without modifying my system software, and all of my NDAs are unavailable. More importantly, text can only be converted into the standard alphanumeric characters. With the IIGS's extended character ability, this is a major drawback. A simple example of the problem comes from the suggestion that when making a new font table you train the character © as the combination of characters (c) so that information will not be lost. If the desktop had been used, the © symbol could have been trained as a ©. A more drastic example is the issue of languages with other characters. An extreme example: if InWords ran in a desktop environment, a user who has a font installed on their IIGS with a Chinese character set could train InWords to recognize Chinese and convert it to Chinese that they could then edit in their word processor. With the use of text mode, this is very hard to do. Because of what InWords does, the failure to use the desktop is not only an inconvenience—it actually limits the capabilities of the program. An NDA version of InWords would be supremely useful, but I'd be happy with just a stand-alone desktop application.

Also missing from InWords is a landscape scanning mode. It is almost impossible to get a good scan out of a paperback book because of the tight binding and the edge on the scanner head. If you could pull the

scanner sideways across the page, however, it would be somewhat easier to capture the text.

Other little gripes are the somewhat underpowered editor and the restrictions on location of font table files. The editor is not very full-featured (or very friendly—there's no help command) but as the expectation is that you will export the text to your regular word processor to do any serious editing, this is excusable. The restriction of the editor that files must be no more than 32K is again frustrating but can be worked around. Not so excusable is the requirement that the font table files be located in the same folder on the disk as the program itself. For people with only a few font tables, this is not a problem. For those who would like to do a lot of scanning and have created a separate font table for each of their common sources, this begins to get a little messy—particularly for those who use InWords from a floppy disk. Because of the way disks on the Apple II work, there can only be 51 files in the root directory. Subtracting a few for ProDOS and the InWords program files, you can only give InWords access to about 45 font tables when running from floppies, unless you put InWords in a folder on the floppy. Luckily, you're not forced to save your converted text to the same directory. Also annoying is a bug that causes InWords to lock up when long horizontal lines are scanned—it apparently spends forever trying to figure out what that long, short "character" is. Luckily InWords allows you to hit control-reset to return to the main menu and continue working without losing data (although Control Panel access seems to be disabled after you do this).

InWords is a powerful and very useful program. Although WestCode somewhat undermined themselves with their failure to use the IIGS Desktop, the program is still very usable, and the possibilities it provides are fantastic. If you have the need to input printed data into your Apple II, or just want to play with OCR, InWords is a fantastic program. Hopefully we'll see more innovative and much needed programs (including an InWords update) from WestCode in the future!

GS+

Allison - Version 1.02

Program by W. Scott Gentry and
Jonah Stich

Retail price: \$35

Requires Enhanced VisionPlus (Visionary GS) Video Digitizer, 1 MB RAM (all memory must be DMA-compatible), System Software v5.0.3 (or later)

New Concepts, Inc.
665 West Jackson Street, Suite 2
Woodstock, IL 60098
(800) 869-9152

Reviewed by Greg Zimmerman

What Is It?

Allison is digitizing software that dramatically increases the quality of the pictures that you get when digitizing using the Enhanced VisionPlus (or Visionary GS) video digitizing board.

A Little Background Is In Order

The Enhanced VisionPlus is a video digitizing board for the IIGS. The board allows the user to input into the GS and digitize photos from a video camera, a video machine, a laserdisc and other video sources. It will work with just about any device generating a NTSC standard video signal, and *most* video cameras. The resulting digitized pictures can then be used in a variety of programs for further processing, including paint programs or multimedia programs like HyperStudio.

For example, if my kids wanted to do a HyperStudio stack on Arizona plant life, they could take video shots of all the plants that they could find and identify, digitize frames from that video using Allison and the Enhanced VisionPlus, and then use the resulting digitized screen shots in the stack they were working on. The possibilities are endless.

Getting back to our story—Virtual Realities stopped selling the Visionary GS, and New Concepts, Inc. (whom I believe are the same people as LRO Computer Sales) began to license the

board from Silicon & Software (who manufactures the board). The name of the board was changed to the Enhanced VisionPlus at some point, but I have recently seen it advertised again as the Visionary or Visionary GS. I'm going to call it the Enhanced VisionPlus board for the rest of the review, but apparently that name is being used interchangeably with Visionary GS by some commercial establishments. It is being sold by LRO Computer Sales (\$289) and Quality Computers (\$249.95). The board may also be available at other mail-order houses, but the two above named sources were the first current ads I located.

Shipping with the Enhanced VisionPlus, and also available separately for \$35 to those who bought the board from Virtual Realities prior to the release of the software, is Allison. Allison is the software that is used in conjunction with the Enhanced VisionPlus to produce digitized images.

For those with boards from Virtual Realities that need only the Allison software, Scott Gentry of Excellent Software Inc. (the developers of Allison) also offers a service wherein he will recalibrate your board to better match the Allison software. The fee for this is \$16, which includes return shipping of your board.

The Details

Allison supports 320-mode color digitizing as well as black and white. It will save pictures in the following formats: Raw Image, Screen, or Allison Preview.

Allison is System Software v5.0.4 compatible, is not copy-protected, and can be run from either a 3.5-inch floppy or a hard drive.

Allison follows the Apple Human Interface Guidelines and is very easy for any experienced Apple IIGS user to operate. In addition, the program is accompanied by an extremely well-written manual that thoroughly explains all of the features of the program in an easy-to-understand fashion, as well as giving many helpful tips on getting the most out of the program.

Allison allows the user to select from among several options to adjust and improve output quality. Dithering, palette, aspect ratio, contrast, and brightness all have varying options that may be selected by the user.

Allison also supports printing of pictures. I got very good quality output using my ImageWriter II.

Also included with Allison is Allison/3200, which can be used to convert raw images created with Allison



into 3200-color pictures, and also to display 3200-color pictures made with either Allison/3200 or other programs, such as The Graphic Exchange, that create 3200-color pictures.

I have spoken at length with both New Concepts and Excellent Software concerning both the Enhanced VisionPlus board and Allison. These are extremely knowledgeable and helpful people who really care about the product and who were able to answer all of my questions.

Is It Any Good?

This software, when used with the Enhanced VisionPlus video digitizing board, can produce results that are nothing short of spectacular in comparison to what you may be used to seeing on the Apple IIGS.

I am a complete amateur when it comes to video, photography, and things like aspect ratios, dithering, etc. (There is not enough room to properly denote my amateur standing.) Also, I have a 10-year-old (at least) bulky Panasonic home video camera that is not what might be considered state of the art. Lastly, my video machine, which is the same age as

the camera, needs to be pounded or jolted occasionally (this is no joke) to get it to work. Yet I was able to get excellent results with the Allison software the very first time I used it. Hard to believe? Well, on the GS+ Disk this issue, in the folder Allison.Rev, are pictures of two of my kids that I took the first time I used Allison. (One of these is also printed out on the previous page.) You can see from these pictures that after 10 years I still haven't figured out how to get the date to change on the video display that is built into the camera—but the pictures that Allison captured speak for themselves.

In black-and-white mode, Allison digitizes in almost real time. In color mode, it not only produces better pictures than its closest competitor, but it works much faster. I was never able to get acceptable digitized video input until this software was released. Now, it's both fast and easy.

Anything Wrong With It?

Not really. The software runs smoothly and quickly. It does everything it claims to do, and it does it better than any other product on the market for the Apple IIGS.

The only problem a user is likely to run into with the software/hardware combination is that it makes digitizing so easy, that memory is used up in a hurry. And the resultant picture creation eats up disk space as well.

Of course, you need the Enhanced VisionPlus digitizing board to use the software, so the resultant combination can set you back a few bucks.

The Summary

Allison is as good as it gets with the Apple IIGS. Amateurs can (and do, at my house) produce great results quickly and easily using this software. The picture quality produced by this software is several steps above anything that I have come across for the Apple IIGS. My kids are using it, I am using it, and I recommend that anyone interested in working with video and the Apple IIGS use it as well. GS+

Warranty Disclaimer

APPLE COMPUTER, INC. ("APPLE") MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, REGARDING THE APPLE SOFTWARE. APPLE DOES NOT WARRANT, GUARANTEE OR MAKE ANY REPRESENTATIONS REGARDING THE USE OR THE RESULTS OF THE USE OF THE APPLE SOFTWARE IN TERMS OF ITS CORRECTNESS, ACCURACY, RELIABILITY, CURRENTNESS OR OTHERWISE. THE ENTIRE RISK AS TO THE RESULTS AND PERFORMANCE OF THE APPLE SOFTWARE IS ASSUMED BY YOU. THE EXCLUSION OF IMPLIED WARRANTIES IS NOT PERMITTED BY SOME STATES. THE ABOVE EXCLUSION MAY NOT APPLY TO YOU.

IN NO EVENT WILL APPLE, ITS DIRECTORS, OFFICERS, EMPLOYEES OR AGENTS BE LIABLE TO YOU FOR ANY CONSEQUENTIAL, INCIDENTAL OR INDIRECT DAMAGES (INCLUDING DAMAGES FOR LOSS OF BUSINESS PROFITS, BUSINESS INTERRUPTION, LOSS OF BUSINESS INFORMATION, AND THE LIKE) ARISING OUT OF THE USE OR INABILITY TO USE THE APPLE SOFTWARE EVEN IF APPLE HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME STATES DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU. Apple's liability to you for actual damages from any cause whatsoever, and regardless of the form of the action (whether in contract, tort (including negligence), product liability or otherwise), will be limited to \$50.

MAX/Edit

Programmed by SEA Software Nederland

Retail price: \$50

Only available from publisher

Not copy-protected

1.5 MB RAM recommended

360 MicroSystems

P.O. Box 1192

Oviedo, Fl. 32765

(407) 365-6714

Reviewed by Jonah Stich

MAX/Edit is a new text editor, intended mainly for programmers. It isn't a word processor—if you're looking for the perfect program to write your great novel with, this isn't it. There's no spelling checker, and it doesn't even have a word-wrap feature. If, however, you're planning on doing some serious programming, this editor deserves a close look.

As any programmer knows, the editor is where most development time is spent. A good editor can increase your productivity tremendously, just as a poor editor can slow you down and frustrate you even more than that bug you've been chasing for three weeks. You'll probably spend more time in MAX/Edit than your current text editor, though, because editing is only one of its many features.

Basically, MAX/Edit (which, by the way, stands for Multi-File Adaptive Extensible Editing Environment) is a whole developing environment, designed to be run from a controlling shell. It's obviously designed for the APW/ORCA shell (it will work with ProSel and Merlin, but this takes a bit of know-how).

Installing MAX/Edit on the APW or ORCA programming environments is fairly easy, thanks to the use of Apple's Installer. If you'd like to install MAX/Edit in the Merlin or ProSel shells, you'll need to put the EXE file `MaxEdit` in the `Commands` folder of your development environment. All of the other MAX/Edit files and folders must then go in the same folder as the controlling shell (Merlin or ProSel). This is not stated in the manual, and if you

don't set it up this way, MAX/Edit will not be able to find its data files and will refuse to execute.

The biggest problem I had with the installation was that MAX/Edit comes with its own tool set to help it manage its memory. Unfortunately, this tool is a system tool set, and must be installed in the `*:System:Tools` folder. Aside from the moral issues of appropriating a system tool set number and the limited space problems for people booting from floppies, this can be quite inconvenient when developing files that are loaded on startup (desk accessories, Inits, and CDevs). If your program crashes while the IIGS is booting, you must use another system disk to start your computer, and the likelihood that this system disk has a copy of the special tool is (at least for me) rather low. Therefore, you cannot return to MAX/Edit to fix the bugs, so you must delete the copy of the program on your normal boot disk, then reboot the computer again to fix the bugs. If the tool were a *user* tool set and could be kept in the same folder as the MAX/Edit program file, this mess could be eliminated.

MAX/Edit has so many features that even its programmers probably have a hard time remembering them all. Learning them for the first time can be quite a task, and it is not made easier by several nonstandard key commands. However, one of MAX/Edit's great features is that it has no permanently defined keyboard layout. What function a given keypress performs is dictated by keyboard templates. MAX/Edit comes with two templates, a "standard" one and one that is a pretty close approximation of the standard APW/ORCA editor's key mapping (it cannot be a perfect duplicate because MAX/Edit has about three times the features of the standard APW/ORCA editor). If neither of these suits your fancy, there is also a built-in utility to define your own key template. You can redefine just about any key on the keyboard—if you like using the delete key for return and the space bar for "find," you're free to do so.

Reading the manual is a real must with MAX/Edit—there are just too many

features and options to know what you're doing without at the very least skimming the manual and looking at the command summary in the back. For instance, when copying text, a window pops up asking you to choose from one of eight possible copy methods, to or from one of the four clipboards. There are four find text methods, not counting forwards and backwards, as well as the repeat last find command. At first, all of the options can be somewhat intimidating and can even slow you down (such as when you want to find some text, press Command-F, and start entering the text to find before selecting the find method from the window that Command-F opened). But after you've been using the program for a while, the abundance of features can come in really handy. You'll probably never use them all, but if you do need them, it's good to know they're there.

MAX/Edit's editor is by far the most powerful editor available for IIGS programmers. It allows you to have up to thirty files open at one time, and you can switch between them almost instantly. Files can, allegedly, be up to eight megabytes long, though in practical terms they're limited to the amount of free memory in your computer. Each file is loaded into its own workspace, and each workspace is completely independent from the others. Each has its own "current text position," its own "current text mode," its own "last find command," etc. You can switch to another workspace, edit the file in it for a while, then come back to the first and pick up right where you left off. Because of the special MAX/Edit workspace management tool set, you can even quit from MAX/Edit, run three other programs, then return to the editor, and your files will still be loaded and ready to be edited. There's even a special AutoLoader that will *automagically* load the specified files when MAX/Edit is started—perfect for developing large programs with four or five source files.

You can define separate tab rulers for each language you use. Big tabs for Assembly and small ones for C? No problem. MAX/Edit allows you to enter MouseText characters into your source files. (The mechanism for this is a bit

complicated—you first open what is called the “ASCII Chart,” then use the copy command to copy the character you want, then return to your file and paste the character where you’d like it. If you want to use several different characters, you must do this for each one.) There is an auto-save feature that will, after a user-defined time has elapsed, kick in and save any modified files. There is also a great macro facility, with which you can tell MAX/Edit to convert any keystroke into a sequence of any other keystrokes. There is a tab editor (both built into the editor and as a separate utility), a special MAX/Edit control panel (where you can set several global MAX/Edit parameters), and a built-in calculator.

MAX/Edit even has its own built-in shell, called Min/Shell. This is a great time saver—simply save your file, press the predefined ‘go to Min/Shell’ macro key (option-s), and start compiling it. If there’s a compilation error, you’re dropped right back into the editor, with the offending line centered on the screen. Fix it, save the file, hit escape to go back to Min/Shell, and you can try again. Even if the editor were substandard, the combination of the built-in shell and the ability to open 30 files would probably make it worth the \$50. The time saved in developing programs is astounding. And if a built-in shell isn’t enough for you, MAX/Edit has a Utility program menu (on which Min/Shell is the first entry). You can write special utilities and install them in MAX/Edit, then define a macro which allows you to run them from the editor with a single keystroke.

Another great feature of MAX/Edit is that it is one of those text-based programs that thinks it might be a desktop program. If you forget to save a file before quitting, for instance, a little window pops up, with a little text picture that looks surprisingly like the standard Apple error face, reminding you to save the file. If you can’t remember the pathname of your program, MAX/Edit has a file selection dialog much like the standard desktop one. There are windows that pop up all over the place. But for the lack of a menu bar, the extreme speed (MAX/Edit is a speedy little bugger), and the lack of a mouse cursor

(though the text insertion point can be moved by moving the mouse; this is sometimes more annoying than helpful, such as when you run out of space on the desk, put a book on the mouse, and completely lose your place in the file), this could easily be a desktop program.

Unfortunately, MAX/Edit is not completely bug-free. To its credit, I have never lost a line of source in the two months I’ve been using MAX/Edit (and I use it for at least six hours every day). However, the bugs are still annoying, and there is the potential for lost code. Probably the most annoying of these bugs is the problem that when you invoke the editor with file name arguments (i.e. files to load when the editor starts), and the files are around 64K, often only the first screenful is loaded. This can lead to heart stoppage the first few times you see it (when you think that the other 60K of your program has been deleted), but it is mostly harmless. Simply reloading the file, this time from within the editor, fixes the problem. Another bug that I’ve seen, though only very rarely, is the insertion of inverse @’s into the source. Luckily, these are few and far between, and it is usually quite obvious what letter was replaced. The other replacement of text that is completely harmless is in Min/Shell. Min/Shell has a command history similar to APW/ORCA’s (pressing the up and down arrow keys will scroll through previous commands). Sometimes, however, instead of the previous command, what looks like a tab ruler (===!====! etc.) is printed on the command line. Annoying, but not fatal.

Also annoying are “features” that were not very well thought out, and an index that is somewhat lacking in its coverage (the things I’ve had to look up were never in the index, although skimming the manual showed me that the features were present in the program). As an example of both of these problems, consider the simple task of changing the language type of a newly created file. In MAX/Edit, all new files are assigned a default language type (ASM65816 in my case, though you can change this to whatever language you like). But, if this isn’t the language type you want, you have to go into the editor

options box (Command-control-o) to change it. Not only is this somewhat counter-intuitive—as the other options in the box are things such as “set scroll speed” and “set mouse return convention” that deal with attributes global to the editor and not specific to any one workspace—it’s also not in the index! This option would be more appropriate in the Advanced Desktop, a section of MAX/Edit that gives you information on all of the workspaces and allows you to switch between them as well as perform operations such as deleting workspaces and manipulating the various clipboards. Also, sometimes when quitting the editor, you will not be prompted to save files that have not been saved. As long as you’re good about saving your source (and most programmers I know are compulsive about it) or have turned on the auto-save feature, you won’t have a problem.

Other annoyance bugs are described in the **Read.Me** file on the MAX/Edit disk. These are classified under the clever euphemism “anomalies” and consist of such things as crashing on exit from MAX/Edit when an internal APW/ORCA command has been terminated with Command-., infinite looping when a macro that has not been defined is executed, and crashing when MAX/Edit is re-invoked from Min/Shell.

For all of the features and power of MAX/Edit, the number of bugs is quite small, and the fact that, for the most part, they are only annoyance bugs and don’t lead to massive destruction of source code makes them almost excusable. MAX/Edit is a programmer’s utility—it’ll never replace AppleWorks GS or EGOed for word processing. However, if you’re all serious about programming, and if you can spare the fifty dollars, MAX/Edit is probably a very worthwhile investment. Its abundance of features can’t be beat, and, after you get past the somewhat steep learning curve, it will almost definitely increase your productivity. **GS+**

Software Of The Month Club

1-year IIGS Subscription: \$131.15
Not copy-protected

SMC International Publishers, Inc.
2180 Las Palmas Drive
Carlsbad, CA 92009
Customer service: (619) 931-8111 ext 509
Technical service: (619) 931-8111 ext 515

Reviewed by Charles W. Snyder, Jr.

How many times have you flipped through your *A+inCider* or *Nibble* magazines and seen the little eye-catching ad toward the back that says, "FREE - 15 disks - FREE, PAY ONLY \$5.00 FOR SHIPPING?" Usually, I stay away from come-ons like this because there isn't anything free that's worth anything in this world today; however, 11 months ago, just after I purchased my IIGS, I was starving for software (now I have more than I need!) and said, "what the heck, what's \$5.00?" Now, 11 months later, I have spent a fair amount of money on the Software of the Month Club (SMC), have received loads of disks, and am happy to say that I am quite satisfied with them. SMC is in the business of exactly what their company name indicates: they compile disks of software—primarily freeware and shareware—and sell these disks to subscribers.

Is the Software of the Month Club for you? To help you decide, I'll describe their program, cover different subscription alternatives and costs, and discuss the advantages and disadvantages of each. In preparing this review, I spoke with David Jessop, one of the technical support people at SMC, who was most helpful.

David indicated that in order to run the software that SMC distributes, you must have a IIGS with a minimum of 1 MB RAM and System Software v5.0 or higher. The software that SMC distributes is obtained from a variety of sources including GENie, CompuServe, electronic bulletin boards and direct submissions by authors. They screen all of their software to determine if it is user friendly and potentially useful to subscribers. They discard the junk. About 60% of what they

receive are decent programs. From that 60%, they choose material for their disks.

There are two subscription alternatives for Apple II users. You elect either the Apple II program or the IIGS program. Needless to say, the Apple II subscription will work on almost all older Apple IIs (except the II and II+) and the Apple IIGS. For \$5.00 shipping and handling, you will receive 15 5.25-inch disks if you elect the Apple II alternative. If you elect the IIGS subscription, you will receive 6 3.5-inch disks for \$6.95 shipping and handling. If you elect to continue in SMC, you must remit \$19.95 to them for a 1-year subscription. During the year, you will receive 16 shipments—one for each month, called "Librarian's Selection"; and an extra shipment each quarter, called "Special Edition." For each shipment, you'll pay \$5.00 for the Apple II subscription (5 5.25-inch disks) or \$6.95 for the IIGS subscription (3 3.5-inch disks). This is in addition to your \$19.95 yearly subscription fee. For the year then, you'll pay a total of \$99.95 for the Apple II subscription or \$131.15 for the IIGS subscription. It is easiest if you set up a credit card account with SMC. You may cancel your subscription at any time, but you won't receive any of your \$19.95 back. Of course, you won't be charged for any disks after you cancel your subscription. You may elect to subscribe to both the Apple II and the IIGS, but I don't recommend it due to the amount of duplication of programs from the Apple II sets in the IIGS sets. Keep in mind that the subscription price does *not* include shareware fees to authors whose programs are included on these disks. Shareware fees are usually indicated on supporting documentation or in the "About..." selection under the Apple on the menu bar of shareware programs. Please support our shareware authors by remitting their fee if you use their programs. User support of these folks will encourage them to write more programs which we can certainly use in these days of diminishing new commercial Apple software!

After you are a registered subscriber, SMC will send you a set of disks each month that include both freeware and

shareware programs covering the usual range of utilities, education, business, graphics, and games software. For instance, in a recent IIGS offering (Vol 97GS), I received the following programs:

IIGS-Specific Files

As The Link Turns: Operation Bug: Arcade game—shoot Macintosh Computers, etc.

Audiozap v0.8A: Full-featured sound recording, playback, and editing system.

Fontasm v1.0: Font editor. Comes with an *excellent* 30-page documentation file.

Menutime NDA v1.0: NDA that puts a clock on your desktop.

Plasmalab: Scenarios of computer generated life—a game of sorts.

Periodic Tables: HyperStudio stack of all the elements.

Towers: Move disks from a center tower to outside towers.

SMC Back Issue Catalog: Covers IIGS subscription only.

Apple II Files

Appleworks Small Business Payroll: Collection of AW templates for calculations and posting of recurring payroll expenses for a small business.

Dogpaw: Utility for displaying/printing text files, from BASIC/ProDOS files to Appleworks.

ShrinkIt V3.0.3: Utility program for archiving files and disks.

Columnist: Creates columns for your document.

Grafe: Edits random-access TXT files under ProDOS 8.

Ping Pong: A "Breakout" clone.

Trig Tricks: Creates animated Sine and Cosine graphs and trig translations.

Science Trivia: Science trivia questions, and capabilities of creating your own questions.

FPAGES: A collection of text tools intended to provide a convenient method of obtaining a useful printout of your text and source files.

When I started my subscription in March, 1990, SMC only offered the Apple II subscription. Frankly, I was disappointed with many of the programs and was about to cancel my subscription. Some material consisted of old programs written in the early 1980s, and some programs were poor quality public domain and were of little use to me. This past summer, SMC began offering the IIGS subscription, to which I switched and found to be a vast improvement over the Apple II subscription. Most of the IIGS programs are recent and represent some of the best freeware and shareware available. David tells me that since he assumed

responsibility for the Apple II library at SMC this past November, he is succeeding in his efforts to insure that more current and better quality Apple II material is included in their disks. I must advise you that I have received some bad disks and programs that don't work. If you call SMC, they will replace bad disks free of charge and will send replacements via First-Class mail. Normal subscription sets are sent third-class and it takes approximately 4 to 6 weeks to get them.

I give the SMC technical support department (1-619-931-8111, ext 515) high marks. There is no toll-free number, but if you are located east of the Mountain Time Zone, you can at least take advantage of the evening phone rates. Their normal business hours are 9:00 a.m. to 4:00 p.m., Monday thru Friday, Pacific Time. You will talk to either Kevin Hunt or David. The few times I have called them, they have been quick to respond and will return your call if they cannot answer your questions on the spot.

Finally, should you subscribe to SMC? I'd say if you fall into one of these categories, SMC may be for you:

1. New computer users who want to build up an inexpensive software library.
2. Beginning and advanced users who are interested in looking at a good representation of available freeware and shareware.
3. Users who don't want to invest in a modem and a subscription to an online information service, but who want to enjoy some of the current online freeware and shareware. Remember, SMC does the screening for you. If they do a good job of selecting programs, we, as subscribers, will benefit.

My suggestion is to give SMC a try. Even if you don't like what you get for the introductory fee of \$6.95 or \$5.00, you can always recycle the disks! **GS+**

Super GS Award Maker

Program by Richard E. Dye

Retail price: \$49 (\$59 with backup disk)
Typical mail-order price: \$38
768K RAM Required

Orange Cherry Software
Box 390 Westchester Ave.
Pound Ridge, New York 10576-0390
(800) 672-6002
(914) 764-4104

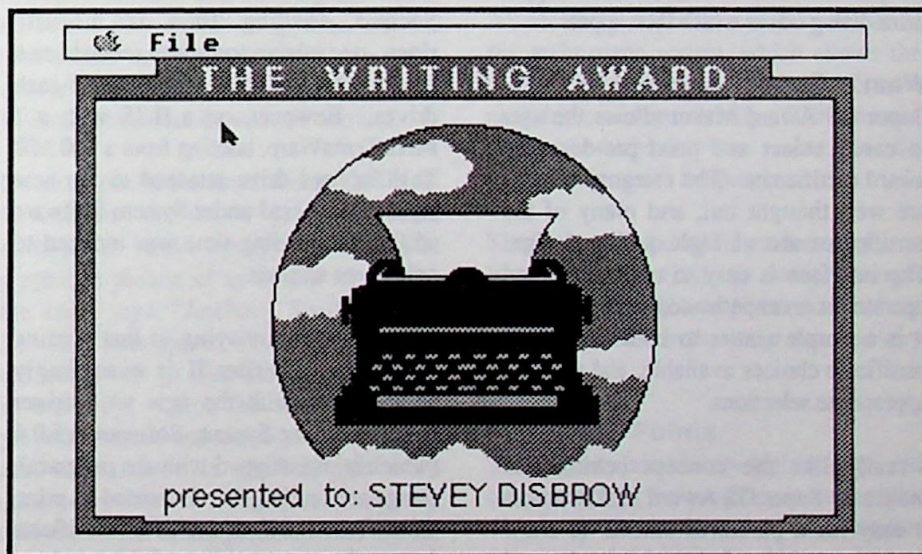
Reviewed by Greg Zimmerman

The Details!

Super GS Award Maker is another addition to the long line of IIGS-specific offerings from Orange Cherry Software.

It is a two-disk set, whose purpose is to allow the user to select pre-designed award certificates that can then be printed.

The program can be installed on a hard drive using instructions included in the short manual. Super GS Award Maker is not copy-protected, so backup copies can be made using the Finder or any reliable



disk copying program (just be sure to rename the copies the same as the originals). It will run properly on either a ROM 01 or ROM 03 IIGS, and is compatible with System Software v5.0.4.

According to customer service (and contrary to statements contained in the manual), Orange Cherry has a free lifetime disk replacement policy.

The tech support is good, and most importantly, the call is free. I found the people at Orange Cherry to be very helpful and knowledgeable about the program.

What's It Got?

Super GS Award Maker has 66 pre-designed certificates and 1 blank certificate intended to be printed and used as awards. The certificates are divided into the following 12 categories: Language Arts,

Science, Social Studies, Physical Education, Math, Achievement, Sports, Library, Art, Music & Drama, Holidays, and Miscellaneous.

Certificates are selected by choosing "Load Award" from the one pull-down menu choice, which gives rise to a screen of 6 boxes indicating six of the certificate category choices. A "more" button on the screen allows the user to see the second set of categories. To choose a category, the user clicks on one of the category boxes, which triggers a pop-up menu that lists the various awards in the selected category. The user then clicks on the specific choice of certificate, clicks on "load," and the choice then appears on the screen.

Depending on which certificate is selected, the user may then be able to enter text, such as the name of the recipient of the certificate or the date of the award.

Page setup and printing are done using the pull-down menu and standard Apple print-dialog boxes which then appear.

What's Good?

Super GS Award Maker allows the user to easily select and print pre-designed award certificates. The category choices are well thought out, and many of the certificates are of high quality design. The interface is easy to understand and operate for an experienced IIGS user, and it is a simple matter to both survey the certificate choices available, and to make appropriate selections.

I really like the concept behind this program. Super GS Award Maker makes it easy for a parent or teacher to show some appreciation for good behavior and accomplishments. All too often, kids only hear from a parent or teacher when they are doing something wrong. But kids need encouragement, and giving them some type of award or appreciation certificate is one good way to keep them motivated in the right direction.

What's Not To Like?

First, there are some minor complaints I have that center around the user interface.

For instance, once printing is started, there is no way (that I could find) to stop it, except of course to reboot the computer, or to turn off the printer. The Command-period choice is not available. Similarly, when selecting a certificate from the pop-up menus, double-clicking on the name of the award that is highlighted produces no results. Instead, the user must click on a separate "load" button. Also, once "Load Award" is selected from the pull-down menu and the six boxes appear, the user must select a category and then select an award. It is impossible to back out of the "Load Award" selection once it has been made. When I tried hitting escape to back out of this situation, I got a pop-up menu with a certificate choice, which crashed the program when it was selected. Another problem is that no letters are allowed when specifying the date to print on the certificate. Only numbers and slash marks can be used for the date. This "feature" is not documented in the manual, nor is the need to hit "return" after entering text.

Second, loading time is a little slow—one minute and seven seconds on a stock ROM 01 IIGS with two 3.5-inch drives. However, on a IIGS with a 7 MHz TransWarp, booting from a 100 MB Toshiba hard drive attached to the new Apple DMA card under System Software v5.0.4, the loading time was reduced to only seven seconds.

Last, and most annoying, is that printing on the ImageWriter II is exceedingly slow. Even with the new v4.1 driver installed under System Software v5.0.4 (which is not shipped with the program), things are *really* slow. I decided to print the "MEDAL OF MANNERS" certificate for my three-year-old son (which is kind of a joke, as he answers to the name "beast"), and it took eight minutes and 52 seconds to print the thing from the hard drive under System Software v5.0.4. Strangely (at least to me), the same certificate only took five and a half minutes when printed from the original disks launched directly, which uses the older v3.0 ImageWriter II driver. This is still slow, and it's only a half-page certificate. At least the program centered

his name on the line provided, which is more than the "Reading" certificate did when I printed it, in only four and a half minutes under System Software v5.0.4. By the way, the beast can't read, so he won't mind if his name isn't centered.

This slow printing put a damper on my enthusiasm for Super GS Award Maker. While I wait for a certificate to print out to give him as a reward for something he does right, I can send him to his room three or four different times for things he does wrong while waiting for the good news.

Well, What's The Wrap-Up?

Super GS Award Maker can be a useful program to parents and teachers that want to make use of award certificates on a regular basis. It's easy to use, and does exactly what it says it will.

However, it does have many user interface quirks, it doesn't ship with the latest System Software, it prints slowly, and it offers few options to customize the certificates.

It basically is 67 Apple Preferred Format drawings that the user can print. Of course they *could* be loaded into a good paint program for better customization and further processing.

So, while I can envision the usefulness of the program in particular circumstances, I cannot envision spending \$49 to effect its purchase. There just isn't that much here.

GS+

TALKING SPELLER II

Program by Richard E. Dye

Retail price: \$49 (\$59 with backup disk)

Typical mail-order price: \$38

512K RAM Required

Orange Cherry Software

Box 390 Westchester Ave.

Pound Ridge, New York 10576-0390

(800) 672-6002

(914) 764-4104

Reviewed by Greg Zimmerman

What It Is And What It Does

Talking Speller II is recommended by Orange Cherry for kids of ages eight to ten. It is part of the Talking Schoolhouse Series and is a sequel of sorts to another program in the series: Talking Speller.

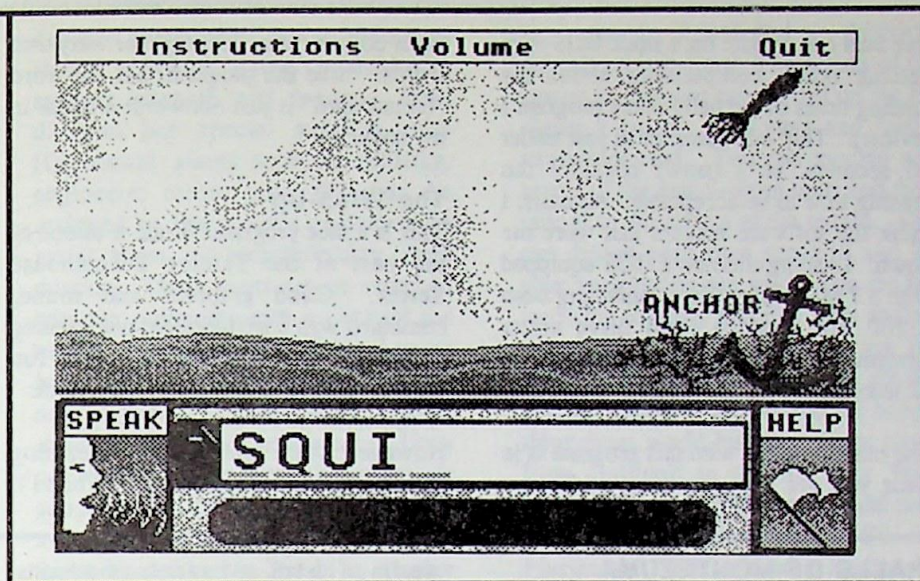
Talking Speller II comes on one disk, which is covered by Orange Cherry's free lifetime replacement policy. It is accompanied by a short, but complete and well-written manual, which describes how to use the program, offers ideas for activities in conjunction with the program.

The program is not copy-protected, may be copied using the Finder (the copy must be named the same as the original), and is hard drive installable. It will run on either a ROM 01 or ROM 03 IIGS, and is System Software v5.0.4 compatible.

The activities on Talking Speller II are broken down into two skill levels, Hard Words and Easy Words. From the main menu screen, the user may get instruction, adjust the volume, or select either the hard or easy skill level.

Each of the two skill levels is then broken down into four parts. Three of the parts are groups of words. These groups are Space Words, Music Words, and Marine words. Each of the two skill levels has the same three word group categories, with the categories containing different words appropriate to the particular skill level. The fourth part in each skill level is Word Scramble.

In the word groups, a screen graphic appears as a voice recites the word that the



user is to spell. While this is happening, the appropriate number of spaces for the letters of the word appear on the screen. There is also a box to click on to hear the word repeated as often as the user desires, as well as a help box to click on (if necessary) to find out the next letter that the user needs to enter. Once the user enters the right spelling, a voice tells the user that the answer is correct, and the program moves on to the next word. Any time that the user enters an incorrect letter, the voice comes on and says, "Please try again." Unlimited incorrect answers are allowed.

For example, in the Hard Words skill level in Marine Words, a graphic of an underwater scene appears, and in that graphic, a picture of an anchor appears as the voice says, "Anchor. Spell anchor." Six short lines appear on the screen where the letters will go after the user types them in correctly. Once the word anchor is properly entered and the voice tells the user that the answer is correct, the word disappears from the six short lines and reappears next to the picture of the anchor in the underwater graphic. Now a picture of a squid appears, five short lines appear, and the voice instructs the user to spell the word squid.

There are seven words per group, and three groups per skill level, for a total of 42 words to learn to spell.

In Word Scramble, a picture of one of the words from the previously described word groups, along with all the letters of the

word, appear on the screen. However, the letters are scrambled. As the user enters the correct letters in the proper order, the scrambled letters disappear as they are selected and reappear in the proper order a little lower on the screen. Once again there is a help button, with the same function as before, as well as a repeat button to hear the word again.

All screens contain the same menu bar as the main menu screen, which allows the user to get instructions, adjust the volume, or quit the current activity. If the current activity is the Main Menu screen, then the quit selection exits the program.

The recommended age level does appear to be appropriate. Children much younger than seven or eight will find many of the words contained in the program to be very difficult.

The Good Points

The graphics are of very high quality, and the concept of the program is very good. The way in which the user makes selections is easy to understand, and children will have no problem navigating their way through the program.

Most important, the reinforcement of the written word with the great graphics and the voice should make the learning and remembering process a lot easier for kids.

The Bad Stuff

With this review, I'm going to stop complaining about programs that load in

less than one minute on a stock IIGS (I'm getting bored complaining about the loading times of virtually every program I review). This one came in at just under 60 seconds, so I (now) consider the loading time to be acceptable. Actually, I think the software makers just wore me down. Loading time on a IIGS equipped with a 7 MHz TransWarp, launching from a 100 MB Toshiba Hard drive under System Software v5.0.4, was reduced to 12 seconds.

The main problem with this program is in what you get for your money. Thirty-

eight dollars (plus shipping) for forty-two words. Like the program, but the word "harpsichord" is just not worth a dollar in my book.

The Wrap-Up

This is a nice program. I rate it as one of the best of the Talking Schoolhouse Series. Good graphics and sound, combined with a well-thought-out teaching concept, make this an easy and fun program for kids to use and to learn from.

However, there isn't a lot of learning activity included for the money. While I'll

stop short of recommending that you avoid the program, I caution that you carefully consider the value you will be receiving for the price you are paying. Maybe a future version will allow the use of data disks with more words that may be purchased separately for about \$10-\$15.

I have four kids, all of whom are young enough to get some use out of the program either right now or at some time in the future. Yet even though I like the program, I would have a hard time justifying the purchase of Talking Speller II. GS+

HALLS OF MONTEZUMA

By Roger Keating, Ian Trout,
Andrew Taubman, Malcom Power,
Gregor Whiley

Retail price: \$44.95

Typical mail order price: \$29

Not copy-protected

Requires 1MB RAM and System Software v5.0.2 or later. Two 3.5-inch drives or hard drive recommended.

Strategic Studies Group
1747 Orleans Ct.
Walnut Creek, CA 94598
(415) 932-3019

Reviewed by Dave Adams

Halls of Montezuma (henceforth known as HOM in this review) is a military strategy game that is built around Strategic Studies Group's (SSG's) Battlefront Game System. HOM focuses on detailed simulations of famous battles in the history of the United States Marine Corps (Uncle Sam's Misguided Children—oops—I meant USMC).

In its earlier Apple II implementations, SSG's Battlefront Game System was a clunky and frustrating gaming system. Navigating through its many layers of menus was a daunting and confusing task. Each game came with a reference card that listed the various menus and how to get to them. It usually looked like a flow chart from hell. Once you got past the initial confusion, you found a remarkable and realistic game. A

wealth of detail and variety of options made each game expandable and constantly challenging. Transferring the Battlefront Gaming System over to the IIGS desktop has finally transformed it into an exceptionally easy and smooth gaming system. All commands are executed with the mouse through menu commands or icons. This is one of the simplest War/Strategy games that you will ever play. It is also one of the most powerful and adaptable games available for the IIGS.

Equipment Issue:

HOM comes on two 3.5-inch disks and is *not* copy-protected. It operates under System Software v5.0.2 and can be easily installed on a hard drive. The game also supports NDAs. Each game box contains a large map showing all the battlefield scenarios that are listed in the game. This allows you to examine the entire battle area while the game is in progress and plan your attacks. The game comes with two manuals, a Scenario Booklet and a Game Manual. The Scenario Booklet is a nice way to walk through the first turns in the first scenario. This acquaints you with the operations of the gaming system. It also gives background information for each of the scenarios in the game. HOM comes with eight scenarios: Mexico City, Belleau Wood, Iwo Jima, Okinawa (there are two scenarios for this battle), Pusan, Inchon, and Hue. Additional scenarios are available on America Online and GENie. These scenarios are taken from *Run5* magazine which is also published by

SSG. Each magazine contains additional scenarios and articles on designing and creating custom scenarios. The magazine is available with or without disks. The Game Manual is an overview of the Battlefront Game System and the Warplan design system. Warplan is the program that lets you modify and create scenarios. It includes a module called "Warpaint" that modifies or creates icons for units, terrain, and features. Both Warplan and Warpaint are available during the game and can be used to modify a game in progress. To really learn the nuts and bolts of a scenario design, you will spend a lot of time reading this particular manual.

"I Have To Follow Them... I'm Their Leader"

One of the nice features of the game is the startup screen. You are treated to a nice picture of the flag raising on Mt. Suribachi as the first few bars of the Marine's Hymn are played. Then you must open a scenario to begin game play. As the game starts, you have the opportunity to pick which side you will command. You can choose either side (the generic Axis or Allies or whichever of the two antagonists you prefer for the specific scenario). You may also enhance one of the sides to make game play easier or more difficult. You can have the computer control the other side or you can control it yourself (if you're really desperate to win). You can also play against another human if you desire. Once you have made your selection the game begins.

The Battlefront Game System allows you, as the commander, to issue orders to your subordinate commanders. Generally speaking, you are placed in the role of a Corps Commander. You give orders to Division Commanders and Regimental Commanders. They, in turn, pass them down to the battalions who will execute your orders. These orders are carried out to the best of the ability of the commands. Various factors affect the speed and effectiveness of the execution of the orders. Units with poor leadership, poor administration, poor supply, high fatigue, and low strength will not execute orders as well as units that have higher ratings. This feature also allows for the "Fog of War." Just because you gave a unit orders to seize an objective doesn't mean that it will occur, or occur in a timely manner.

Once orders are issued, you sit back and watch as combat is resolved and units move to the objectives that you assign them. Your better units will generally carry out their orders to the best of their abilities. Poor units will not do as well. If a unit has been assigned to seize an objective and it encounters the enemy, it will engage the enemy. All supporting units will rush to assist the engaged unit. Events such as this can really mess up your strategy if one little enemy unit is holding up the advance of three or four regiments. Each regiment has a variety of different orders available to it. A regiment can be in reserve (to rest up and replenish supplies), perform a variety of attacks (probe, preparatory, assault, exploit), defend, delay, or support another unit. Units can attack and move at night, but this has an adverse effect on their readiness for battle during the next day.

Scenarios last a certain amount of days—generally this is based on the actual time that the real battle took. Each day is broken into three daylight rounds and one night round. It is possible to conduct intensive assaults for extended periods, but it uses up the strength of a unit very quickly. It is best to limit assaults to once in a 24-hour period. Units should be rested at least once a day also. Units that have been

depleted by casualties and are exhausted should be sent to a reserve position to recover strength and losses. Each division has special support units (Divisional assets such as artillery, engineers, recon, etc.) that can be assigned to the various regiments to assist them in accomplishing their mission. These assets can be switched between regiments with a click of the mouse. Proper use of Divisional assets is the key to successful game play. Take care of them and withdraw them when attrition affects their performance. Units can receive support from other assets (air support, naval gunfire, etc..) by assigning support points to a unit. Support points can be limited by weather conditions and are generally available only during the day. Points can be assigned to different units as you see fit.

All of these actions are simplified by a simple point-and-click interface. Orders are assigned by clicking on the appropriate division, selecting a regiment, and then clicking on the appropriate icon for battle orders. Support points are assigned in a similar way. You can check on the status of your units by double-clicking on the regimental icon or moving the cursor over the unit and holding the mouse button down. After one or two rounds of playing, you have mastered the mechanics of the game and can then concentrate on strategy. Once you are finished giving orders, you watch the results and prepare for the next round of action. If you have ever played regular board wargames, then you can probably remember having to consult a confusing rulebook to figure out movement points and penalties. Then you would have to figure in the terrain bonuses or penalties and check your supply routes to ensure that they were clear. In HOM, all of that is done for you by the computer so that you can concentrate on fighting and winning the battle.

Adapt, Improve, Overcome...

The strongest feature that HOM has going for it is the ability to alter and design scenarios. The Scenario Booklet has a simple walkthrough to modify one of the scenarios included with the game.

For example, I included an Airborne Assault on Iwo Jima. The beauty of the system is that you can do this on the fly *during* a game. It is easy and fairly straightforward. You can modify any aspect of the game—from units and their ratings to the effects of the terrain. You can design your own scenario and let your imagination run wild. You are not restricted to using Marine Corps battles. Simply edit the units as you see fit. Many of the additional scenarios are not Marine Battles at all but various battles throughout world history. These range from Gallipoli to the Six Day War. There are many WWII scenarios and even one based on the Apple IIGS vs. Macintosh. You are only limited by your imagination or research time.

Remedial Training

Even though Halls of Montezuma has many excellent features, it does have some deficiencies. The manuals are weak on the actual set-up and design of new scenarios. The sections on terrain and using the tool palette leave a lot to be desired. Although many of the features can be figured out, using them correctly is a matter of guessing. There is mention of a *Run5* magazine article that gives complete descriptions of scenario design and creation, but that involves spending more money and waiting a few weeks until it arrives. The Scenario Booklet could also use a few illustrations concerning the order icons. As it stands there are no illustrations in the Scenario Booklet and the illustrations in the Game Manual are for a different computer. Although they are similar to the IIGS windows, there are differences. As it is, you must read in the text what each icon stands for and then count over "from left to right and from top to bottom." It shouldn't take that much to fix this—a simple quick reference card could take care of it.

Another feature that needs to be addressed is the time spent moving units after combat is completed. With custom scenarios involving large numbers of units, the time lag between the end of combat and the next orders phase can be as long as 4 minutes! I should qualify that statement by saying that this is not

the case with the scenarios that come with the game. The wait while the units move during the included scenarios is not that extreme. However I am playing HOM off of my 60 MB hard drive with a TransWarp installed. When I played the game off of 3.5-inch floppies on an unaccelerated ROM 03 the game slowed down considerably in this phase.

If you are one of those people that insists on total control of every unit in a game, then you are probably not going to like the Battlefield Gaming System. You cannot place every battalion wherever you want it and control its

actions. However I feel (from personal military experience) that the system is much more realistic than a "total control" type system. In the military, you issue the orders and your subordinates make it happen. This is reflected in the Battlefield Game System. It is simply far more realistic.

The Bottom Line, Sir... Mission Accomplished!

Halls of Montezuma is a very good Military/Strategy game. Although it is a bit slow in some areas, it is still a strong game and well worth purchasing. I have been having a great time playing the

many different scenarios and creating my own. I highly recommend this game to any person looking for a serious simulation and strategy game. If you have a modem, you can download about 20 different scenarios from America Online or GENie. Each issue of Run5 contains scenarios that you can create for continued game play. Hopefully, this game will develop a following like Mean 18 and we will soon have hundreds of public domain scenarios available. I look forward to SSG's next IIGS game. **GS+**

* GravenStein Apple IIgs Users Group *

Dedicated to serving IIGS users with information and education

P.O. BOX 751454 PETALUMA, CALIFORNIA 94975-1454

A Lot of Fun and Friendship!

GSAUG is a nonprofit volunteer group started in 1988.

Over 100 members enjoy the benefits of GSAUG. Some members are beginners, others are experts. But they all know the secret to making their Apple computer more valuable to themselves.

Members also have a lot of fun and make lots of friends while learning, sharing and growing.

Membership is only \$25 per year.

For more info, contact our BBS, or Don Forbes (voice) 707-224-5110, or Big Russ on America Online.

Bulletin Board System

707-538-8129, the club's electronic bulletin board, can get you through many jams with over 80 megabytes of software and information can be accessed at baud rates up to 2400.

The message areas are very active with general discussion and questions and answers in all areas and levels of Apple II applications.

Give it a try! Full access, however, is for GSAUG members only.

GSAUG members can buy hardware too! 20/30/40 MEG HD \$200/250/300. Octoram memory boards/w 2megs, \$120! It pays to be a GSAUG member.

Monthly meetings are where the excitement begins. On the third Saturday of each month, members of the GravenStein Apple IIgs User's Group meet for a day filled with activities.

Activities include presentations by national and local software and hardware vendors, special interest group (SIG) meetings, a public domain software library, raffles, and much, much more.

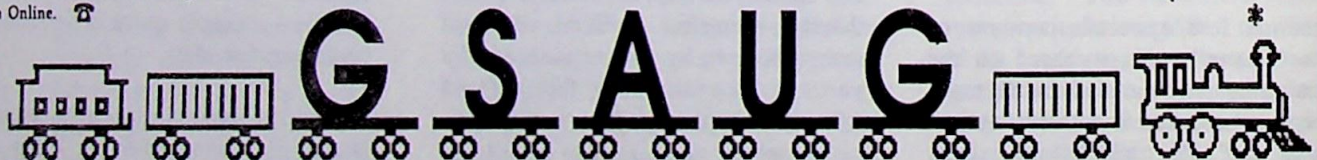
GSAUG meets at Grant School, 100 Grant Avenue, Petaluma. Meetings start at 10:00 a.m.

The Software Library GSAUG maintains is public domain and shareware software for the Apple IIgs system—over 200 disks!

We update and catalog the library monthly. We sell copies of disks to members during the monthly meetings, for a nominal fee. GSAUG also distributes Apple II System Software.

Club Newsletter

Our GravenStein travels from coast to coast and is full of candid reviews, how-to articles, rumors, club news, meeting information and a priceless list of volunteer experts on Apple II.



BUYING AD SPACE IN GS+ MAGAZINE

- 1/3 page - \$30
- 1/2 page - \$45
- 2/3 page - \$60
- 1 page (save 17%) - \$75
- 2 pages (save 22%) - \$140

Save an additional 20% by placing the same size ad in more than one issue!

Prices good through August 15, 1991.

Request your ad space today to guarantee these low prices!

DEADLINES:

- May-June 1991 (Volume 2, Number 5) April 15, 1991
- July-August 1991 (Volume 2, Number 6) June 15, 1991
- September-October 1991 (Volume 3, Number 1) August 15, 1991

All ads must be camera-ready copy. Payment must accompany ad. Make checks payable to **EGO Systems**, or call us (9 a.m. to 6 p.m. Monday thru Friday) to charge it on your credit card. For more information contact:

GS+ Magazine c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366
Voice phone: (615) 870-4960

If you need to ship your ad to us using a service other than the U.S. Postal Service, please call to make arrangements. If you wish to place an ad for a product we have not reviewed, we request that you include a review copy with your ad.

GS+ ORDERING INFORMATION

GS+ is published bi-monthly and sold for \$3.00 an issue for the magazine only, and \$8.00 an issue for the magazine + disk. But, if you sign up for a yearly subscription (that's six issues), you can get the magazine only for \$2.50 an issue, or the magazine + disk for \$6.00 an issue! To sign up, send this completed form (or a photocopy) along with a check or money order (payable to **EGO Systems**), or your credit card number, to:

GS+ Subscription Services
c/o **EGO Systems**
P.O. Box 15366
Chattanooga, TN 37415-0366

Name: _____ Phone: () _____
Address: _____
City: _____ State: _____ Zip: _____

- Subscription - Magazine only - \$15/year
- Subscription - Magazine + disk - \$36/year
- Sample issue - Magazine only - \$3
- Sample issue - Magazine + disk - \$8
- Check or money order enclosed
- Bill my MasterCard
- Bill my VISA Expires: ____/____/____
- Card No.: _____
- Signature: _____

If you prefer to use your credit card to order by phone, give us a call at (615) 870-4960, Monday through Friday, between 9 a.m. and 6 p.m. EST. All subscriptions will start with the next issue published. Please allow 2-8 weeks for delivery of first issue. Tennessee residents add 7.25% sales tax. Add \$1.50 an issue (\$9 for subscriptions) if you want First-Class delivery. Add \$1.50 an issue (\$9 for subscriptions) for foreign orders (this pays for air mail to Canada and Mexico or surface to all other countries). Add \$5 per issue (\$30 for subscriptions) for air mail to all other countries.

ATTENTION!!! La FRANCE arrive...

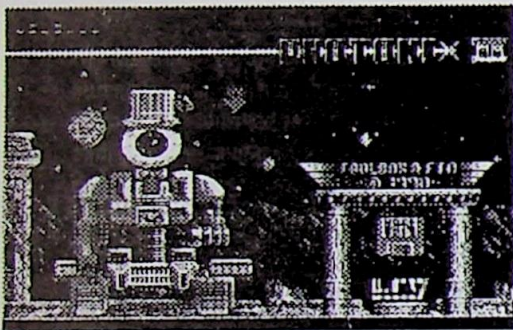
3 New and Amazing programs for your Apple IIGS

PHOTONIX II - BOUNCING BLUSTER II - SPACE SHARK

Now!!! for the American users of Apple IIGS

PHOTONIX II

The fastest copier for reproducing Apple IIGS and Macintosh diskettes.

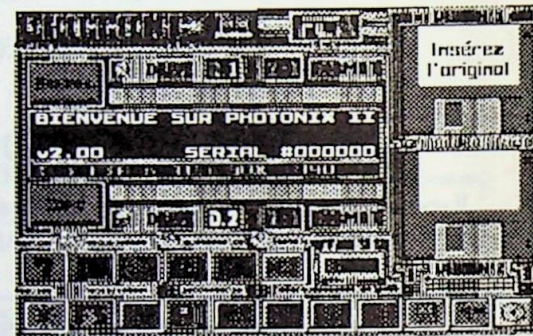


Graphical Interface

Mouse Support

Stereo Sound

*Runs on Apple IIGS
with Roms 01 or 03*



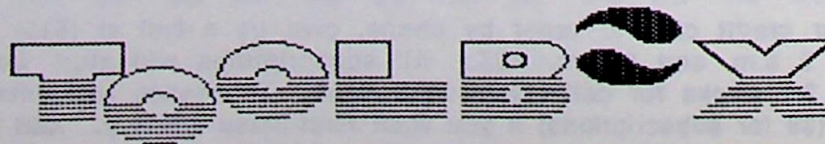
By Pierre Raynaud-Richard - Olivier Goguel - Denis Barritault - Daniel Bar - Olivier Bailly-Maitre

FUNCTIONS : Rapid copy - Mass copy - Anti-Virus - File Re-organization - Rapid Format
Directory Reading - Parameters saved in BRAM...

PHOTONIX II can be installed on a hard disk, it is reliable, and allows the verification of copies made.

PHOTONIX II is a program written in English and is delivered with user manual in English.

PRICE : \$45.00 + \$5.00 for shipping by airmail



6, Rue Henri Barbusse - 95100 Argenteuil - France

Fax : 0 11 33 1 39 47 44 08

BOUNCING BLUSTER II

Breakout style game with screen editor

GS/OS 5.0.2 (or later) compatible
Installable on Hard Disk

Terrific patterns, magic capsules,
psychedelic monsters. You will need a lot
of skill and dexterity to get to the end of
Bouncing Bluster II.

The game exploits all of the graphical and
musical capabilities of the Apple IIGS.

It is delivered with user's manual in English.

PRICE : \$59.00 + \$5.00 for shipping by airmail

SPACE SHARK

A Defender style arcade game for the IIGS

The object of the game is to destroy all of
the enemies who want to invade you. You
pilot a space ship around with a laser
cannon forward and aft and bombs. **Space
Shark** consists of 15 levels of play that
are more and more crazy... The game is
very fast and exploits all of the graphical
and sound capabilities of the Apple IIGS. It
can be played using a mouse or a joystick.

Space Shark is delivered with user's manual in English.

Free demo disk on request !

PRICE : \$49.00 + \$5.00 for shipping by airmail

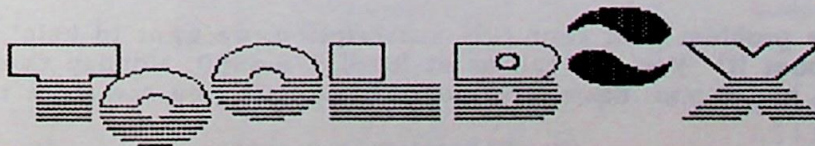
PHOTONIX II + BOUNCING BLUSTER II + SPACE SHARK

for only \$120.00 (+ \$5.00 for shipping by airmail)

All orders must be prepaid in U.S. funds drawn on an U.S. bank. We accept check in U.S. dollars.

Send us your address

you'll receive a free bonus disk
for your **APPLE IIGS Computer !!!**



New Products

Compiled by Steven W. Disbrow

The following product descriptions were taken from press releases that we have received here at *GS+* Magazine. As with all press releases, the products described therein may or may not actually exist. But, in an effort to scare up some business for those companies that have the *guts* to actually support the IIGS, we thought we should share them with you.

Such A Deal!

Roger Wagner Publishing, Inc. has just announced what might be the best HyperStudio (and IIGS) promotional campaign yet. For *absolutely nothing*, Roger Wagner Publishing, Inc. will loan copies of HyperStudio, a TouchWindow, a hand scanner, a laserdisc player and a video digitizer to any *school* that wants to conduct a multimedia workshop! Also included is a 50-page workbook describing how to conduct the 1- or 2-day workshop.

This "Multimedia Test Drive Kit" is, according to the press release, free for the asking to any school that wants it. Is it just us, or does this sound like an incredible way to show off what the IIGS (and HyperStudio) can do? So, if you are an educator, and you are sick of all the Macs and PCs that are showing up in your school, get on the phone (or jump into your word processor) and contact Della Smith at Roger Wagner Publishing, Inc. before everyone else does!

Della Smith
c/o Roger Wagner Publishing, Inc.
1050 Pioneer Way, Suite P
El Cajon, CA 92020
(800) 421-6526
(619) 442-0525 (FAX)

Look! Up On Your Screen! It's . . . SuperConvert!

After a *lengthy* beta-test period, SuperConvert—the commercial successor to SHRConvert—is finally about to ship. Normally, I would take apart the press-release and make all sorts of snide editorial comments, but I was a beta-tester for SuperConvert and can tell you that everything the following press release says is true:

"SuperConvert is *the* link between your IIGS and virtually any graphic! This new program converts formats from Apple II's, Macintosh, IBM, Atari ST, Amiga, Commodore 64/128, and even computer independent electronic formats such as GIF.

It's easy to use. Just transfer the desired graphic onto a IIGS disk, via modem or other means, then convert it to super hi-res with SuperConvert. Converted images can be printed directly with SuperConvert with a variety of options, including the ability to print wall-size posters!

But SuperConvert is not just for converting formats from other computers! Use its powerful "Remap Image" command to easily convert a 320-mode image so it appears correctly in 640-mode programs. Or change a color image into a gray-scale or black-and-white picture.

SuperConvert can be used to produce color slides from any 320-mode graphic. This feature is perfect for teachers' classroom materials, game players' authentication, and anyone who wants crystal-clear screen shots. It even generates "font sample" pages, "font key" charts and can make any image appear as your "desktop background."

Special Exchange Offer! Owners of Roger Wagner's "The Graphics Exchange" can purchase SuperConvert directly from Seven Hills Software for only \$20 by including the original disk or original manual cover.

NOTE: Formerly SHRConvert, SuperConvert is **NOT** shareware! This commercial version is approximately three times larger than the last shareware version, with many additional features. SuperConvert is available as an update to previously registered owners of SHRConvert.

Suggested retail is only \$39.95
Requires an Apple IIGS with 1MB RAM
and at least one 3.5-inch disk drive."

Seven Hills Software
2310 Oxford Rd
Tallahassee, FL 32304
(800) 627-3836 or
(904) 575-0566 from 9-5 M-F EST
Online via AppleLink, America Online,
GENie ("SevenHills") or CompuServe
("72437,3165").

Attention Publishers!

We want to help you get the word out about your *IIGS-specific* products. Send us a press release and we'll chop it up to fit here. Send those press releases to:

America Online: GSPlusDiz

US-Mail:
GS+ New Products
P.O. Box 15366
Chattanooga, TN 37415-0366 **GS+**

HAVING PROBLEMS?

If you are having a problem with your *GS+* subscription, we want to help! But we can't help if we don't know about it! You can call us at (615) 870-4960, Monday through Friday between 9 a.m. and 6 p.m. Eastern Standard Time. Or, you can write to us at:

GS+ Subscription Services
P. O. Box 15366
Chattanooga, TN 37415-0366

Feedback

How did you first hear about *GS+* Magazine?

Are you a subscriber to *GS+* or was this a sample issue?

If you are a subscriber, do you receive the Magazine and Disk? If you do not receive the disk, why not?

How would you describe your level of computer experience?

- babe in the woods
- novice
- I get by
- fairly proficient
- experienced
- power-user
- digital deity

Are you a member of a user group?

- Yes
- No

Please tell us a little about your IIGS system. Do you have a:

- Hard drive
- Modem
- Dot Matrix Printer
- Laser Printer
- 3.5-inch drive
- 5.25-inch drive
- Scanner
- Digitizer
- Accelerator card
- Fan
- Extended Keyboard
- Anything else?

How much memory do you have in your IIGS?

Is your IIGS part of an AppleTalk network?

- Yes
- No

Do you have a SCSI card in your IIGS?

- Yes Which one?
- No

How would you rate the following articles from this issue of *GS+*?

Dave Hecker Interview

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

Working With The Toolbox

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

How would you rate the following programs from this issue of *GS+*?

EGOed v1.33

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

OSLib

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

Quick NDA

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

The New Order

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

Transfusion v1.1.1

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

How would you rate the reviews in *GS+* Magazine?

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

How would you rate *GS+ Magazine*?

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

How would you rate the *GS+ Disk*?

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

How would you rate *GS+* overall?

- 1 (poor)
- 2
- 3
- 4
- 5 (excellent)

We want to have the best, most reliable advertisers in the business. Who would you recommend that we try to get? Why?

If there was one thing you could change about *GS+*, what would it be?

Do you think you will renew your *GS+* subscription? If not, please tell us why.

Anything else you want to say? Feel free to add additional sheets.

Send this sheet (or a photocopy) to:

GS+ Feedback
P. O. Box 15366
Chattanooga TN, 37415-0366

GS+



☐ Pro Series Drives • Pro R45 Removable • RamFAST • GS Memory Board

TMS Peripherals offers a complete line of SCSI hard drives for Apple computers, from 40MB's to 210MB's, at incredible savings. But we're not just about great prices, we're about excellent service too! Each factory fresh TMS drive is completely tested and verified by our technicians before it is shipped. Nevertheless, should you experience a problem with your drive (within the first 30 days) TMS will ship you a replacement and pick up the suspect drive—at no cost to you, leaving you with virtually no down time! After this grace period, your TMS Peripherals 2 Year Warranty becomes effective. In the unlikely event that you experience problems, call our Toll-Free Tech Support Line. If your problem persists after thorough troubleshooting by our technical staff—just ship us the suspect drive and we will send you a replacement unit within 48 hours!

Standard features include an International Auto-Switching Power Supply, Selectable Pushbutton SCSI ID Switch, 2 SCSI Ports (allowing Daisy Chaining), Shielded Power Supply, VCA Mechanism and all necessary hardware. Of course, each drive also comes with TMS Peripherals' Toll-Free Tech Support and our enthusiasm for each of our products.

For your convenience, we're open 8am-10pm (EST) 7 days a week. All products ship Federal Express. Visa, Master Card, American Express, Optima, C.O.D.'s and P.O.'s accepted.

**All prices subject to change.*

TMS Pro Series Drives*

Pro 40	\$349	12ms
Pro 105	499	12ms
Pro 120	759	10ms
Pro 170	829	10ms
Pro 210	899	10ms

TMS Pro R45 Removable Drive*

Pro R45, \$519 • 20ms, extra carts \$72

CV Technologies*

CV Technologies 0, 1, 2, 3, & 4 MB
GS-Memory Boards \$89, 142, 194, 246, & 299
RAM Chips: 256k 100ns (set of 8), \$25
1MB 100ns (set of 8), \$69
CV Technologies RamFAST/SCSI with 256k RAM,
\$174 (only \$169 when purchased with a drive!)

Zip™ Technology*

ZIPChipGS, Model 1500, 8 MHz/8k
Cache memory, \$189
ZIPChipGS Plus, Model 1525, 8 MHz/16k
Cache memory, DMA Compatible, \$239
ZIPGSX, Model 1600, 8 MHz/16k Cache memory,
DMA Compatible, \$269

TMS Peripherals

Authorized Quantum Dealer



1.800.626.MEGS
23123 SW 58th Avenue
Boca Raton, Florida 33428-2036
407.482.3821 • FAX: 407.451.9635