November
December
1994

Volume 6
Number 2

The *First* Apple IIGS® Magazine + Disk Publication!

Features

Programming the IIGS - Part 4:
Program Building Blocks

Mr. Priceguide Tells You Everything You Need to Know About Buying a Laser Printer for Your IIGS

Working With the Toolbox - Part 14:
The TextEdit Tool Set

Programs

Copy Icon - Copies Icon From the Finder Desktop
Elucidation - Edits Finder File Type Descriptor Files
Geeker - Edits the Finder's "Geek" Preferences
Ultimiter I - An Ultima I Game & Character Editor

Reviews

AUGE CD #1
The SimpleScript Workbook
Ultima I: The First Age of Darkness

Plus

The Hottest IIGS Rumors
Information on the Newest IIGS Products

Writer's Block

By Steven W. Disbrow

Bye, Michelle

Before I can talk about anything else, I have to say a special thank you and farewell, to our Production Manager, Michelle Ribaric. Michelle left us recently to take an office management position elsewhere in Chattanooga. Over the past year and a half, Michelle has helped us to slowly but surely gain ground on our production schedule until we actually got to the point where we were sending out magazines *before* the months on the cover had passed! She also became very good at juggling advertisers and keeping all of us here at the office on schedule. Best of all, she was a great sport and always seemed willing to put up with the B.S. of three guys trapped in a tiny basement. If, as many of you have been telling me, *GS+* Magazine has actually improved over the last year or so, it's in large part thanks to Michelle's hard work!

Hello, Bob

So, now that Michelle is off managing another office, who will keep us on schedule? Who will be our guiding light as we attempt to make up even more time in our (God help me) *never ending* quest to enlighten IIGS owners everywhere? Why, Bob, of course.

Yes, our own Robert "Bob" Ribaric (Yes, he is in fact related to Michelle, and to me for that matter. Nepotism is *very* inexpensive.), is stepping up to the plate to add all of Michelle's duties to his own. That means Bob will be handling layouts, advertising, and other day-to-day production duties, as well as keeping his old job of being our order fulfillment person. So, please join me in giving Bob a big hand, as he becomes even more entangled in my fevered self-employment dreams.

Can You Help?

Speaking of order fulfillment, over the past few months I've been doing some research into subscription and renewal trends for *GS+* Magazine, and I've noticed that for the first time ever, the number of renewals we get has been declining. With all of the positive feedback we've gotten about recent issues of the magazine, I'm hoping that this decline is due to folks moving on to other computer platforms and just letting their subscriptions lapse. But, whatever the reason, it means that we are beginning to lose ground in our efforts to expand our readership. So, I'd like to ask everyone out there for a little help. (Don't worry, it won't hurt!)

All I need is for everyone to convince just one *new* person to subscribe to *GS+* Magazine between now and the end of the year. Just *one*. If everyone out there can do this, we'll *double* our readership and *GS+* Magazine will last a lot longer.

The Spiders Were a Hit!

Last issue's full-color cover certainly was a hit! Thankfully, only a few people actually thought that the Earth had been taken over by giant spiders, and the bloodshed was kept to a minimum. (For the record, our cover-spider was a resident of our office building that had set up shop just outside Joe's window. Sadly, she was recently killed [we suspect the landlord] and her egg sac lost.)

With regard to future full-color covers, as I write this I don't know how many colors will be on this issue's cover, but if we can keep finding advertisers that are willing to cover the cost, we'll keep putting out full-color covers. (From what I understand, Sequential Systems has gotten a *lot* of response to their full-color ad in the last issue, so we might not have *that* much trouble.)

Procyon Products Now Available

In the last issue, I hinted that we might begin selling the products of yet another company. I'm happy to report that this has indeed come to pass. As of now, you can order any of Procyon's products directly from us. These products used to be carried exclusively by Resource Central, but when Resource Central decided to get out of the mail order business, Procyon called us up to see if we would be interested in selling them. The rest, as they say, is marketing history. The Procyon products that we now carry are: GNO/ME, Splat!, Switch-It!, and Pick 'n' Pile. For more information on these products and how to order them, check out "What's New?" elsewhere in this issue.

Christmas Sale!

Speaking of selling stuff, in the spirit of the Christmas season we are having a sale! All back issue magazines and disks are 25% off, and Addressed For Success is only $30! Complete details can be found in the "Back Issue Information" section, and in the Addressed For Success ad elsewhere in this issue. (However, I can tell you that the sale ends on December 22nd! So place your order now!)

Things Change

When I first started writing this column, I intended to use it to talk about everything *but* IIGS-related topics. However, as the years have gone by, I've found myself becoming more and more of a businessman and less and less of a philosopher. Since making money is more fun than philosophy, I suppose this was inevitable. Still, it points out a change in myself that I *really* don't like.

The worst part is, every time I see myself doing it, I try to think of some other great topic to write about, and I draw a blank. Well, that's not exactly true. There are lots of important things I'd like to write about (gun control, school prayer, Bosnia, etc.) but I find myself unable to write about them out of fear that I'll tick off a large number of you, and drive the magazine right into bankruptcy.

This is, I suppose, a side-effect of the Political Correctness hysteria that's swept America lately—people are afraid to speak up for fear that they will upset some people, create a back-lash, and have their life ruined. Or, as rap/heavy-metal musician Ice-T once said, "In America, you can say anything . . . but you can't say some things." Unfortunately, with each passing day this becomes more and more true. I don't know about you, but this makes me sick to my stomach.

So, what can we do about it? Well for starters, speak up! If you have an opinion on something, don't be afraid to let it be known. It might not do much for your popularity, but so what? If people stop doing the right thing, and only do the things that are popular, we'll all end up being a bunch of good for nothing politicians.

Well, what do you know! I managed to write about something important after all!

Diz

P.S.

Since I've got a bit more space, I'd like to take a moment to make another request. Specifically, when you send in a *hand-written* feedback form, problem form, or letter, please take the time to *print* what it is you want to say. Just in the past week, we've received two or three very long and apparently thoughtful letters that we simply can't read because the handwriting was absolutely inscrutable. This is very frustrating, because it seems that the people that wrote these letters had something important to say, but I'll probably never know what it was.

CONTENTS

FEATURES

PROGRAMS

REVIEWS

All products marked with a bullet (•) were review copies provided by the publisher.

DEPARTMENTS

ADVERTISERS

GS+ Magazine

November-December 1994
Volume 6, Number 2

Publisher, Editor
STEVEN W. "DIZ" DISBROW

Technical Editor
JOSEF W. "GONZO" WANKERL

Production/Marketing Coordinator
ROBERT A. "BOB" RIBARIC

Landlord
DEL MONTIER

On The Cover

Last issue's color cover was nice . . . but this time around we get back to the basics and try to pack this issue with as much information as we can!

Subscription rates - Magazine only:
1/2 year (3 issues) - $15
1 year (6 issues) - $24

Subscription rates - Magazine w/Disk:
1/2 year (3 issues) - $20
1 year (6 issues) - $36

Tennessee residents add 7.75% sales tax.
Add $2.00 per issue if you want First-Class delivery.
Canadian and Mexican orders add $2.00 per issue.
Other foreign orders add $2.00 per issue for surface delivery or $5 per issue for Air Mail.

Send orders, ads, inquiries, and address changes to:
GS+ Magazine
P.O. Box 15366
Chattanooga, TN 37415-0366

Or phone Monday-Friday 9 am-5 pm Eastern Time:
(800) 662-3634 (orders only)
(615) 332-2087 (tech support)
(615) 332-2634 (fax)

If you have a submission for *GS+* Magazine, send it to:
GS+ Submissions
P.O. Box 15366
Chattanooga, TN 37415-0366

GS+ Magazine can also be contacted on these online services:
America Online: GSPlusDiz
Delphi: GSPlusDiz
GEnie: Diz
Internet: Diz@genie.geis.com

GS+ Magazine is published bimonthly by:
EGO Systems
9482 Daisy Dallas Road
Soddy Daisy, TN 37379
(DO *NOT* SEND MAIL TO THIS ADDRESS—USE FOR UPS AND OVERNIGHT DELIVERIES ONLY!)

GS+ Magazine is produced on the Apple IIGS using the following products:
EGOed - Text Editing
GraphicWriter III - Interior Page Layouts
AppleWorks GS - Cover Layouts
Proof pages are printed on an Apple LaserWriter IINT.
We use a Macintosh LC III as a file server because we have to.

Letters

Diz and Company,
Really loved seeing the color cover on your anniversary issue! Congrats on five years of work that has really meant a lot to us die hards in the IIGS world.

It was good to see Gerald Lester's review of "Quick Click Calc" from the Byte Works [in *GS+* V6.N1]. I have been really interested in a decent GS/OS spreadsheet program, since I find the one in AppleWorks GS totally inadequate for serious users. I still think the spreadsheet in AppleWorks Classic (3.0 or 4.0) with all the Time Out goodies (including T.O. Graph) is hard to beat for speed and reliability. It sounds like after a few of the bugs are worked out Quick Click Calc may be a program worth looking at. The price is sure right!

On another subject, have you ever given any thought to writing and selling a commercial version of EGOed, but adding all the bells and whistles to make it a viable word processor? Since the demise of AppleWorks GS 2.0 I think there would be a market for it. We haven't had a decent GS/OS stand alone word processor that was ever worth a darn, and I would gladly give up AWGS since that's the only module I use from it (bugs and all). . . .

Lastly, although "Your Money Matters" did not receive a really super review in *GS+* Magazine, I really like it! I've been using it for over two years now and have never experienced the problems your reviewer found. I do think the documentation could be better, but once you really get into the program it's quite logical the way everything works. It really has some amazing capabilities if you take the time to set them up. I think a lot of people were expecting it to be IIGS version of the Apple II Quicken, but this was really thought out to include the serious investor, not just someone looking to balance their check book. I really think it's the most useful program written for the IIGS!

Steve Schmidt
Berthoud, CO
Internet: Smitola@AOL.com

Thanks for your comments on the reviews of Quick Click Calc and Your Money Matters. While I don't personally agree with your assessment of Your Money Matters, you aren't the only person to write in to tell us how much they liked the program (and how far off-base we were in our review). If there is ever another major update to the program, we'll probably re-review it. In the meantime, I'm working on a review of another financial program called Financial GeniuS. Look for it in the next issue.

As for a commercial version of EGOed the answer is "Yes." I have indeed considered doing a stand-alone version of EGOed, but I scrapped those plans after doing some research into how well IIGS software actually sells. However, after our experience with selling Addressed For Success (it's been selling quite well), I'm seriously considering restarting the EGOed project. All I can say is, "stay tuned."

Diz

Hi Gize!
Concerning *Giant Spider+* (Great Scott! Next it'll be Gomer Says "Golly Sergeant") V6.N1, I would like to pass along a comment on the fine article by Antonio Gonzolez entitled "II Scary: Halloween Fun With Your IIGS." In the paragraph "Making Your Own Music" the author states that "The MIDI adapter will probably be the hardest thing to find. Most of the ones that were made specifically for the IIGS have disappeared, but, Apple still makes one that will work with the IIGS . . . ."

Please, please, please let your readers know that internal MIDI adapter *cards* are the adapters made for specific computers, and that *under no circumstances* should anyone buy the over-priced, under-valued Apple MIDI interface (in fact, anyone with the Apple MIDI interface should probably dump it on an unsuspecting Mac head). External Macintosh MIDI interfaces work perfectly with the IIGS with more features at less cost. For instance, the MacMan Macintosh MIDI interface, by MIDIman is all metal (not plastic), has one MIDI input and *three* MIDI outputs, comes with a serial cable that connects directly into the IIGS modem port (or printer port if you prefer), has a serial *throughput* switch that allows you to connect your modem to the MacMan, thus having access to either your MIDI or your modem by pushing a button instead of swapping cables, has cool red, green and yellow LED's that flash when MIDI data is being transmitted (which can aid in troubleshooting), costs less than the Apple interface and comes with a lifetime guarantee (and no, I am not in any way affiliated with the MIDIman company).

The author is correct in directing your readers to Sound Management which sells the MacMan (guess which interface I use) and other IIGS MIDI stuff.

Eric Heim
Warren, MI
Internet: E.HEIM3@genie.geis.com

Dear Diz,
I'll start this letter by telling you I enjoy your magazine and will continue to subscribe for as long as you publish it. Almost every issue has at least one article or feature that is useful or informative to me. I understand "you can't please all the people all the time," but it seems to me you try very hard to find the right balance between novice and professional level materials.

I am a professional programmer with close to twenty-years experience on text-based mainframe computers, but I am a relative novice when programming my IIGS, especially when it comes to the graphics-based desktop. This leads to many frustrating hours when I know what I want to do but not how to do it.

This brings me to the point of this letter, namely a suggestion for an article, or series of articles, for your magazine. Specifically, how does one go about debugging a desktop program? What is Nifty List and how do I use it? What about GSBug? The ORCA debugger? Splat? When I hear the "Boink!" and the system is locked-up by a crash into the monitor, what do I do to find out where the program was executing so I know where to look in the code to make changes?

If you don't feel this kind of an article would be of sufficient interest to your readers, perhaps you could put together a "suggested reading list" and publish that instead.

Actually, suggested reading lists could be compiled to cover a number of areas, either by subject or language. I know your prime focus on programming is "real" programming languages, but perhaps your readers could benefit from collections for HyperCard, Logo and HyperStudio, in addition to topics like control panels, NDAs and CDAs and languages like C, Pascal and Modula-2 . . . .

Skip Helbig
Elyria, OH
Internet: W.Helbig@genie.geis.com

That's a really good suggestion Skip! You'll be happy to know that Joe is already working on an article about Nifty List for our next issue. In the meantime, I hope you've been finding my "Programming the IIGS" series helpful in getting you started with your IIGS programming. You should also check out all the installments of Joe's "Working With the Toolbox" series that you can get your hands on.

As for where you can get the products you mentioned . . . . GSBug and the ORCA/Debugger are available from the Byte Works (GSBug is sold as part of the ORCA/M package and includes complete documentation). Splat! is available from us for $39.95 (see "What's New?" elsewhere in this issue for more information), and Nifty List is available on all of the major online services.

Diz

Dear *GS+*,
I am a recent subscriber to your magazine and I like it very much. I noticed that you have available a program called EGOed v2.0. Apparently this program can read and write word processing files in RTF (Rich Text Format.) I would very much like to have such a program. Please let me know which issue it was available in and how much you charge for that disk.

By the way, does EGOed lite support RTF. . . ?

Greg Grigorov
Casselberry, FL
Internet: G.Grigorov@genie.geis.com

EGOed v2.0 was published in **GS+** *V5.N3 and it does indeed load and save RTF files. (For those of you that don't know, RTF is a format used on the Mac, Windows and other computers. By loading and saving RTF files, EGOed v2.0 lets you share formatted text files with those computers, and all your font information remains intact!) For complete ordering and pricing information, see "***GS+** *Back Issue Information" elsewhere in this issue. Sorry, but EGOed lite does not support RTF at all.*

Diz

Dear *GS+*:
On page 20 of issue V6.N1 (September-October 1994), there is an article about a finder extension called Find Original. The picture of a IIGS screen shows it in the Extras menu along with a number of folders and applications (if I'm not mistaken). This is like the Macintosh system which has an Apple menu folder into which an alias can be put for easier access. Presently, I have three aliases which are on my Finder desktop. How is it possible for me to get these application aliases off my desktop and into the Extras menu? Or the Apple menu? Or wherever else I can put them?

Calvin Kim
Venice, CA
Internet: C.KIM10@genie.geis.com

The Finder extension shown in that screen shot is a program called Quick Folder that we published back in **GS+** *V3.N6. It doesn't create aliases, it merely lets you place commonly used folders, data files, and applications in the Extras menu so that they can be easily opened or launched. If you are interested in getting a copy, check out "***GS+** *Back Issue Information" elsewhere in this issue. (By the way, if you ever see something in one of our screen shots or in the back issue listings that you want more information on, please give us a call!)*

Diz

Diz:
. . . Is there a way to put a bootable copy of AppleWorks GS on *one* high density 3.5-inch disk and boot and run it, using the Apple SuperDrive card and an Apple SuperDive? This setup . . . would make AppleWorks GS more convenient without all the disk swapping. Of course I realize that for the price of the SuperDrive card and the SuperDrive you could have a hard drive which would make the other approach less valuable . . . .

Paul Leach
Cincinnati, OH
Internet: prleach@tso.uc.edu

Well Paul, according to the Finder, AppleWorks GS takes up almost 800K on disk, which means that you would be left with about 640K for your System Software on a 1.44MB High Density disk. I think this can be done (although you wouldn't have room for things like printer drivers and desk accessories), but frankly, it just isn't worth it. My advice is to bite the bullet and buy a hard drive. You won't regret it.

Diz

Dear Steven,
. . . Is the Second Sight board the achievement of the TurboRez GS project from RezTek? I hope to read a review about it in the next issue!

. . . I've found a fix for the bug in Finder 6.0.1's Magic Routing feature! If you set the *auxType* of the **System** folder of your boot disk to "$00000002" it will work as expected . . . !

Michaël Guitton
Bouguenais France

Thanks for the letter Michaël! I tried your fix for the Magic Routing bug and it works great! *Thanks for finding such an easy fix for this amazingly annoying bug! (By the way, this bug doesn't appear if you drag your* **System** *folder out onto the desktop. That's how the bug got past Apple . . . all of the System 6.0.1 beta testers had their* **System** *folders out on their desktops!)*

As for your question about the Second Sight card . . . it is in no way related to the TurboRez project. We'll have a review of the Second Sight card just as soon as we can get our greedy little hands on one!

Diz

Diz
In reply to the letter published in *GS+* "Letters" in V6.N1, from "Name Withheld," France, I would like to relay my limited experience with a product supplier. As you will have noticed, I am also a "Foreign" Apple IIGS owner. Following the advertisement for DiscQuest Encyclopedia by Sequential Systems in *GS+*, I finally decided to purchase the product. As a customer I feel I can rate Sequential Systems "AAA" as an international supplier, as the speed with which my order was processed and queries answered was excellent. I do not believe I could have received better service even if the supplier was in Australia . . . .

Brian Fiddes
Victoria, Australia

Diz:
. . . I have some good news for Ryan Lanctot. There exists a shareware program for the Macintosh to convert IBM TrueTypes to Mac TrueTypes and back. The program is called TT Converter and is written by Chris Reed. The shareware fee is $10 . . . . Here is the information to contact the author:

Chris Reed
3409 Clearview Dr.
San Angelo, TX 76904-8108
Internet: chrisreed@aol.com

. . . As for the comment that *GS+* advertisers are not used to dealing with foreign (European) customers, this is completely true. Every time I called one of your advertisers, they were without exception very surprised that they were being called from Europe. However, they were very friendly in dealing with us and made absolutely no problems about having to send something to Europe. As a note to your advertisers, I would like to ask them to include their FAX numbers since this is a much more practical way for us to contact them. Because there exists a six to nine hour time difference it is not always easy to telephone, especially when the only listed number is a 800 number, which is useless outside the US.

. . . In the article "A Graphic Discussion," a lot of file formats were mentioned. Would it not be possible to include the file format information on the disk when this information is available? Usually you do this when the information is in a Technical or File Type note . . . . And yes, I would be interested in a discussion of other formats like sound or music files . . . .

Jan Maes
Boom, Belgium

Thanks for the information on the TrueType font converter Jan!

As I mentioned in our last issue, I think that all *of us in the IIGS market could stand to improve our handling of foreign orders. I'm trying to improve that situation here in* GS+ *Magazine, and we'll encourage our advertisers to do likewise. Be sure to keep me posted on how we are actually doing, and feel free to write in with suggestions on how we can improve things!*

Finally, you are exactly right about the technical information on the graphics file formats. We should *have put those on the* GS+ *Disk. If we have room on this issue's disk, I'll try to get them on there. Be sure to check the* ***a.Read.Me*** *file on the disk for more information.*

Diz

Dear *GS+*
. . . Recently, I upgraded my IIGS by adding a 60MB hard drive, 2MB of RAM, and System 6.0.1. So far, I am in complete awe over the capabilities and power of System 6.0.1. but I am experiencing a few difficulties that I hope you can rectify.

1) Once I discovered how to create aliases, I immediately created a folder in which to store aliases of application start-up icons. However, I have been unsuccessful in my attempts to create aliases of BASIC applications . . . . Is there a way around this obstacle?

2) I am having trouble getting programs contained on more than one disk to boot properly from the hard drive. For example, The Immortal (which is not copy protected), locks up every time I try to load it from the hard drive. The program comes on two disks and I've tried keeping disk one in one folder and disk two in another folder. I've also tried putting disk two's contents in with disk one. What am I doing wrong?

3) . . . I see two 5.25-inch drive icons when I restart the system. I only have one 5.25-inch disk drive, what happened?

4) How come System 6.0.1. does not seem able to boot or recognize my Apple DOS 3.3 programs? Yes, I have installed the DOS 3.3 system translator . . . .

5) While using Copy II+, I created working copies (using the HD parameter) of copy protected programs such as Print Shop GS so that I may load then onto my hard drive. Most of my copying attempts have been successful, but a few have not. For example, when I used the copy files option in Copy II+ to place all the Silent Service files onto my hard drive, the program worked. However, as soon as I put all of the files into a folder to clean things up a bit, I get an "unable to find .SYSTEM file" error message. Why does it work one way and not the other?

6) As I began loading programs onto the hard drive, I noticed that Copy II+ (version 8.4 and 9.0) will not run properly when booted from the hard drive. I assumed that since System 6.0.1. and Copy II+ are both system utilities, they must be clashing. Is this diagnosis correct?

Christopher P. Barron
Stevensville, MI

1) You can only create aliases of folders and applications. BASIC programs are not applications, they are in fact data files that are processed by the BASIC.System application.

2) Older programs, especially games, will often not run from a hard drive. There are ways around this, but unfortunately the techniques for solving these problems vary from program to program. We might do an article on this problem in the near future, but for now, my advice is to hop on to an online service and ask the folks there if they have found work arounds for the programs you have. (There was recently a conversation on Delphi that dealt with this exact question.)

3) The Finder always shows two 5.25-inch drives, regardless of how many you have connected. This is annoying, but there is no reliable way for the Finder to sense how many drives you have connected, so it defaults to two (which is the most you can have connected).

4) The Apple DOS 3.3 File System Translator was not intended to let you run DOS 3.3 programs. *All it was intended to do was to give you a way to copy files from your old DOS 3.3 disks to your other disks.*

5) My answer to number 2 applies here as well. But, in specific, many older programs simply don't know about or like folders. So, if you try to run them from one, they won't.

6) Copy II+ needs a specific version of the ProDOS 8 software to work properly. In other words, it just won't work with the current version of ProDOS 8. To make sure it will work, you should always boot the disk that Copy II+ came on. (My advice: Stop using Copy II+ completely. It was a great program in its time, but it hasn't been updated in years and it doesn't like the new System Software at all!)

Diz

If you have a question, comment, or criticism about *GS+* Magazine, we want to hear it! Due to space limitations, letters may have to be edited and we can not answer every letter here in *GS+* Magazine.

If you want a personal reply, please include an e-mail address (preferred), a daytime phone number (and the best time to call), or enclose a self-addressed, stamped envelope with your letter.

Please address all letters to:

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

Or you can e-mail us at:

GEnie: Diz
Delphi: GSPlusDiz
Internet: Diz@genie.geis.com

Or, you can FAX your letters to us at (615) 332-2634. *GS+*

FAXination™

- Allows printing to an external fax mode from any standard GS/OS-based application!
- Faxes graphics and text with ease!
- Support for Pointless™ TrueType™ fonts!
- Use the Deferred Send feature to send one or more documents to the same fax station with a single call!
- Use the Deferred Send feature to create a fax document when RAM is low, and then send it later when more RAM is available!
- Phone Book feature addresses faxes and fills in cover sheets automatically!
- Answers and receives fax calls automatically, or only by manual command!
- Maintains a log of faxes sent and received!
- Use Send or Receive Logs to view cover information the actual document, print the document, or change the address and forward it to another person!
- Automatically supports Class-1, Class-2, and Class-2.0 fax modems!
- No knowledge of modems needed! Automatically senses which port and which modem is attached! Simply plug in the modem, install the software, and reboot!
- Optional Call Progress window allows you to see what is happening while sending or receiving a FAX!
- FAXination is a CDEV with an NDA interface to provide quick access to FAXination controls.
- PrintPicker NDA allows switching between the FAXination printer and your printer with ease (GS/OS System 6.0, or higher only)!
- Special desktop application allows even users with less than 2 Meg RAM to send Deferred FAXes!

FAXination requires GS/OS v5.04, or higher, 1½ MB RAM, and a hard disk drive. System 6.0, or higher, and 2 MB RAM is highly recommended. ***FAXination*** supports only external FAX/Modems. Shipping the first week of December 1994. Taking advanced orders on ***FAXination.***

	Before December 15, 1994	After December 15, 1994
FAXination	**$49.95**	($79.95)
FAXination with 14,400 external FAX\Modem	**$149.95**	($189.95)
Tracer Sanction & Mind Shadow (combo)	**$29.95**	($39.95)
Ultima I	**$29.95**	($39.95)
Quickie 3.2 Hand Scanner	**$89.95**	($149.95)
Quickie 3.2 Hand Scanner with Inwords OCR	**$99.95**	($249.95)
Quickie-C (Adaptor)	**$99.95**	($129.95)
Quickie 3.2/Quickie-C Bundle	**$189.95**	($259.95)
Quickie 3.2 (Update from 3.1)	$24.95	

Call 1-800-777-7344

Order direct from Vitesse and save $$

Vitesse Inc., P.O.Box 929, La Puente, Ca. 91747-0929

(818) 813-1270 • FAX (818) 813-1273

This is a limited time offer, prices subject to change without notice.

Fast, Smooth, Uncomplicated

Programming the IIGS Part 4: Program Building Blocks

By Steven W. Disbrow

At the end of our last installment (in *GS+* V5.N6) I said that this time around I'd be giving you a sample application shell along with this article. However, after re-reading parts one through three of this series, I noticed that I've not yet discussed the actual user interface elements that make a IIGS program a IIGS program. So, I'm going to back up a bit and discuss these IIGS application elements before moving on to a completed program. The point of this article will be to try and help you understand these building blocks so that you can make the mental link between the source code that you write and the application that you eventually see on the screen.

Basic Concepts

One point that *GS+* Magazine has been beating to death over the last few years is that, while the IIGS can indeed run older Apple II software, it can and should be considered as its own computing platform. There are many things that make the IIGS stand apart from other computers, and to write a good IIGS program, you have to have an understanding of all the things that make the IIGS unique. In past installments of this series, we've discussed two of these elements: GS/OS and the IIGS Toolbox. In this installment, we'll discuss a few more things: The IIGS desktop metaphor (menus, windows, and controls), and the use of resources.

The Desktop Metaphor

Also known as "the Apple Desktop Interface," "the Mac interface" and even, "the damn desktop," the desktop metaphor that is used by true IIGS software is probably one of the most loved, hated, misused and misunderstood elements of the IIGS. For years after the introduction of the IIGS, users and programmers alike treated the desktop like Dr. Seuss's green eggs and ham—they heard about it, thought they might like it, but were afraid to try it. Because of this, and the shrinking of the IIGS market, there are very few really good desktop applications available for the IIGS. (In my opinion, this is not because the desktop metaphor is hard to program, it's just that a lot of the folks originally programming the IIGS were old IIe and II+ programmers and they just couldn't be bothered to learn that newfangled desktop. The ironic thing is that most of these same folks ended up programming the Mac or Windows. But, that's another article entirely.) Fortunately for the IIGS, others were willing and able to meet the challenge, and we do have quite a few excellent applications that we can use. So now, let's get back to the issue at hand—the desktop and how it affects you as a programmer.

The "desktop metaphor" is an abstract concept that describes the basic way that you interact with your IIGS. When working under the desktop metaphor, the entire IIGS screen represents a virtual desktop that your work is arranged on. When you open a document (i.e. a file), it is placed on your desktop so that you can work with it. When you close a document, it disappears from your desktop so that it is out of the way. The desktop itself has a finite area, and there are ways to cover up or expose parts of the desktop, just as if you had dumped a bunch of papers on top of it.

And that, boys and girls, is about all that the IIGS desktop metaphor has to do with a real desktop. Just about everything else found on the IIGS desktop is a interface element invented (or stolen) by Apple to round out the interface. So, let's look at these elements and how they relate to the desktop.

The Menu Bar

The menu bar is probably the most familiar part of the IIGS desktop. It sits at the top of the screen (i.e. at the top of the desktop), and allows the user to make selections that tell the active program what to do next. The menu bar itself is made up of one or more individual *menus* that contain the individual choices that will be available to the user. The names of these menus appear in the menu bar and give the user an overview of the types of choices each menu contains. The individual choices in each menu are called *menu items.*

For example, all properly written desktop programs should have a menu bar with at least three menus in it: The Apple menu, the File menu and the Edit menu. The Apple menu (which, by the way, is represented by an apple icon and not the word "Apple") should contain menu items that give the user information about the application, as well as all of the new desk accessories (NDAs) the user has installed. The File menu should contain items that allow the user to work with the files (i.e. documents) that the application creates, as well as allowing the user to quit the application. The Edit menu should contain the four standard edit items (Undo, Cut, Copy, Paste and Clear) and any additional editing items that make sense for the application.

Beyond these three menus, the menu bar should contain whatever menus make sense for the application. For example, a word processor's menu bar would probably also contain Font, Size and Style menus to allow the user to change the appearance of the text in their documents.

Setting up Your Menus

When you are setting up the menu bar for your program, the first thing you need to do is take some time to think about exactly what your program will actually *do*. Next, you need to decide how you want the user to tell your program to do these things. Of course, all this menu planning should probably be done at the same time you are doing the rest of the up-front planning for your program. (The amount of planning and preparation for any program varies from programmer to programmer, but for your first couple of IIGS programs, I strongly suggest that you completely plan out your menus and the other elements of your program's interface, *before* you start writing code.)

After you have the basic design and contents of your menus decided on, you should define them using whatever method you feel most comfortable with. Basically, there are two ways to do this: Hand code the definitions yourself, or use an interface editor like Genesys or Design Master to create them.

For your first few programs I strongly recommend that you *not* hand code your menus (or any other of your interface elements). The reason is that creating these things by hand is a tedious and time consuming process with a *lot* of opportunities for creating problems that can keep your programs from working correctly. If you use a interface editor though, you can be pretty certain that your interface elements have been created correctly and should work. Besides, with most interface editors, you can easily save the things you create in source code format, which you can later study, reuse and customize. (This is the way that I personally work. I create the basic interface elements of my programs in

Genesys, save them out as Rez code for the resource compiler, and then fine tune the Rez code to get exactly what I want.)

Using Your Menus

Once you have your menus designed and "coded," you need to be able to interact with them from inside your program. This can basically be broken into three steps: Creating your on-screen menus from the definitions you've made, responding to menu selections the user makes, and disposing of the menus you create when your program is through.

Creating your menus on screen is pretty straightforward. You just make the `NewMenuBar2` tool call and pass it a reference to your menu bar definition. `NewMenuBar2` looks at the information contained in that definition, and uses it to create your menu bar, menus and menu items on the screen. (Note that a "reference" to your menu bar definition [and the other types of definitions we will be discussing] can be one of three things: A handle to the definition, a pointer to the definition, or the resource ID of the definition. Along with the reference itself, you also tell `NewMenuBar2` which of these three types the reference is, and `NewMenuBar2` takes appropriate action. We'll be talking more about this in the section on resources, below. By the way, if you don't know what a "handle" or a "pointer" is, go back to part one of this series and start over.)

Responding to menu item selections isn't a lot tougher. In your main event loop, you will be told by the `TaskMaster` tool call that the user has selected a menu item, along with ID number of the item that was selected. (Don't worry, we'll talk about `TaskMaster` in a future article; for now, all you need to know is that `TaskMaster` is a tool call that makes the job of managing the desktop interface *much* easier.) You can then branch to the appropriate procedures and functions in your program that handle that menu item.

Disposing of your menus at the end of your program is simplest of all: You simply shut down the Menu Manager and it takes care of the rest. (We discussed shutting down tools in our last installment, so go back and re-read that article if you need to.)

Windows

In the real world, when you want to work with something, you just take it out and plop it down on your desktop. However, if you did that on the IIGS, you would quickly run out of desktop space. While you *can* place things directly on the IIGS desktop, it's not something that you really want to be doing. (The menu bar is drawn directly on the desktop, but that's about the only thing you would ever want to go directly on the desktop. Of course, this rule doesn't apply to the Finder, which draws icons [like the Trash icon] directly on the desktop.) Instead, you will want to draw and maintain your documents inside of windows.

The reason for this is that if you maintain your documents in windows, they will be "above" the desktop, and the user can move and rearrange them easily. This makes it very easy to accommodate a large number of documents within the limited area of the IIGS desktop. (Note that even if you are only working with one document, you should present it to the user in a window. In other words, don't even think about drawing your documents directly on the desktop.)

Another good reason to place a document in a window is that if you do, you can allow the user to work on a document that is larger than the size of the desktop (i.e. larger than the screen.) Imagine a word processor that only handles 25 lines of text . . . horrible, right? Well, that's about what you would be limited to if you restricted the user to just the visible area on the screen. But, if you place your documents in windows, you can give those windows some scroll bars that the user can use to view and work with different parts of the document. (We'll discuss scroll bars in a little while.) This is a very good thing to do.

Setting up a Window

As with menus, it's very important that you spend some time up front deciding exactly how you want your windows to look. In this particular case, I strongly recommend that you *always* use an interface editor like Genesys to design your windows, simply because windows are very hard to get "just right" without using one of these editors.

When you define your window, you'll specify lots of different things about it. Among these things are: The initial position and size of the window, whether or not it has any scroll bars, if it has a title bar, close box, zoom box, and what colors it uses.

The actual contents of a window can be almost anything you can imagine. But, unless you are willing to write the code to make your imaginings show up, you are limited to filling your window with standard IIGS controls (buttons, line edits, etc.) when you first design it. We'll be talking in detail about these controls later on, but for now, all you need to know is that you can place just about any one of these controls into the content area of a window. (If you want your window to contain something other than one of the standard controls, you will have to draw that yourself.)

Using Your Window

After you have the basic characteristics of your window defined, you use it in pretty much the same way you use a menu bar definition to create a menu bar. You pass your window definition information to the `NewWindow2` tool call, and it uses that information to create your window and draw it on the screen. Once the window has been created, you need to know how to actually interact with it. Again, the `TaskMaster` tool call is the key to this problem.

Each time through your application's main event loop, `TaskMaster` will report back to you any event (mouse click, key press, etc.) that has occurred, along with a pointer to the window that it occurred in. (The System Software and the Toolbox almost always refer to windows by using what is known as a "window pointer." This is a pointer to an internal record that the window manager uses to keep track of the window. When you create a window by using the `NewWindow2` tool call, it will return the window pointer of the newly created window. You can then use this window pointer to compare against the window pointers that `TaskMaster` returns to you in the event loop. Of course, some of the events returned by `TaskMaster` won't have a window pointer associated with them, but we don't need to worry about those for this discussion.)

By using the information you get from `TaskMaster`, you can easily determine what happened in a window and take appropriate action. For example, if you are displaying a graphic in a window, and `TaskMaster` tells you that that window needs to be redrawn (for example, your window was partially covered by another window and then it's uncovered), you can respond to this event by redrawing the graphic. Here again, this is similar to the way you handle your menu bar: `TaskMaster` tells you what happened to a window, and you respond by calling the appropriate procedure or function in your application. The main thing to remember is that each window your application uses can be completely different from all your other windows, and the window pointers returned by `TaskMaster` are the key to keeping them sorted out. (Note that if you wish, you can also create your windows with a custom draw routine that

TaskMaster will automatically call for you when it's time to redraw your window. That way, you don't have to do hardly *anything* to maintain the appearance of your window—at least not in the main event loop.)

Of course, all this potential diversity means that getting rid of windows can be a bit more complex than just shutting down the Window Manager. For each of your windows, there may be a special data structure that you create to keep track of the state of the window's contents. If that's so, simply using the CloseWindow tool call won't magically make that information go away too. So, when you close your windows, you have to be certain that you clean up after them. Once you do that however, you can in fact get rid of a window simply by passing its window pointer to CloseWindow.

Controls

At several points in this article, I've referred to "controls" that you can put in windows. So, let's take a look at what these controls are, and how you use them.

Controls are, well, controls. They let the user interact with a program by putting a relatively familiar face on some potentially bizarre concepts. For example, instead of forcing a user to figure out and type in the amounts of red, green and blue needed to create a new color, a IIGS program can just present the user with three scroll bar controls that allow the user to mix these values right on the screen. Want to show the amount of time remaining while your program figures a loan amortization? Just put up a thermometer control and fill it up as the process progresses. Controls work well for simple interactions too: Want to get a "yes" or "no" response out of the user? Don't force them to type it, just give them "Yes" and "No" buttons to click on.

So, basically, controls are used by programs to do one of three things: Get information from the user, display information to the user, or allow the user to control some aspect of the program.

Using Controls

Now you might suspect that using controls from your programs is similar to using menus and windows. Right you are! As with these other interface elements, you first design and define your controls (again, I recommend using an interface editor when designing your first few controls), and then you create them by passing their definitions to the NewControl2 tool call. (Controls can also be created by the NewWindow2 call—if a window has a "control list," associated with it, NewWindow2 will call NewControl2 automatically to create each control in the window's control list.)

Once the control is in a window (every control has to be in a window—this is another great reason to use windows instead of working directly on the desktop), you interact with it by—you guessed it—waiting for TaskMaster to tell you when something has happened to it.

Whenever the user does something to affect a control, TaskMaster will let your application know by returning the handle to the affected control along with the part of the control that was "hit" by the user. (Controls can be made up of one or more parts, each of which has a unique part number. By looking at this part number, you can decide exactly how to respond to the users action. For example, if the user clicks on one of the arrows at either end of a scroll bar, you know that you need to scroll one line in the appropriate direction. If each arrow didn't have it's own part number, you would have a heck of a time figuring out which way to scroll.)

Another way users (and your program) interact with controls is by setting and getting their values. This value is stored in the ctlValue field of the control's record. (FYI, after they are created, you access run-time information about controls and menus via handles, but run-time information about a window is always accessed via a pointer to the information. I have no idea why Apple did things this way.) For example, a check box control can have one of two values: Off (zero), or on (non-zero). If the check box has a non-zero value, it's on, otherwise it's off. Other types of controls use the ctlValue field in similar ways, and some don't use it at all (for example, it makes no sense for a simple button). (Note that you should always use the appropriate tool call [either GetCtlValue or SetCtlValue] to access the ctlValue of a control. *Never* work directly with the ctlValue field unless you absolutely have to!)

After you finish using your controls, you really don't need to do anything special to get rid of them. When you close the window that a control is in, the CloseWindow tool automatically disposes of any controls in that window.

Types of Controls

So, what kinds of controls are available to your programs? Here is a quick run down:

Simple Buttons

This is absolutely the simplest control there is. These buttons simply appear on the screen with a text string inside them. When the user clicks on one of these buttons, TaskMaster reports a hit on it. What happens after that is entirely up to your program.

Check Boxes

Almost as simple as "simple buttons," check boxes are still pretty braindead, but incredibly useful. A check box is just that—a box with a descriptive string next to it. When the user clicks on one of these controls, TaskMaster flips its value for you and reports the hit. It's then up to you retrieve the current value of the check box and act accordingly.

Icon Buttons

Icon buttons are just like simple buttons, except instead of looking like a simple button, they are drawn in the shape of a specified icon. You could, for example, draw an icon that looks like a heart, and use it as an icon button. Other than that though, these are just like simple buttons.

Line Edit Controls

These are extremely useful controls that let the user enter and/or edit a single line of text. When you define one of these controls, you can specify how many characters the control will hold, and what the initial text will be. This can be extremely useful for, say, giving the user a default file name to edit. Another neat feature of line edit controls is that you can also use them for password entry. If you set up a line edit control for password entry, it doesn't display what the user types, it simply displays the same "password" character (like perhaps a "*") for each character typed.

List Controls

List controls are one of the most useful, and complicated, controls available on the IIGS. With a list control, you can show the user a list of items that they can make selections from. That may sound boring, but the items that you put in your list aren't restricted to just text strings. A list control can display text, a graphic, or both. As I said, however, lists are pretty complicated to put together and use . . . if you insist on doing it Apple's way. My advice is to use Joe's Miscellaneous Library list management routines. (The documentation for it is on your *GS+* Disk.) They make working with list controls *much* easier.

Picture Controls

Picture Controls exist for one reason only—to give you an easy way to display a picture in a window. (Note that I'm

talking about a QuickDraw II picture [see the QuickDraw II chapter of Toolbox Reference Volume 2 for details on QuickDraw II pictures] and *not* a screen shot or 3200 color picture.)

Pop-up Menu Controls

As the name implies, these are controls that contain a menu, similar to the menus that you use in a menu bar. In fact, you set up the menu used by one of these controls in almost exactly the same way that you would set up a menu for use in a menu bar. The menu is then tied to the control, and when you click on the control, the menu "pops up" and you can make a selection from it. (Here again, I strongly suggest that you set up this type of control using an interface editor.) When you need to know the item that was selected from the pop-up menu, you just look at the `ctlValue` field (using `GetCtlValue` of course) and it will tell you the item number that the user selected.

Radio Buttons

The neat thing about radio buttons is that only one of them can be "on" at any given time. So, you can create a group of radio buttons and use them to force the user to make a choice. But, what if there are several types of choices that you need the user to make? Well, that's solved by the fact that radio buttons must be created in *families* so that you can logically group them together. For example, if you have four radio buttons on the screen, two of them might be labeled "Cat," and "Dog," and the other two might be "Black," and "White." It makes no sense to force the user to choose from one of these four items, but if these radio buttons are logically grouped into the two families, "Pet Type" and "Pet Color," they can now choose one from each of the two separate families.

Scroll Bars

Scroll bars, while fairly difficult to use the first few times, are probably the most useful control available to the IIGS programmer. Scroll bars actually do everything a control can do: They give the user information, they allow the user to give the program information, and they allow the user to control the application. Scroll bars give the user information by showing them that there is (or isn't) more to be seen in a list or document. Scroll bars get information from the user when the user uses them to change what she wants the program to show in a list or document. And finally, scroll bars can be used as "dials" to allow the user to change program settings—like maybe the speed at which a paint program performs color cycling.

Size Box Control

Size box controls exist for one purpose only: To let the user know that "If you click on me and drag the mouse, the thing I'm attached to will change its size and shape." That's about all there is to them.

Static Text Control

As you might guess from the name, static text controls exist to display text that never changes. This is a strange thing, because they can in fact be used to display changing text, via the use of "substitution strings." It works something like this: You set up a static text control and you give it a string like this—"Hello *0! You seem like a real *1 guy!"—to display. Then (by the use of various and sundry pointers and tool calls) you can have the control substitute completely new strings where you see the "*0" and "*1" characters. The best part is that you can change these substitution strings whenever you wish and that you can have up to ten of them ("*0" to "*9") in each static text control.

TextEdit Controls

Conceptually, TextEdit controls are only one step removed from a line edit control. Instead of restricting you to a single line of plain text, TextEdit controls let you create and edit large blocks of styled text. Beyond that simple concept, however, is a *very* complex control. In fact they are so complex, that that's all I'm going to say about them for now.

Thermometer Controls

Thermometers are used as visual progress or status indicators. When you define a thermometer control, you simply give it a maximum value. Then, as your task progresses, you change the `ctlValue` field of the thermometer (using the appropriate tool call of course), and the thermometer fills up accordingly. When the `ctlValue` is 0, the thermometer is empty. When the `ctlValue` is equal to the maximum you defined, the thermometer is full.

Rectangle Controls

When you first hear about rectangle controls, you might think, "Geez, that's stupid." (Well, OK, *I* did.) But, wait a second, these things are actually pretty useful. They give you an easy way to draw rectangles on the screen, and since they are controls, the system will automatically redraw them for you when needed. Another side effect of them being controls is that `TaskMaster` is ready and willing to tell you when there is a hit on one of them, which means that you can "box off" areas of your window with invisible rectangles and you can tell when the user clicks in those places.

Custom Controls

If none of the above controls is exactly what you want, the IIGS Control Manager provides a means for you to create your own custom controls. You have to create it, draw it, determine when it's been hit and otherwise manage it yourself, but it's really not as difficult as it sounds. However, I'd strongly recommend that (for your first few IIGS programs at least) you should stick with the predefined controls.

Definitions?

Throughout this article, I've been talking about "defining" things and "definitions" that you pass to various tool calls to create menus, windows and controls. So, just what the heck have I been talking about?

Well, these definitions are in fact standard data structures that Apple has created to specify how these interface elements are to be created. When you "define" a new control or other interface element, all you are doing is "filling in" the values that these data structures require.

For example, the definition for a simple button control requires that you specify a rectangle (which tells where the button will appear in a window), an ID number (so that TaskMaster can properly identify the control for you), a "definition procedure" (which tells the system that this is indeed a simple button and not another type of control), two different sets of flag bits (which control the appearance of the button), and a reference to a string to draw inside the button. All of these things must appear in the definition in a certain order, and all of them work together to let the system create and maintain the control.

As I said before, you can create this definition in one of two ways: Hand code it in the language of your choice or use an interface editor to create it.

If you hand code your definitions, you simply find the appropriate definitions in your languages "header" files, create some variables of the appropriate type and initialize them with the appropriate values. This is a pain.

If, however, you use an interface editor like Genesys, you simply *draw* what it is you want to create on the screen, and the editor uses what you've drawn to create the appropriate source code or resources.

So, What Are Resources?

OK, I keep mentioning these "resources"—so what in the world are they?

Well, basically, resources are pre-initialized definitions that you keep in a file separate from your program. (Actually, resources are usually kept in the same file as your program, but they are kept in the resource fork of the file, which is, logically, a separate file.)

So, what does *that* mean?

Imagine that you are working on a program, and you've hand coded all your control, menu and window definitions so that they get compiled every time your program does. Your program works perfectly, but one control is slightly out of alignment with the others on the screen, and you want to change it so it looks better. With the way you have things set up, you have to load your code, edit the definition, *re-compile* the whole program and run it to check your changes.

If, however, that definition was in a resource, all you would have to do is run your resource editor, change the position of the control *on screen* and then save your changes. By not hard coding these definitions, you can save a *lot* of time on your programming projects.

Using Resources

OK, so once you have your definitions in resources, is there anything special you have to do to use them? No, not really. Unless you are writing a program that makes extensive use of resources, all you really have to do is make sure the Resource Manager is started (and if you use the `StartUpTools` tool call we discussed in a previous installment, it takes care of that for you) and then go about your business. The other tool calls I've mentioned (`NewMenuBar2`, `NewWindow2` and `NewControl2`) all use the Resource Manager when necessary to load the resources they need to get their jobs done.

Best of all, you don't even have to worry about cleaning up after these resources. When the system is finished with them (i.e. when quit your application), they will automatically be disposed of by the system software.

So, *Now* What?

Hopefully, this article has given you a pretty good grasp of the parts that make up the desktop interface and how you actually get all that stuff up on the screen. At this point, I recommend that you go and read Apple's *Human Interface Guidelines* book along with the *Toolbox Reference* chapters on the Control Manager, Menu Manager, Window Manager and Resource Manager. The *Interface Guidelines* book will tell you *why* you need to follow the desktop metaphor, and the chapters of the *Toolbox Reference* should fill in the technical blanks that I've left in this article. And, as always, if you have any questions, please don't hesitate to write.

In our next installment, we'll take a look at something that I've mentioned a *lot* in this article: `TaskMaster`. ***GS+***

Warranty Disclaimer And Copyrights

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

IN NO EVENT WILL EGO SYSTEMS' LICENSOR(S), AND THEIR DIRECTORS, OFFICERS, EMPLOYEES OR AGENTS (COLLECTIVELY EGO SYSTEMS' LICENSOR) 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 SOFTWARE EVEN IF EGO SYSTEMS' LICENSOR HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. BECAUSE SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF LIABILITY FOR CONSEQUENTIAL OR INCIDENTAL DAMAGES, THE ABOVE LIMITATIONS MAY NOT APPLY TO YOU. EGO Systems' Licensor'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.

And *furthermore*, some of the programs on this *GS+* Disk contain materials from the ORCA/C, ORCA/Pascal, and ORCA/Modula-2 Run-Time Libraries, copyright 1987-1994 by Byte Works, Inc. Used with permission. All other material is copyright 1994 by EGO Systems, unless otherwise noted. Purchasers of the *GS+* Disk may use the source code and/or libraries in their own programs, provided that the following notice appears somewhere in the program or in its documentation:

"Portions of this program were created using source code and/or libraries from *GS+* Magazine, and are copyright 1989-1994 by EGO Systems. Used with permission. For more information on *GS+* Magazine, write to:

***GS+* Magazine**
P. O. Box 15366
Chattanooga, TN 37415-0366"

Glossary

In each issue of *GS+* Magazine, we present a glossary of some of the more common terms in the IIGS world and some of the more uncommon terms that we use in each issue. If you have a term or bit of jargon that you would like to see explained, let us know and we'll try to get it in a future "Glossary" installment. Also, don't forget about the glossary that's in your IIGS owner's manual! At this point, it contains many more terms than the *GS+* Glossary!

Past installments of the *GS+* Glossary can be found on your *GS+* Disk in the plain ASCII text file, **Glossary** (see "How to Use Your *GS+* Disk" for more information). Entries marked with an "*" have appeared in previous installments of the *GS+* Glossary and are repeated here for our beginning readers or because they have relevance to topics discussed in this issue.

Alias

An "alias" is a small file that "points" to another file. So, for example, if you have an alias for a folder, and you double-click on the alias file in the Finder, the Finder looks inside the alias to determine which folder it points to, and then opens that folder. The best thing about aliases is that you can create them and then move them to another place on your hard disk (or even on a different disk). You can then use this alias as if it were the original file, without having to actually find the original file.

Unfortunately, on the IIGS, aliases only work inside the Finder, and they only work with applications, folders, and disks (shared or local). (In other words, you can't use aliases from any other program, and you can't make an alias for a document file.)

Clipboard

The Clipboard is a holding place for a scrap of information. (See "Scrap" below.) The standard Edit menu items are used to cut, copy, and paste scraps of information to and from the Clipboard. The Clipboard can be kept one of two places. First of all, it can be in a special area of memory that is used by the Scrap Manager (see below). Or, the Clipboard can be in a file on disk. (This file will contain exactly the same information that you would find in memory, if the Clipboard were in memory.) At any given time, the contents of the Clipboard can either be in memory or stored on disk. If the Clipboard is on disk, it is up to the current application to load it into memory when necessary. If it's in memory, it is up to the current application to save the Clipboard to disk before quitting to the previous application or shutting down the system.

D.P.I. *

"D.P.I." stands for "Dots Per Inch" and is a measure of the quality of output you can expect to get from a printer. The more dots per inch a printer can produce, the finer the resolution of the printouts. For example, the ImageWriter II is a 72 d.p.i. printer, the StyleWriter is a 360 d.p.i. printer, and most LaserWriters are 300 d.p.i. Some newer laser printers are capable of 600 to 800 d.p.i. By comparison, most books and magazines are printed on devices that produce 1,200 d.p.i. or more.

Internet

The Internet is a network of computers that covers the entire planet Earth. What we now know as the Internet was originally created by the United States military to link military installations and college research facilities. Over the years, the military's influence waned, and the net became hooked to more and more universities and the amount of information available on it grew. One side effect of this expansion has been that "normal" people have begun accessing the Internet in record numbers, and governments the world over have begun trying to figure out how to restrict access to it and to censor the information flowing through it. (Recently, it's become fashionable to refer to the Internet as the "Information Superhighway." But, as described by the media, this does not actually exist . . . yet.)

PPM

"PPM" stands for "Pages Per Minute." This is a measure of the number of pages a printer should be able to produce in one minute. In general, ink jet printers (like the Hewlett Packard DeskJet) are the slowest, dot matrix impact printers (like the ImageWriter) are faster, and laser printers (like the Apple LaserWriter) are the fastest. Note however that "official" PPM figures for a given printer are usually determined by the manufacturer, and have little bearing on real world usage. Even so, PPM figures are usually a fairly reliable indicator of the relative speed of a printer.

Publish & Subscribe

Publish & Subscribe is a means of automatically sharing and updating information between documents and applications. A *publisher* document (or application) makes information available in an *edition* file. A *subscriber* document (or application) then subscribes to that edition file and loads in the data that is inside the edition file. Then, each time the subscriber is opened, it can check the edition file to see if it has changed. If it has, the subscriber reloads the data from the edition to make sure it has the latest information. (A subscriber can actually check an edition file for changes whenever it wants [depending on the application], but it should always check when it is opened.)

Resolution Enhancement

Resolution enhancement is a technology used in printers to improve the final resolution of printed documents. In most cases (such as laser printers) this involves printing with various sizes of dots, or more precisely positioning standard-sized dots on the page.

Scrap

A "scrap" is a bit of information. For example, when you select some text in your editor, and then select the Copy item from the Edit menu, that creates a scrap containing the text you had selected. Scraps can be text, a sound, a picture, or just about anything that a programmer can think of. On the IIGS, scraps are stored on the Clipboard (see above). These scraps and the Clipboard are manipulated by the Scrap Manager (see below).

Scrap Manager

The Scrap Manager is a tool set that allows programmers to manipulate scraps of data and the IIGS Clipboard. (See "Scrap" and "Clipboard" above.)

Snail-Mail

"Snail-Mail" is the less than kind nickname that has been given not only to the American postal system, but to postal systems throughout the world. This nickname is based on the observation that, sometimes, physical mail seems to travel slower than a snail. (Example: We once mailed a magazine to ourselves as part of a regular subscription mailing. It showed up a month later.) Of course, this doesn't stop them from raising postal rates every few years. *GS+*

Apple IIGS® Owners, Are You Topless?

You are? Well for goodness sake, put a shirt on! But, not just *any* shirt . . . an official *GS+* Magazine T-shirt! Yes that's right, now you can have one of these fabulous shirts for your very own. Too good to be true you say? Well it *is* true! For only $11.50 you can have your very own *GS+* Magazine T-shirt! And, they come in a variety of exciting sizes, just look:

Marvelous Medium!
Luscious Large!!
Exciting Extra-Large!!!
Extra Exciting Extra-Extra Large!!!!

To get yours, send this completed form (or a photocopy) along with a check or money order (**payable to "EGO Systems"**), or your credit card number, to: EGO Systems, P.O. Box 15366, Chattanooga, TN 37415-0366.

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

# of Medium Shirts:..... # of Large Shirts:..... # of Extra-Large Shirts:..... # of XX-Large Shirts:.....
❑ Bill my MasterCard ❑ Bill my VISA ❑ Check or money order is enclosed (US funds only, payable to "EGO Systems")

Card No.:..... Expiration Date:...../.....
Signature:..... Total ($11.50/shirt):.....

The Exceptionally Fine Print

The shirt shown in the photo is one of our special tie-dye t-shirts. These shirts sell for $15 each. At this point, we only have one Medium tie-dye shirt left in stock. If you want one, be sure to include a note with your order and to include the correct amount. Note: When this last shirt has been sold, no more will be made!

If you prefer, you can use your credit card to order by phone. Give us a call at 1-800-662-3634 (foreign orders call 615-332-2087), Monday through Friday, between 9 a.m. and 5 p.m. Eastern Time. Or, you can FAX your order to us at 615-332-2634. *Tennessee residents add 7.75% sales tax.* First Class shipping to the United States, Canada and Mexico, or surface mail to all other countries is included in the $11.50 price. Add $3.50 per shirt for Air Mail to all countries other than Canada and Mexico.

Note: Unlike the knockoff shirts made in the sweatshops of Cupertino, official *GS+* Magazine t-shirts are 50% cotton and 50% polyester. *DO NOT* be fooled by imitations!

SEASONGS GREETINGS! GIVE YOUR IIGS SOME NEW SIGHT & SOUND WITH A SECOND SIGHT & SOUNDMEISTER CARD. - RAM & HARD DISK UPGRADES & MORE!

RAM CHIPS

RAM Chips for your....
Expanding your... Apple IIgs Memory Card, Æ GS-RAM, Æ RAMWorks, Super Expander & Compatibles
768K Exp. Set (24 41256).....$24.00
41256.....$1.25
Expanding Your GS Juice+, S&S ÆGS RAM + / II , CV Tech or Chinook 4000
41024 (1 Meg) $3.00
Bank of 8.....$24.00
Expanding Your Æ GS-RAM Ultra or RAMPak4 GS
44256 (256x4) $4.00
Bank of 8.....$32.00
Expanding Your OctoRAM, GS Sauce, Q-RAM GS, Pacemark 4 Meg RAM Card
1 Meg SIMM $24.00
AE GS RAM III 1 x 4 Zip.....$25.00

PC Transporter

101 Keyboard for PCT $29.00
PCT IBM Keyboard Cable.....$7.00
MS DOS 5.0 Package.....$49.00
3.5" 720K Ad-on Drive.....$49.00
4464 Zip /PCT Mem. Exp. .....$2. ea
TransDrive 5.25/3.5" Combo$89.00

Motherboards/Service Parts

IIgs Power Supply Exch $59.00
IIgs Power Supply New $79.00
IIgs Keyboard Exchange $45.00
IIgs Keyboard Refurb $55.00
Apple //e or //c Case.....$10.00
IIgs Case $25.00
ROM 01 Motherboard Exch..$149.00
ROM 03 Motherboard Exch..$199.00
Upgrade: ROM 01 to ROM 03
W/Working ROM 01.....$175.00
W/NonWorking ROM 01 .....$225.00

Refurbished Printers

ImageWriter I $129.00
ImageWriter I (Wide).....$99.95
ImageWriter II.....$199.00
ImageWriter II Sheet Feeder ..$79.00
9 Pin Parallel Printer $75.00
IW-I/II Ribbons.....6 for ..$12.00
Print Heads
IW-I (Exch Only).....$35.00
IW-II (Exch. Only) $69.00
Epson 9 Pin Heads (Most) $59.00
We rebuilt print heads

Accessories

IIgs Internal Fan.....$14.95
9 Pin Joystick , IIgs $12.95
Mouse Pad (Rubber/Staticfree)...$2.95

Cables & Switch Boxes

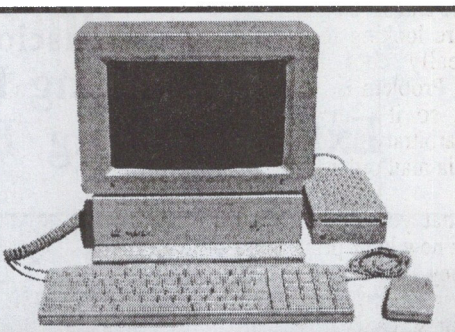
METAL, GBBS Sysops. HST/v.32bis hardware hndshk cables! (Specify)....$14.95
19 Pin Drive Converter.....$14.00
IIgs,e,c to ImageWriter I/II.....$7.95
Fullnet Connector $14.95
Switch Boxes..... $19.00 - $29.00
ADB Replacement cable.....$5.95
SCSI Cable, 25-50 or 50-50.....$9.95
Power Cord (US or Intl) $3.00

PERIPHERAL CARDS

Zip GS 7Mhz / 32K Cache ..$159.00
Zip Chip 8 Mhz (//e, //c) $139.00
RAMFast SCSI Card (256K)..$139.00

Manuals & Software

Misc. Software. Games, Utils, More. $5.95 to $19.95. Ask For List.
Apple Manuals Avail. Ask For List



System Includes:
All New
AppleColor RGB Monitor
External Apple 3.5" Drive • Keyboard
Mouse • 1024K
System 6.0.1
ROM 03
$599.00

More Apple IIGS Accessories

Apple ADB Mouse II (New) $69.00..Refrb ADB Mouse .$49.00
Apple IIgs System 6.0.1 Disk/Manual Set $35.00
Apple IIgs System Transport Case (Full System).....$19.95
SoundMeister IIgs w/Digital Session v1.0.....$69.00

Modems

Hayes SM1200 (Refurb) $29.00
2400 External (New) $49.00
Omron Ext 2400/Fax/MNP5..$59.00
14.4 External MNP1-6 v.32bis $99.00
Hardware Hndsk cable (IIgs) $14.95

Bulk Disks

3.5" DSDD Disks 50 for $12.50
3.5" DSHD Disks 10 for $2.90

Alltech Electronics Co.

2618 Temple Heights. Oceanside, CA 92056
9-5:30 PM Mon.-Fri.

Orders Only: VISA MasterCard

800/995-7773

Sales Inquiries: & Tech Support: 619/724-2404
Fax (School PO's, Quotes, Info.): 619/724-8808
BBS (FutureNet #30) 300-14.4 v.32: 619/724-4660

With Allison™ V. 1.0.5. Instantly 'grab's from any video source. Upgrades available. Call .

*** Reduced ***
VisionPlusEnhanced IIgs $149.00

Apple FDHD $199.00

Disk Drives

5.25" Generic Full Height w/19 Pin $69.00
5.25" Apple Disk II (Refurb) add $10.00 for above.
Apple DuoDisk 5.25" Exch.....$129.00
Apple DuoDisk 5.25" (Refurb) $159.00
AppleDisk 3.5" (Platinum).....$165.00
AppleDisk 5.25" (Refurb) $165.00
Apple FDHD (SuperDrive) New $199.00
Apple 3.5" 800K Drive Service Exchange $85.00

SEQUENTIAL SYSTEMS

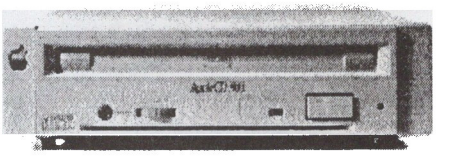
Enhanced DiscQuest Support & Photo CD !
Give your IIgs a Second Sight!
Second Sight: VGA Card for IIgs $179.00
Second Sight & 14" SVGA .28 d.p.Mon .$399.00
Q:Talk LTO (AppleTalk & 32K for IW-II).....$69.00

RAM Cards

Apple 1 Meg Memory Exp. with 1 Meg.....$39.00
GS-RAM IV (4 Meg).....$109.00
GS SuperRAM 4 Meg with 1 Meg.....$74.00
GS Super RAM w/2 Meg $99.00. 4 Meg$119.00

SCSI Hard Disks

40 Meg External w/AEC Card (Refurb) .$159.
40 Meg External Drive (Quantum).....$139.
RAMFast SCSI Card Rev D 3.01ez.....$139.
SyQuest 44.....$259.00 SyQuest 88. .....$359.
External SCSI Drive Cases
Case & P/S for 3.5" or 5.25" HH.....$45.
Case & P/S for 3.5" Drive (1.5 " or less)..$45.
Case & P/S for Full Hight 5.25" or 2 HH.$65.
Case & P/S for Quantum ELS/EPS Drive:
Power provided via any free DB19 port..$19.



DiscQUEST..... $79.00

DiscQuest: CD ROM for your IIgs!
Sequential Systems CD ROM Software
DiscQuest (Incl. Family Doctor).....$79.00
DiscQuest & CD 150 .....$265.00
DiscQuest , CD 150 & RAMFast $395.00
Apple CD-ROM 150 External.....$189.00

Monitors

AppleColor RGB, IIgs (Refurb).....$169.00
AppleColor RGB IIgs (NEW) $225.00
Aamazing SVGA 14" ,28 DP (For SS).....$225.00
Apple Color Comp or Color 100 (exch) ...$109.00
Composite Green (Refurb).....$29.00
Apple Monitor][(Refurb).....$49.00
Apple Monitor /// (Refurb).....$35.00

CA Residents please add 7.75% Sales Tax • COD Orders, Cash/Certified Funds Only • Minimum Shipping & Handling Charge $5.00 (This covers most orders) • Some products are refurbished products and carry a 120 day warranty. • New products are covered by a one year warranty. All Repairs are warranted for 120 days. • Some RAM Chips may be new but pulled, all RAM has a one year warranty. • NO Returns without an RMA number, please call before shipping anything back to us. • A 20% Restocking Fee is applied to all returns due to incompatibility or unneeded product or Products purchased in error. Please be sure what computer you have before placing an order. • Educational P.O.s Accepted, Others on AOC only. • Prices and availability subject to change at any time without any notification whatsoever. *Discounts on certain merchandise only and min. Qty Applies.

Mr. Priceguide Looks at Laser Printers

By Erik "Lurch" Kloeppel

Wow! You've gone and saved a bunch of money, and you are looking to spend it on something really cool for your computer system. Problem is, you have only $700. OK, so it's an arbitrary figure. I'm *full* of arbitrary figures. Now, what can you get via mail order for $700?

Well, you pick what you like, but this week, I'm buying a new laser printer with this imaginary money.

Ah, I can hear your screams from here! "Oh, My Heavens! What kind of crummy laser printer can you *possibly* get for $700?" You might well be surprised; there are more than 20 laser printers available for less than $700, some even under $400! And most, if not all, are worth your consideration.

What's available on these bargain basement boxes? A cynic such as myself might say, "The same stuff that's on the more expensive printers, only less of it." And, while there is a little bit of truth in that statement, in general, 'taint so. The most common difference is the print resolution. These printers are all 300 dots per inch (d.p.i.). More expensive printers may be 300 d.p.i, 600 d.p.i or even 1200 d.p.i, and generally have some form of resolution enhancement. (Resolution enhancement is a routine in the printer firmware that adjusts the laser itself to improve print quality. This is usually done by adjusting the size or placement of the dots.) If you think that 300 d.p.i. isn't a good enough resolution, you might be interested to know that this very magazine is printed on a 300 d.p.i. printer! [Which cost $3,500 five years ago! - Ed.]

Dot's Right!

"Dots? I thought this was about laser printers, not dot matrix printers!"

It is! Trust me. If you want to get really really picky, most printers today are dot matrix printers. What varies is the means of putting those dots on the paper. Dot matrix printers (DMP - pronounced "dimpy") use a print head that slams a bunch of wires against the paper to create the dots (and, since they are actually beating the tar out of the paper, these printers are more properly called dot matrix *impact* printers). Ink jet printers spray very regular ink drops at the paper (graffiti gone horribly wrong), and laser printers draw very small dots on a print drum with a laser beam.

Hmm . . . I should probably explain more about how laser printers work. Impact printers and ink jet printers are "direct" printers—they put ink right on the page. Laser printers are "indirect" printers. It might help to think of laser printers as copier machines that read electronic pages from your computer instead of from a glass window on the top of the machine. Once the image is in the printer, and ready to print, a laser beam is fired at a photoelectric drum just like the one in any standard copier. This effectively puts the image on the drum. Next, toner is sprinkled on the drum, where it sticks to the appropriate places (i.e. where the laser beam hit) and is then transferred to the paper. This is over-simplified, but you should get the basic idea.

Before I move on, I should probably mention a cousin of the laser printer: The LED printer. An LED printer works in almost exactly the same way as a laser printer—the main difference is that instead of using a laser beam to place the image on the print drum, an LED-based printer uses light emitting diodes (LEDs) to place the image. Because of this difference, LED printers are a bit slower, but they have fewer moving parts and run quite a bit cooler. One thing that *isn't* different is the print quality that you get out of an LED printer. Unless you have the eyesight of a test pilot (and a magnifying glass) you probably won't be able to tell the difference between the output from a laser printer and an equivalent LED printer. The reason I'm mentioning this type of printer is that people often confuse them with true laser printers. In fact, I've even included a couple of LED printers in the list of printers that accompanies this article (the Okidata and Texas Instruments printers), just to show you how similar these types of printers are.

Inka Dinka Doo

Impact printers get their ink from a ribbon, ink jet printers get their ink from an ink cartridge, laser printers get their ink from toner. Toner is an interesting substance. If you ever get a chance to see it (and you will if you buy a laser printer), it looks a lot like soot. In its raw form, toner acts a lot like soot too—viscous, sticky, gets into everything, incredibly black, and harder to clean up than anything else you can think of. It's composed essentially of ink dust and small plastic beads that melt easily. Fortunately, this incredible mess is tightly packaged in a "toner cartridge" which is easily installed or removed from your printer—a job that is just about as easy as changing the ribbon on your ImageWriter II.

Some toner cartridges also contain the print drum and other printing equipment. This is a good thing because after significant use, the toner (which is a bit abrasive) tends to abrade and scratch the print drum, which leads to deteriorating print quality. In other words, you should change your print drum and associated parts periodically or your beautiful print will start to look sad. Not to worry too much though, it usually takes a *long* time for any visible deterioration to show. Fortunately, you normally need worry about this only if you do a lot of cartridge refilling. New cartridges usually come complete with drum and other essential, replaceable items. Printer manufacturers usually suggest that you not refill your toner cartridges, but realistically, they are good for two or three refills at least, and a good bargain, if you can avoid spilling toner all over the house. A better choice falls between refilled and new cartridges: There are companies out there that will remanufacture a cartridge for you (i.e. refurbish the drum and fill it up with toner), for *half* the cost of a new cartridge.

Other Considerations

Other items that affect the final price of a laser printer include "Pages Per Minute," "Interface," "Emulations," and what forms of media it can print on.

Figure 1
Toner Cartridge Remanufacturers

LaserTek
3455 W. Reno Ave.
Las Vegas, NV 89118-1601
1-800-252-7374

National Business Machines
P. O. Box 1932
Stafford, TX 77497-1932
1-800-765-0707

Pages Per Minute (ppm) is commonly thought to be the most important concern when buying a laser printer. It's certainly talked up at least as much as CPU speed and automotive horsepower. While it may be as important as either of those two items, I disagree with the overall importance placed on any of them. In fact, I suggest pages per minute is the *least* important concern for a home based printer. Very few laser printers are slower than 4 ppm, which is plenty fast enough for home or small business use. If the pages per minute is really your primary concern, not to worry; there is a range of speeds in this list, from 4 ppm to 10 ppm. (For comparison, here are the "official" speed figures for the ImageWriter II: 0.5 ppm in "Best" mode and 2 ppm in "Draft" mode. These figures were taken from the *Apple Imaging Facts* propaganda booklet, so your mileage may vary.)

Here's another thing to note about print speed: The figures quoted are usually supplied by the manufacturer, and as such should be taken with a grain of salt. Usually, print speeds do not take into account the time needed to send the file to the printer, and for the printer to translate it into something it can deal with. So don't get upset if you measure your actual print speed at something significantly less than that advertised. The ppm is just a yardstick—and all 4 ppm printers with the same amount of memory will print at about the same speed.

Speaking of RAM, the amount of RAM in the printer is something to consider too. As with your computer, the more RAM the merrier. For a laser printer it means improved print speed, and is of fair importance. Just about any page description language translates your simple page into something substantially larger, and graphics pages, or pages with a lot of different fonts on them, can *really* eat up RAM. If there is no room in the printer's RAM for the complete translated document, the printer will translate only portions of it at a time. It's still faster than your old ImageWriter II, but it slows things down a lot. If you have more RAM in your printer, the printer can translate more pages at a time, and have the next page ready when the previous page has finished printing—much more efficient than jumping back to the computer for bits and parts of individual pages. Fortunately, it's usually just as easy to add RAM to printers as it is to add it to your IIGS—just add a memory card or a SIMM and away you go.

Now, what the heck do I mean when I say "emulations"?

But I Want Color!

Color printers in our price range (less than $700) usually fall into one of two categories: Impact or inkjet. A multi-colored ribbon is used in the impact printers like the ImageWriter II, and several little ink cartridges are used in the case of color inkjet printers such as the Hewlett Packard DeskJet 550C or the Apple Color Printer. There *is* a third category: Thermal Wax Transfer, where little globs of different colored wax are deposited on the paper. This method results in a "one shot" printer ribbon, and is used in printers such as the Fargo Primera color printer. So, with all this in mind, we'll not be discussing any color printers in this article. I merely mention them because they are sometimes lumped together with real laser pritners in the ads, and to dispell any reasonable hope of getting a color laser printer for less than $10,000. For most home and small business use, it would be much more cost effective to zip over to your local Kinkos or PIP Printing for any color work you might need done.

Habla PostScript?

In an effort to sell their printers to as wide a range of customers as possible, the printer manufacturers have arranged for their printers to "emulate" the page description languages of certain popular printers. For instance, virtually every printer in our list emulates at least one type of Hewlett Packard (HP) laser printer. In general, this means that if your software works correctly with a particular HP printer, any printer that emulates that same HP model should also work. The trick here is to be aware that there are two or three different flavors of HP printers, and emulations for one version will likely not work well on printers that support a different page description language. So, if you have software that requires the HP LaserJet 4, odds are quite good that it will not work on a printer that emulates only the HP LaserJet II. Fortunately, there are printer drivers for the IIGS that work well with most Hewlett Packard printers: Independence and Harmonie. Using one of these should let you make use of any HP emulating printer around without too much fuss. (See the "Apple IIGS Parallel Cards & Printer Driver Software" sidebar for contact information, and be sure to ask the publisher if their drivers will work with the printer you plan to buy. Note that Hewlett Packard printers use a page description language known as "Page Control Language" or "PCL." There are several different revisions or levels of PCL [which are, for the most part, backwards compatible], and the name of a HP printer does *not* necessarily correspond to the version of PCL that it uses. [For example, the HP LaserJet IIP uses PCL 4, not PCL 2.] Printers that emulate HP printers will often be described by the PCL revision they support rather than saying something like, "emulates the Hewlett Packard LaserJet 9.")

Emulations are, in my book, the most important item to consider when hunting for a laser printer. Almost all laser printers offer HP emulations (a good thing to have when working with 8-bit applications). Since everybody offers compatibility with Hewlett Packard, I need something more interesting to draw my wallet from my pocket. PostScript compatibility is enough to rip my wallet out of my fingers before I get it open. PostScript is one of the (if not *the*) best page description languages available, and any Apple IIGS can make use of it right out of the box by using the LaserWriter printer driver that comes with the IIGS System Software.

In other issues of this very magazine, Joe has introduced (and improved on) a program that lets you easily print PostScript files from your IIGS: LASERbeam (last seen in *GS+* V5.N4). And, Don Lancaster did a long series of excellent articles for Computer Shopper on the PostScript language—using an Apple IIe with his laser printer. (If you have access to the Lancaster articles, the IIGS will work just as well as [if not better than] the IIe Mr Lancaster used.)

Insert Tab B . . .

Now we come to what I consider the second most important consideration in buying a laser printer: The interface. Right off the top of your head you can probably think of two interfaces, but believe it or not, there are actually four used with laser printers. The first two you are no doubt familiar with: Serial and parallel. You may even have guessed AppleTalk (a.k.a. "LocalTalk"), but almost nobody ever guesses SCSI. (No fault there, SCSI is obscure when used as an interface between computer and printer, and very rarely used. There are also printers that have EtherNet interfaces, but since there is no EtherNet card for the IIGS, we won't be discussing those.) Serial is the most common interface in the Apple II world—most of our printers use it, and all of our external modems do. A laser printer with a serial interface is easily attached to your Apple II—and you probably already have the cable.

On the other hand, the most common interface in the printer world is the parallel interface—something that is

(unfortunately) not built into your Apple IIGS. It's a good thing there are parallel interface cards for the Apple II out there! (See the "Apple IIGS Parallel Cards & Printer Driver Software" sidebar for a list of vendors.)

Then there's AppleTalk. Every IIGS has an AppleTalk connection built into it. Unfortunately, very few laser printers do, but enough do to make it worth considering, and considering hard. One of the best reasons to get a printer with an AppleTalk interface is that if you already have an ImageWriter II, you can upgrade it (for about $100) into an AppleTalk printer, and then you can hook both printers to your IIGS at once. (For more information on setting up something like this, see the "Apple (Jive) Talkin'" article in *GS+* V5.N1.)

As I said, SCSI interfaces are few and far between in the world of laser printers. A fair number of printers do have SCSI ports, but it's to hook a hard drive up to the *printer*. You are supposed to use this hard drive to store your fonts, thus freeing up your computer's hard drive for important stuff—such as your mother's birthday card or your own wedding invitations. Nobody seems to be able to tell me if this would work with your IIGS, or, more to the point, if it would be worth doing. I'm inclined to say "Yes" it's do-able, but "No," it's not worth it, unless you *seriously* collect fonts. (Either way, you would need to find a programmer to write a PostScript program to download the fonts from your computer to the printer's drive.)

So, what's the *real* difference between the interfaces?

Well, serial is slowest, usually running at 9,600 bps (the same as your basic dot matrix printer). If you have a large document to print, you will wait correspondingly longer for it if you use a serial connection. Serial is also the least expensive, since your IIGS already has two serial ports, and you quite likely already have a serial printer cable.

Parallel is faster. Its transfer speed is dependent on the CPU speed of both the computer and the printer, but speeds in the vicinity of 57,600 bps are not unreasonable, and faster speeds are quite possible. Of course, it will cost you more, since you have to find and buy a parallel card and cable so you can use your printer, but the difference in speed may make the expense worthwhile for you. Quality Computer and Sequential Systems each sell a good parallel card. (Again, see the "Apple IIGS Parallel Cards & Printer Driver Software" sidebar for vendor information.)

The third alternative is my favorite. As I said, every IIGS has an AppleTalk port, and several good printers also have AppleTalk ports built in. AppleTalk is my choice for several reasons. First, it's faster than serial—230 Kbps instead of 9,600 bps. Second, it's built into your IIGS. Third, if you have more than one computer, you can usually share the printer—either by using two different interfaces on the printer, or by putting everything on the AppleTalk network.

What's Your Type?

Another point to consider is the number and type of fonts built into the printer. Using built-in fonts results in significantly faster printing, since the printer doesn't have to translate font data sent from the computer. All it has to do is look at the print job and say to itself "yup, that there is an Italic Bold Helvetica 'M'." If you don't use one of the built-in fonts, the printer not only has to decipher what the character is, but also what font it is, then adjust everything accordingly. Fortunately, the most commonly used fonts are frequently built in, so, unless you intend to use a lot of off-the-wall fonts, you can count on a fairly speedy print process. Check the back issues of *GS+* Magazine for lots of additional info on fonts and printing from your IIGS. ("How Fonts Work" in *GS+* V3.N2, "How Printing Works" in *GS+* V3.N3, and "TrueType on a LaserWriter" in *GS+* V3.N5)

There are font cartridges, ROM packs that can be plugged into most laser printers, that can seriously expand the number of fonts "built into" your printer. But for home use, this is probably a horrendous waste of money. The cartridges cost anywhere from $50 to $500, and it would take a *lot* of printing for the speed increase to pay for itself.

Paper Pusher

The print media you can use in a laser printer is essentially a non-issue. Given the proper paper-feed trays, or even hand-feeding, most printers can print on just about any piece of paper that fits. You do need to be a bit concerned about the thickness (weight) of the paper. The path the paper takes through the printer is the deciding factor here. Most printers have a rather convoluted path going up, down, and all around, while a few offer a "straight-through" path. These "straight-through" printers can handle much heavier paper than the other printers. This means business cards, formal letters, and shopping lists can easily be produced via your laser printer. However, if you intend to use the printer for business, there may come a time when you wish to make transparencies. Fortunately, most printers can handle it, but you must be sure to use the special transparency sheets that won't melt in your printer! If you think that mess in the kitchen sink is nasty, you haven't tried to clean melted plastic off the heated rollers in a laser printer.

Laser printers can print on envelopes, but in many cases you have to feed them to the printer one at a time, or in very small batches (less than 10 at a time for some printers). So, if you plan on printing lots of envelopes, make sure that the printer you buy has an appropriate envelope tray available, and that it holds a decent number of envelopes.

Everything else about any given laser printer is aesthetics. Dimensions range from something that will fit nicely on

Apple IIGS Parallel Cards & Printer Driver Software

GrafStar II Parallel Card - $35
Grappler Parallel Card - $45
Alltech Electronics
2618 Temple Heights
Oceanside, CA 92056
619-724-2404

Apricord 8 Parallel Card - $39.95
Grappler Plus Parallel Card - $89.95
Quality Computers
20200 Nine Mile Rd
St. Clair Shores, MI 48080
800-777-3642

Qprint I Parallel Card - $44.95
Qprint II Parallel Card - $62.95
Sequential Systems
1200 Diamond Circle
Lafayette, CO 80026
303-666-4549

Harmonie Printer Driver Software
Vitesse Inc.
P. O. Box 929
La Puente, CA 91747-0929
818-813-1270

Independence Printer Driver Software
Seven Hills Software
2310 Oxford Rd
Tallassee, FL 32304-3930
904-575-0566

What do Toner Cartridges *Really* Cost?

When buying a laser printer, you need to consider how much it's going to cost you when you need to get new toner. A quick perusal of the various catalogs reveals that costs range from $20.79 for the Okidata 400e toner cartridge to $140.95 for the Epson Action Laser 1000 cartridge.

Looks like a no-brainer, doesn't it? The Okidata printer costs about $440, the Epson about $525, and look at the differences in toner costs! What's not obvious is the hidden cost. For that $20.79, all you get from Okidata is a toner cartridge. You don't get a new print drum for *that* price. A drum will cost you $158.15. Of course, you need not replace the drum every time you replace the toner—you can usually go at least three toner refills before you *have* to replace the drum. (The local dealers I talked to suggested that you not go longer than that because the toner used is very abrasive, and the Okidata drum is softer than most other printer drums.)

Of course, the Okidata still works out to less expensive to replace cartridges: $20.79 x 3 = $62.37 for toner, plus $158.15 totals $220.52 for three toner replacements. Compare that to $140.95 x 3 = $422.85 to replace the cartridges in the Epson. What this doesn't tell you (and the dealers couldn't tell me) is how many pages either cartridge is good for, so it's entirely possible the Epson cartridge is still more economical.

Here is a list of eight more printers and their toner/drum costs, and as you can see, there is a wide disparity in price.

Printer	Toner/Drum Cost	Printer	Toner/Drum Cost
Apple Personal LaserWriter NT	$64.97	Apple LaserWriter Select 310	$87.99
Epson Action Laser 1500	$140.95	Hewlett Packard LaserJet IIP	$64.97
Hewlett Packard LaserJet 4L	$59.97	Lexmark 4037SE	$132.95
Panasonic KXP-4410	$37.95	TI microWriter PS23	$46.59
Developer:	$145.00		
Fuser:	$149.15		
Drum:	$119.25		

Obviously this is not all the printers available, nor even all the printers available for under $700, but it *is* representative of the costs involved in cartridges for all printers. The one remaining factor involved in the print process that I was not able to obtain is the number of pages each toner cartridge is good for. (For that information, you would probably need to contact the manufacturer.)

Also, these prices are for name brand new cartridges. You can save a ton of money buying remanufactured cartridges. For instance, remanufactured cartridges for some printers that can be had for as little as $40.87. (See Figure 1 for the addresses of two toner cartridge remanufacturers.)

When all is said and done, if you have to be careful of each and every penny you spend, you need to check not only the obvious cost of the toner cartridge, but whether you need to replace other parts as well. Also of concern is the number of pages your proposed toner cartridge is rated to print. An inexpensive cartridge is of limited value if you have to replace it very often.

your desk to "get a printer stand." Weights range from 15 or so pounds to easily twice that. Paper input and output trays range in capacity from 100 sheets to 250. There is even disparity in where the input and output trays are placed, so be sure that the printer you get will fit in the area you have set aside for it.

Penny Pinching Printing

Of course, there are "hidden" costs to using a laser printer. In fact, it's still one of the most expensive ways to produce a printed page. Using your old ImageWriter, with its nasty fading ribbon and a good quality piece of paper, it probably costs in the vicinity of half a cent per printed page. Figure a good sheet of paper costs half a cent (one ream at $2.50 = 0.5 cents per sheet). The printer ribbon can be re-inked and re-used for a long time—probably long enough to make the expenditure negligible (I'm still using the same ribbon that came with my ImageWriter II, years ago). But let's ignore the re-inking possibilities. A new ribbon costs about $1.25, and is good for a couple hundred pages or so before fading to a quality you wouldn't want to use in a professional letter. So: $1.25/200 = 0.625 cents per page. Not much at all.

An ink jet printer is roughly twice as expensive as an impact printer, and the ink is much more expensive per unit bought. New ink cartridges run about $15, and I'm told they last about a thousand pages ($15/1000 = 1.5 cents per page), and they don't fade.

Believe it or not, your choice of laser printer affects your cost per page, because they all use different toner cartridges, use different amounts of toner per page, and get different numbers of pages out of each cartridge. The different cartridges vary significantly in cost, from as little as $50 to as much as $175 or even higher. One cartridge I am familiar with off the top of my head costs about $60, and is rated for 3,000 pages (these are text pages, with 5% total page coverage, not graphics pages, which are something else entirely). A little tap on the calculator shows $60.00/3000 = 2 cents a page. Frequently the more expensive cartridges have a longer "life," and are thus more cost effective, though this is not always the case, since some also contain the print drum and other printing hardware. Also, the first toner cartridge is included with the printer when you buy it. Of course, the price per page goes down a bit if you use "remanufactured" toner cartridges, just as it would by using re-inked ribbons and reloaded ink jet cartridges. (See Figure 1 for contact information.) Cost per page is obviously not the concern today that it was several years ago. In fact, the average cost per page for laser printers has dropped from the 10-15 cents/page range to the 1-3 cents/page. That is a cost I can live with in the interests of better quality print. (See the "What do Toner Cartridges *Really* Cost?" sidebar for more information.)

Why Buy a Laser?

Thought you caught me, didn't you? If I'm concerned with print quality and cost, why don't I just buy an ink jet? It's virtually impossible for the naked eye to detect the difference in quality between ink jet and laser (300 d.p.i.) output, and ink jets can be bought for under $200. Why spend at least twice as much for a laser printer?

For a couple of reasons, actually. First, unless you use high quality paper with your ink jet, even high quality ink can run when it gets damp—even a moderately sweaty hand can smudge the print. And let's not mention rain or spilled coffee! Granted, the paper isn't that much more expensive, nor is it difficult to find, but the laser printer finishes its print job by actually using hot rollers to iron the page as it leaves the printer. This action melts the little plastic beads in the toner and fuses everything to the paper permanently. It takes more than a little rain or sweat to smudge a page from a laser printer.

Second is special effects. Companies like Paper Direct (201-507-1996) sell special foils that must be used in a laser printer. Using these foils you can add spectacular effects to what would otherwise be an ordinary printed page. Prism effects and metallic colors can be used to really spice up your printed page.

Sure, these foils and papers are a lot more expensive than the regular printed page, but they aren't used to cover the entire page—just small portions of it. Isn't a little added expense sometimes worth it? Wouldn't you like your report/resume/hate mail to *really* stand out?

Third, even the slowest laser printer will blow the doors off of a ink jet printer when it comes to printing speed. You might not *think* that you mind waiting ten minutes for a page, but when you can get that same page in less than two minutes, the extra cost is worth it.

Fourth, I don't know of any ink jet printers that understand PostScript—which should have been my *first* reason for buying a laser printer!

Buyer Beware!

Recently a new type of laser printer has come to market, and it's one I (as a loyal Apple nerd) dissapprove of. NEC has released the NEC SilentWriter SuperScript 610. At first glance, it's a pretty good looking printer. Until you realize it's been designed from the ground up to work best with Windows. In fact, it runs exactly counter to the industry's growing tendency to put more and more powerful RISC processors in laser printers, shifting the processing load from the computer to the printer. The NEC 610 is optimized to use the printer driver built into Microsoft Windows, and to use the ***computer's CPU*** for most of the work! I dislike a system designed to dump more processing back onto the computer. Be *very* careful of printers like this, they won't work with your Apple IIGS.

One printer to watch out for that you might think *should* work is the Apple LaserWriter IISC. It's from Apple, isn't it? And it's one of those SCSI printers I mentioned earlier. No problem, right? Not so. This particular printer is a "QuickDraw" printer, which means it uses Macintosh QuickDraw as its page description language (not PostScript or the HP PCL language), which means, of course, that it works only with a Mac. Apple included a beta version of a SCSI laser printer driver on the System 6.0 CD-ROM disk. I won't promise that it works with the Apple LaserWriter IISC, much less any other SCSI laser printer, though. (With Apple printers, make sure that they have an AppleTalk port, and that they use PostScript, *not* QuickDraw!)

The bottom line for buying a laser printer, then, is to do your homework beforehand, and make sure you can return the printer if it *doesn't* work with your IIGS. But that's just common sense, isn't it?

Not to Worry

A point of fact, all of the printers in the accompanying list *should* work with your Apple IIGS computer (even if that means adding a parallel card). It's up to you to decide which features are important to your particular application.

I've left reliability and ease-of-use for last largely because they are not important. Um . . . let me say instead that the differences between the various printers in these areas are not important. Reliability for modern laser printers seems to hover at about 50,000 pages. The printer manufacturers go to great lengths to make these things idiot-proof (even I couldn't easily break one), so unless you pour beer into it, your printer will probably last longer than your car. Ease-of-use is so subjective it's hard to measure. While I might find it a simple matter to reach an extra inch to grab my printed copies, you might find that stretch just a little too much for comfort. And you might have no trouble clearing the occasional paper jam, where I might have to call my wife because my hands are too large to fit between the fibberdegibbit and the anti-static rhodoflange.

Finally, most laser printers come with a little LCD window that at least provides codes to tell you what is going on inside. You use the codes (along with the booklet that comes with your printer) to help change settings that affect the way the printer operates. You also look there for error codes—when a print job fails, it's is usually helpful to know why. There are three errors that you *will* see if you use a laser printer: Out of Paper, Out of Toner, and Paper Jam. Big deal. Each is easily and simply corrected. Of course, there are other possible problems which range from the "button your fly" variety to "uh-oh, Houston Control, we have a *problem*," but they are usually few and very far between. Anyway, everything you might see there is usually discussed in some manner in your owner's manual, so you won't be totally in the dark.

To sum it all up, just about any laser printer on the market will work on your IIGS, though you may need a bit of extra equipment or cabling. All you have to do is decide which printer suits your needs best, and how much money you are willing to spend. With operating costs under 3 cents a page, and some models selling for under $400 dollars, laser printers are a viable and cost effective means of putting your thoughts on paper.

What I Recommend

Of course, I don't dare tell you *which* printer you should buy. Your needs will almost certainly vary from mine, and your preferences are, well, yours. But, it you want my *personal opinions* . . . .

I personally think buying any SCSI-based printer is a bad risk. If you buy one, you are taking your sanity into your own hands, since the only known SCSI printer driver is designed to work with the Apple LaserWriter IISC, and is a two year old beta release to boot.

Of all the printers mentioned in this article and in the accompanying list, there is only *one* I would tell you flat-out not to buy, and that is the NEC SilentWriter SuperScript 610. Simply put, it won't work with your IIGS.

For a low use "I wanna look good, but I don't print a lot" printer, I'd suggest getting the Okidata 400E. Since it emulates HP PCL 4, it will work with your IIGS (with the addition of the proper printer drive software), it's very inexpensive, and uses a serial interface. Of course, if you want a printer with PostScript compatibility, I'd suggest the DecLaser 1152. It prints 4 ppm, supports serial, parallel, *and* AppleTalk interfaces, and emulates HP PCL 4 and PostScript Level 2. (PostScript Level 2 is a newer version of the PostScript language. The Apple IIGS LaserWriter driver was written to work with original, Level 1, PostScript, but it should work fine with a Level 2 printer.) Another choice might be the Texas Instruments microWriter PS23. It's an LED printer, not a true laser printer, but it's slightly faster (5 ppm), supports Level 1 PostScript, has parallel and AppleTalk interfaces, and it is slightly less expensive than the DecLaser. [Editor's Note: As a rule of thumb, those of us here at *GS+* Magazine recommend that, for the best IIGS compatibility, you should get a printer that comes with PostScript and an AppleTalk interface. If that printer *also* emulates an HP printer, you've got the best of everything.]

That's it! Armed with this absolutely essential article, and the equally necessary list of printers and printer specs, you are ready to do battle with Dimpy and the forces of evil and win that most coveted of prizes: Your very own laser printer. *GS+*

Mr. Priceguide's List of Laser Printers

(Prices shown were the best available at press time.)

Apple Personal LaserWriter NT - $649
Vendor A, 4 ppm, Interface: S, AT
Emulations: PostScript

Apple LaserWriter Select 310 - $499.99
Vendor L, 5 ppm, Interface: S, P
Emulations: PostScript

Brother HL 6 - $585
Vendor C, 6 ppm, Interface: S, P
Emulations: PCL 4, Prowriter, Epson

Brother HL 6v - $685
Vendor C, 10 ppm, Interface: P
Emulations: Call vendor for details

Canon LBP-430 - $488
Vendor G, 4 ppm, Interface: P
Emulations: PCL 5

Citizen ProLaser 6000 - $659
Vendor F, 6 ppm, Interface: S, P
Emulations: Call vendor for details

DEClaser 1152 - $659
Vendor J, 4 ppm, Interface: P, AT
Emulations: PostScript Level 2, HP LaserJet II

DEClaser 1800 - $689
Vendor M, 6PPM, Interface: S,P
Emulations: PCL, Epson, IBM

Epson Action Laser 1000 - $528
Vendor G, 6 ppm, Interface: P
Emulations: PCL 4, Epson

Epson Action Laser 1500 - $618
Vendor G, 6 ppm, Interface: S, P
Emulations: PCL 5, Epson, HPGL

Hewlett Packard LaserJet 4L - $599.99
Vendor K, 4 ppm, Interface: S, P
Emulations: PCL5

Hewlett Packard LaserJet IIIp - $699
Vendor E, 4PPM, Interfaces: S, P
Emulations: Epson, HPGL 2, PCL 4, ProWriter

Hewlett Packard LaserJet IIp - $489
Vendor B, 4 ppm, Interface: S, P
Emulations: Epson, HPGL2, PCL 4, ProWriter

Hewlett Packard LaserJet IIp+ - $495
Vendor B, 4 ppm, Interface: S, P
Emulations: PCL 4, Epson, ProWriter

Lexmark 4037 SE - $639
Vendor G, 5 ppm, Interface: S, P
Emulations: PCL 4, ProWriter, Lexmark 4019/4029

Okidata 400E - $435
Vendor C, 4 ppm, Interface: S, P
Emulations: HP PCL 4

Okidata 410E - $575
Vendor G, 4 ppm, Interface: S, P
Emulations: HP PCL 5

Panasonic KX-P4400 - $375.95
Vendor D, 4 ppm, Interface: P
Emulations: HP LaserJet IIp

Panasonic KX-P4410 - $439
Vendor G, 5 ppm, Interface: P
Emulations: HP LaserJet IIp

Panasonic KX-P4430 - $585
Vendor C, 5 ppm, Interface: S, P
Emulations: HP LaserJet III

Sharp JX 9400 - $425
Vendor C, 6 ppm, Interface: P
Emulations: ProWriter, HP PCL 4, Epson, Diablo

Star LS-5 - $498
Vendor C, 4 ppm, Interface: P
Emulations: Call vendor for details

Star LS-5EX - $635
Vendor H, 8 ppm, Interface: S, P
Emulations: HP LaserJet III

TI microWriter LED Basic - $494
Vendor F, 5ppm, Interface: P
Emulations: Epson, HPGL2, PCL 4, ProWriter

TI microWriter LED PS23 - $649
Vendor I, 5 ppm, Interface: P, AT
Emulations: Epson, HPGL2, Postscript Level 2, PCL 4, ProWriter

Interface Codes: S = Serial, P = Parallel, AT = AppleTalk
Vendor Key: See the table below for vendor addresses.
Note: Printer capabilities and pricing change with the wind. We reserve the right to be dead wrong in the information presented in this table. Also, some of the printers offered at these prices may be refurbished or rebuilt. Before purchasing check with the vendor to confirm all details about the printer you want to buy.

Code	Vendor
A	Shreve Systems 1200 Marshall St Shreveport, LA 71101 318-424-7987
B	Compu-D 6741 Van Nuys Blvd Van Nuys, CA 91405 818-787-3282
C	Corporate Raider 1449 39th St Brooklyn, NY 11218 718-633-7916
D	Lyco Computers Box 5088 Jersey Shore, PA 17740 717-494-1030
E	Harmony Computers 1801 Flatbush Ave Brooklyn, NY 11210 718-692-3233
F	Multiple Brand Superstore 105 Long Acre Ct Frederick, MD 21702 800 944 3808
G	A Matter of Fax 65 Worth St New York, NY 10013 212-941-8877
H	MidWest Micro 6910 US Route 36 E Fletcher, OH 45326 513-368-2309
I	Vanderbilt Technologies 800 Goodlette Rd #103-132 Naples, FL 33940 813-434-2727
J	Bulldog Computer Products 3241 E. Washington Rd Martinez, GA 30907 706-860-7364
K	Comp USA Direct 15167 Business Ave Dallas, TX 75244 800-266-7872
L	Tredex 225 E 17th St New York, NY 10016 310-301-0300
M	Insight 1912 W 4th Ave Tempe, AZ 85821 800-755-9638

GS+ Back Issue Sale!

Sep-Oct 1989 (V1.N1)
- Fabulous first issue of *GS+* Magazine
- Less than 30 copies left!
- Reviews: Arkanoid II, Crystal Quest, ORCA/C, Rocket Ranger, Silpheed, Test Drive II, TransWarp GS, TurboMouse ADB

May-Jun 1990 (V1.N5)
- AppleFest Report
- Beginner's Guide to System Disks - Part 1
- GS/OS prefixes - PreFixer CDev
- Brush with Greatness - How your IIGS makes colors
- Reviews: CMS 45MB 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, ZipGS

Jan-Feb 1991 (V2.N3)
- AppleFest/Long Beach '90 & Apple II Achievement Awards
- Interview with Jim Carson of Vitesse, Inc.
- Introduction to System Software v5.0.4
- RAM Namer - A CDev that allows you to rename RAM disks
- *GS+* program updates: Battery Brain v1.1, EGOed v1.32c, Teach Translator for GraphicWriter III v1.1
- Reviews: ZipGSX, LightningScan, Design Your Own Home, Print Shop Companion IIGS, Your IIGS Guide, Dragon Wars, 2088: The Cryllan Mission - Second Scenario, Space Ace, Sinbad & the Throne of the Falcon

Sep-Oct 1991 (V3.N1)
- Protecting Your Investment - A Guide to Surge Protection
- A Conversation with Roger Wagner - Part 2
- Working with the Toolbox - Part 4: QuickDraw II
- FGS - A desktop program that generates Fractals
- *GS+* program updates: EGOed v1.36, Autopilot v1.1, NoDOS v1.6
- Reviews: two 100MB hard drives, Nite Owl Slide-On Battery, ORCA/Integer BASIC, ORCA Talking Tools, Storybook Weaver: World of Adventure HyperBole, HoverBlade, Shareware: DeskTop Painter, SoundSmith, IIGS Classic: Bard's Tale IIGS

Jul-Aug 1992 (V3.N6)
- KansasFest 1992
- Introduction to 3-D Graphics - Part 3: Speeding Things Up
- Working with the Toolbox - Part 8: The Control Manager
- Understanding FSTs
- Using rBundles in Your Programs
- Quick Folder - A Finder Extension that allows you to open folders from the Finder's Extras menu. **Requires System 6.**
- Extra Bits - A Control Panel that lets you change the new Battery RAM parameters that System 6 didn't provide a Control Panel for. **Requires System 6.**
- *GS+* program updates: EGOed v1.7 **(requires System 6)**, Quick DA v2.0 **(requires System 6)**, Replicator v1.3
- Reviews: ZipGS (10MHz CPU/64K Cache), Gate, Space Fox, Utility Launch & Utility Works

Sep-Oct 1992 (V4.N1)
- Apple EXPO East
- Open From Desktop - A Finder Extension that allows you to open any item on your desktop from the Finder's Extras menu. **Requires System 6.**
- ii Notes - A 20-page NDA notepad. **Requires System 6.**
- Miscellaneous Library - A collection of useful routines to use from any programming language that supports linking to standard libraries
- *GS+* program updates **(require System 6)**: Autopilot v2.0, Quick DA v2.1, EGOed v1.7.1
- Reviews: ContactsGS, GSymbolix, Kangaroo, ORCA/Debugger, UltraCat, Storybook Weaver: World of Make-Believe

Nov-Dec 1992 (V4.N2)
- Understanding Accelerators
- The Basic IIGS
- Working with the Toolbox - Part 9: The Menu Manager
- Font Reporter - A program that lets you display and print out any font in your system. **Requires System 6.**
- Miscellaneous Library (updated)
- *GS+* program updates: EGOed v1.8 **(requires System 6)**, Replicator v1.3.1
- Reviews: AutoArk, 1990 GEM Apple II CD-ROM, IIGS System Transport Case, Out of This World, TrueType Font Collection, Universe Master
- Review updates: Desktop Enhancer v2.0, Pointless v2.0

(All programs after V4.N2 require System 6.0.1, unless otherwise noted)

Jan-Feb 1993 (V4.N3)
- The World at Your Fingertips
- Understanding the Desktop
- Batt Reporter - A program that generates plain English reports from battery RAM configuration files
- Rainbow - A Finder extension that lets you change the colors of your device icons
- Miscellaneous Library (updated)
- *GS+* program updates: Battery Brain v2.0, Open From Desktop v1.0.1, Rebuild Desktop v1.1, EGOed v1.9
- Reviews: CV-Ram Memory Card, StyleWriter printer, ProSel-16, TransProg III v1.1, Ant Wars, FloorTiles,Quest for the Hoard

Mar-Apr 1993 (V4.N4)
- Beginner's Guide to Finder v6.0
- Working with the Toolbox - Part 10: LineEdit
- LASERbeam - A program that lets you download PostScript files to a PostScript printer
- Font Memories - A control panel that lets you keep your bit-mapped fonts on a disk other than your startup disk
- EGOed lite - a smaller, faster version of the EGOed New Desk Accessory
- Miscellaneous Library (updated)
- *GS+* program updates Rainbow v1.0.1, NoDOS v1.8
- Reviews: Salvation—Deliverance, DreamGraphix, The Manager, The Passport House Letter, The Lost Tribe, DuelTris

May-Jun 1993 (V4.N5)
- The Scavenger - Using your IIGS with CD-ROMs from other computers
- Apple EXPO West Report
- Anna Matrix - a Cool Cursor Editor
- *GS+* program update: Cool Cursor v2.0, Miscellaneous Library
- Reviews: Apple Desktop Bus Mouse II, Baccarat, Key Fonts Pro CD-ROM, MAZER II: The Ghost of Mordaine, Pick 'n' Pile, Shanghai II: Dragon's Eye, Solarian GS, Twilight II, TypeWest Volume 1

Jul-Aug 1993 (V4.N6)
- System 6.0.1—For Users
- KansasFest 1993
- Catch the .WAV: A Guide to Scavenging Sound Files
- Secrets of Writing Twilight II Screen Blankers
- Finder Binder: Avoid the annoying "An application can't be found for this document" dialog by connecting documents to an application
- *GS+* program updates: AutoSave v2.0, EGOed lite v1.0.1, Extra Bits v1.0.1
- Reviews: Castle Metacus, HardPressed, The Lost Treasures of Infocom, Treasures From Heaven: Quest for the Hoard 2, Your Money Matters, Zip Drive

Sep-Oct 1993 (V5.N1)
- So You Bought a Hard Disk . . . Now What?
- Apple (Jive) Talkin'
- An Introduction to Object Oriented Programming
- File Dump: A complete Object Oriented Programming example written in ORCA/Pascal v2.0.1.
- *GS+* program updates: Anna Matrix v1.0.1, Cool Cursor v2.0.1
- Reviews: Applied Engineering's High Density Disk Drive, Apple II SuperDrive Controller Card, MODZap, soniqTracker, ORCA/Pascal v2.0.1, SoundMeister, TypeSet

Nov-Dec 1993 (V5.N2)
- IIGS Maintenance–Part 1: The Mouse and Keyboard
- SCSI ("Simple Connections," Says Igor.)
- Balloon v1.0: A finder extension that lets you extract files from ShrinkIt Archives
- CD-ROaM: An application that lets you scavenge files off of CD- ROMs
- KaBlooie! A version of the classic game Minesweeper for your IIGS
- Reviews: 3D Logo, Focus Drive Hard Card, Prism, Tulin Floptical Disk Drive

Jan-Feb 1994 (V5.N3)
- IPC (Igor's Playful Code) - A guide to using IPC on the IIGS
- EGOed v2.0: Read and write RTF files, plus a new color menu
- MIDI Surgeon: Convert MIDI data files to MIDI Synth format
- Reviews: Ancient Glory, Apple Extended Keyboard, AudioClips, GNO/ME 2.0, HP DeskWriter 550C Printer, HyperLogo, NCS Pro 240 Hard Disk, Pedigree

March-April 1994 (V5.N4)
- Programming the IIGS - Part 1: Getting Started
- Playful - A Finder extra that plays ALL rSounds in ANY type of file!
- What Is This? - A Finder extra that gives you information on any icon you select.
- LASERbeam v1.1 - Now, download PostScript files and FONTS to your PostScript printer!
- Miscellaneous Library - New routines let you read Macintosh resources!
- Reviews - Addressed for Success, ORCA/Debugger vs. Splat!, ORCA/Modula-2

May-June 1994 (V5.N5)
- Programming the IIGS - Part 2: Programming the *IIGS*
- Mr. Priceguide Looks at Hard Disks - Advice on buying a hard disk mechanism.
- FLI Convert - An application that converts PC FLI animations into PaintWorks animations!
- MoreSound - An application that lets you change the events in the Sound control panel.
- Reviews - DiscQuest, MS-DOS File Utilities, Salvation: Bakkup v2.0, Spectrum

July-August 1994 (V5.N6)
- Programming the IIGS - Part 3: GS/OS and the Toolbox
- So You Bought a Hard Drive Mechanism. Now What?
- Working With the Toolbox - Part 12: Standard File
- Clip On - View the System Clipboard from any desktop program
- Sun Dial - A great new clock NDA
- What To Do - NDA to do list manager
- Reviews - Six Pack, The Tinies

Sep-Oct 1994 (V6.N1)
- Halloween Fun With Your IIGS
- All About IIGS Graphic Formats
- Find Original - Easily locate the files your Finder aliases point to
- Table Scraps - The best IIGS scrapbook program yet
- Reviews: DiscQuest Encyclopedia, Quick Click Calc

Sale Prices - Good Through 12/22/94

$3.75 For Each Magazine

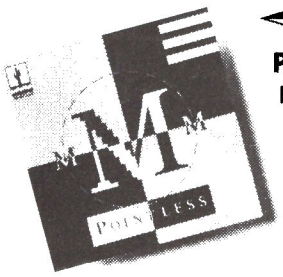
$4.75 For Each Disk

$7.50 For Each Magazine and Disk Set

All prices include postage and handling. (Orders will be sent First-Class to the U.S., Air Mail to Canada and Mexico, and surface mail to all other countries. For Air Mail to all other countries, add an additional $3 per issue. If you pay by credit card, you will be charged only for the additional postage needed.) *Tennessee residents must add 7.75% sales tax.* Mail back issue requests to: *GS+* Back Issues, c/o EGO Systems, PO Box 15366, Chattanooga, TN, 37415-0366; or call 1-800-662-3634 (or 615-332-2087), Monday through Friday, 9 a.m. to 5 p.m. Eastern Standard Time, to order or 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 credit card number, expiration date, and signature. *Any issues not listed are sold out!*

The Best of the West

Since 1990, WestCode Software has created the best software you can get for your Apple II computer. And, now for a limited time, we're offering you the best prices ever on our highly-acclaimed products.

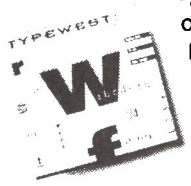


≺ Fantastic looking text on your printer.

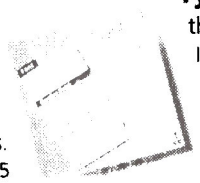
Pointless 2.0 All your documents will look better than ever before, without unsightly hard-to-read jagged text. Pointless brings TrueType font technology to your Apple IIGS, providing impressive laser-sharp text on your ImageWriter, StyleWriter, DeskJet, LaserJet, and other printers. Plus, text on the screen will appear smooth at even the largest sizes.
For the Apple IIGS, System 5 or later. Reg. $69.95
Sale $39.95

The quick brown fox
The quick brown fox

ImageWriter II printing before and after Pointless



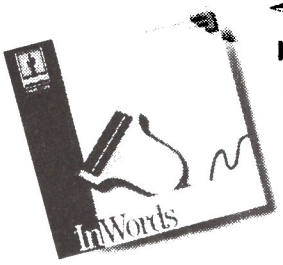
TypeWest An outstanding collection of 40 TrueType fonts for use with Pointless. These professionally-designed typefaces are perfect for home, school, or work. Don't waste your time and money on lesser quality fonts from other sources.
For the Apple IIGS with Pointless. Reg. $49.95
Sale $29.95



TypeSet The perfect companion for Pointless that helps you manage your TrueType fonts. It provides a What-You-See-Is-What-You-Get Font menu that displays your fonts in their own typefaces, font sets to work with custom groups of fonts, and extensive font reports.
For the Apple IIGS with Pointless. Reg. $49.95
Sale $29.95

Stretch your disk space. ≻

HardPressed 1.0 If you're always searching for more disk space, then you need HardPressed — the best disk compression software for your Apple IIGS. It virtually doubles the space on any type of disk. Your 40 MB hard drive will be able to store 80 MB, or a 2 MB RAM disk can store 4 MB! HardPressed works behind the scenes, transparently compressing your files as you save them. When you open a compressed file or launch a compressed application, HardPressed automatically expands it for you.
For the Apple IIGS, System 6 or later. Reg. $69.95. AutoArk trade-up $24.95
Sale $39.95



≺ It's like typing 500 words per minute.

InWords 1.1 Optical Character Recognition (OCR) software that let's you use a Quickie or LightningScan hand-held scanner to scan pages from books, magazines, letters, etc. into text that you can load and edit in your favorite applications: AppleWorks, PublishIt, BeagleWrite, GraphicWriter, HyperStudio, and others. NEW version 1.1 adds improved recognition, speed, line handling and more! Why type? Get InWords instead!
For the Apple IIe or Apple IIGS. Reg. $129.00
Sale $49.95

Call 1-800-448-4250

Order direct from WestCode Software and save! 30-day money-back satisfaction guarantee.

WestCode Software, Inc., 15050 Avenue of Science, Suite 112, San Diego, CA 92128 • (619) 487-9200 • Fax (619) 487-9255

Limited time offer, prices subject to change. All brand and product names are trademarks of their respective companies, and you know who they are. © 1993 WestCode Software, Inc.

Rumors, Wishes & Blatant Lies

By Prof. G. S. Gumby

Magically Fictitious?

Did you know that there is a fix for the annoying "Magic Routing" bug in Finder 6.0.1? You didn't? Well then, you better go read this issue's "Letters" column to find out about it!

What's that? You want *me* to tell you? Well, OK.

To correct the Magic Routing bug, simply change the auxType of your System folder to $00000002. That's all there is to it! Now go read the "Letters" column so you know who to thank for this cool fix!

Wolf DDD (IIGS)(IIGS)(IIGS)(IIGS)

Let's see, I already told you about SimCity for the IIGS .... Did you know that there also exists an almost 100% complete version of Wolfenstein 3D for the IIGS? Well there is! However, there's some doubt that it will ever be published because .... Oops! I just remembered that I can't tell you why. However, if you would like to see it published (i.e. if you would be willing to cough up 40 bucks for it) let me know and I'll pass the information along to the hesitant publisher.

Battling FAX Software

Lets see, at last count, there were two different people trying to come up with send and receive FAX software packages for the IIGS. Now, out of nowhere another company throws its hat into the ring. The question is, who will get their software out first? (The other question is: Will we have a big comparison of all these FAX products? We will if we can get our hands on all of them!)

That's Not It!

We've gotten a lot of letters asking, "Is the Second Sight SVGA card the fabled TurboRez card?" The answer is: No, it isn't. The TurboRez and Second Sight cards are completely different products. (By the way, the Second Sight card is scheduled to be shipping by the time you read this. The TurboRez? Well ....)

Speaking of Which ...

Several software publishers have announced their intention to update their software to take full advantage of the Second Sight card. So, I was wondering, assuming that the Second Sight card does ship soon, what software from *GS+* Magazine would you like to see updated to work with it? Let me know and I'll pass your suggestions along to the boss man.

Go Figure

The boss man tells me that only about half of the people that have purchased Addressed For Success from us have actually sent in their registration cards. So, if you've bought Addressed For Success, take a few minutes to fill out and send in your registration card! (The number of cards we get back is one of the criteria the boss is using to determine whether or not there will be another update—so the more we get back, the better!)

Sniff

Man, I sure do miss Applied Engineering. If they were still around, I could make up one of my famous "AE/Star Trek" products and have everyone calling up asking if it was true. Aw, what the heck ....

Warp Speed!

Some former Applied Engineering employees have just announced that they are going into business for themselves! And this time, they'll be sticking strictly to Apple IIGS products!

The new company will be called WarpField Engineering, and their first new product will be a portable hard drive called the "Voyager." According to a WarpField spokesperson, the Voyager "... will not repeat the mistakes we made with the Vulcan." The Voyager will be an external drive that plugs directly into the SmartPort and contains a pass-through port for your other disk drives. By plugging directly into the SmartPort, the Voyager will not need an external power supply and can be easily moved from one IIGS to another. (The Voyager will also have standard SCSI and IDE ports so that you can connect it directly to any controller card that you already have.) The Voyager will initially be available in 240MB, 540MB and 1GB sizes at prices of $300, $500 and $800 respectively.

Other WarpField Engineering products currently on the drawing board include:

- An incremental backup program that allows you to easily save and restore different "generations" of backups.

- A nine-disc CD-ROM "jukebox." Tentatively called the "Disk Spin 9," this product should be finished sometime in 1995.

- A next generation cellular phone/modem that fits in the palm of your hand and allows you to communicate from just about anywhere. Code named "the communicator."

And that's not all they have coming up. Look for more exciting products from WarpField Engineering in the future!

Why Not?

Last issue's *blatant lie* about a PowerMac emulator for the IIGS provoked a couple of readers to send angry letters of complaint. (Thanks folks, you've really cut down on my firewood bills.) Now, while a PowerMac emulator for the IIGS is so far-fetched that any sober person would dismiss it out of hand, has anyone out there given any thought to a coprocessor card for the IIGS with a PowerPC chip on it? Such a thing *could* be done ... couldn't it?

DigiSofty

The folks at DigiSoft Innovations tell me that after many unforseen delays, their IIGS-specific CD-ROM collection of public domain and shareware software (which will probably be called the "Golden Orchard Apple II CD-ROM") should be out by the end of 1994. They also wanted me to let you know that they have a new preferred e-mail address. That address is, "jagaroth@leland.stanford.edu" so be sure to use it from now on.

Ask Mr. 8-Ball

Yes indeed, it's time once again to turn to my best source to get confirmation for the world's most nagging rumors:

Gumby: The boss tells me that the AutoArk update has been finalized and might even ship before the end of the century. Any truth to this?
8-Ball: Yes, definitely.

Gumby: The Newton and PDA markets seem to be falling apart. Is anybody making any money off those things?
8-Ball: Outlook not so good.

Gumby: Since Big Red Computer Club is closing its door at the end of this year, a lot of IIGS products will be left without homes. Will anyone pick up those products?
8-Ball: Better not tell you now.

Gumby: A major IIGS company is said to be working on big updates to two of its most popular products. Will these both be released before the end of 1994?
8-Ball: Outlook not so good. *GS+*

How to Use Your GS+ Disk

The first thing you need to do is **make a backup copy of your *GS+* Disk with the Finder!!!** Do *not* make your backup on your hard disk! Instead, copy the *GS+* Disk to another 3.5-inch disk (this is *very* important). Next, put the original in a safe place. If you are having a problem making a backup copy, give us a call at (615) 332-2087. 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 v6.0.1 or later. (Note that all of the programs on this issue's disk [except EGOed lite] *require* System 6.0.1!) Next, 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 backup *GS+* Disk. (From the Finder, just double-click on the Installer icon.) *It is* extremely *important that you use the Installer that is on your backup* GS+ *Disk! Do* not *use any other copy of the Installer!*

When the Installer window appears, select the item you want to install from the list on the left-hand side of the window, and the disk you want to install it on from the list on the right-hand side of the window. Then click on the Install button. For more information on using the Installer, refer to your IIGS owner's manual.

Before you attempt to use your backup *GS+* Disk, please take a few minutes to read the **a.Read.Me** file for any last minute corrections or information. If you do not already have our EGOed lite text editor installed in your system, you can use the Teach application supplied with System Software v6.0 to read this file.

Installing EGOed lite

The following is a detailed example of how to install EGOed lite. The other programs are installed in a similar manner.

- Start up your IIGS with System Software v6.0 or later—the version of EGOed lite that is on this *GS+* Disk *requires* System 6! (Your *GS+* Disk is *not* a startup disk, so don't try starting your computer with it.)

- Insert your backup copy of the *GS+* Disk into a drive and run the Installer program that is on your backup *GS+* Disk. It is *very, very* important that you run the Installer that is on your backup *GS+* Disk and *not* some other copy of the Installer.

- When the Installer finishes loading, click on the Disk button on the right-hand side of the Installer window until your startup disk appears. (If you only have one 3.5-inch disk drive, you will have to remove the backup *GS+* Disk from the drive and replace it with your startup disk. You should also refer to the "Making Room" section below for hints on how to free up room on your boot disk.)

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 programs on it. If you do, we will not be able to stay in business. It really is that simple!

- On the left-hand side of the Installer window, you will see a list of the items on the backup *GS+* Disk. One of the items in this list should be "EGOed lite." (If EGOed lite is *not* in this list, quit the Installer and begin again. Be sure that you are running the copy of the Installer that is on your backup *GS+* Disk!) Once you see the EGOed lite item, click the mouse on it so that it becomes highlighted.

- Click the mouse on the Install button in the middle of the Installer window. The Installer will then install EGOed lite on your startup disk. If you only have one 3.5-inch disk drive, you may have to switch disks several times. Just insert each disk as the Installer asks for it.

- When the Installer has finished, click on the Quit button in the middle of the Installer window. This should cause your IIGS to restart.

- When your IIGS finishes restarting, pull down the Apple menu and select EGOed lite (note that you have to be in a desktop program like the Finder to have access to the Apple menu).

- When it finishes loading, notice that EGOed lite has its own menu bar. Select Open from the *EGOed lite* File menu and then put your *GS+* Disk in a drive. You should see a list of the files and folders on the *GS+* Disk.

- Open the **Documentation** folder on your backup *GS+* Disk and then open the file **EGOed.lite.Docs**. This file contains complete documentation on how to use EGOed lite. *Please take a few minutes to read this documentation.*

Making Room

If you do not have a hard drive, you will probably have to remove some files from your startup disk to make room for the new desk accessories, control panels, and other system files on your *GS+* Disk.

Towards that end, we have prepared the following list of "expendable" files that you can "safely" remove from your System Software v6.0.1 startup disk to free up some space. (We've put quotes around "expendable" and "safely" because almost *all* of the files in the IIGS System Software have some sort of use! The files listed here are the ones that are the "least" useful for a specified hardware setup.)

Be sure that you *never* delete *any* files from your original System Software boot disk! Always work on a backup copy!

System Software v6.0.1

If you use the System 6.0.1 **:Install** disk to create a minimal, 800K, System 6.0.1 boot disk, that disk will have 26K free when the installation is finished.

It must be noted that *all* of the files on this disk are *very* important and the files that you can *safely* remove depend, for the most part, on your hardware setup. So, please read these instructions carefully before removing *any* files.

The first two files you can delete depend on what you will be doing with your IIGS. If you will not be running AppleSoft BASIC programs, you can remove the file **BASIC.System** (11K) from the root directory of the disk. If you will not be running ProDOS 8 software, you can remove ***:System:P8** (18K).

If you do not care what time it is, you can delete the following file:

***:System:CDevs:Time** (10K)

After that, the files that you can safely remove depend on your *hardware setup*.

If you have a ROM 01 IIGS, you may delete the file:
***:System:System.Setup:TS3** (42K)

If you have a ROM 03 IIGS, you may delete the following file:
***:System:System.Setup:TS2** (37K)

If you do *not* have a 5.25-inch drive, you may delete the following 8K file:
***:System:Drivers:AppleDisk5.25**

If you do *not* have a printer, you may delete the following file:
***:System:CDevs:Printer** (5K)

Finally, if you have deleted all control panels, and you won't be installing any control panels from the ***GS+*** Disk, you can also delete the 18K file:
***:System:Desk.Accs:ControlPanel**

Removing some or all of these files will give you ample room (up to 138K on a ROM 01 IIGS and up to 133K on a ROM 03 IIGS) on your startup disk to install EGOed lite or any of the other system utilities from your backup ***GS+*** Disk.

What is EGOed lite?

EGOed *lite is a New Desk Accessory (NDA) text editor that we provide in each issue of* **GS+** *Magazine.*

When you install EGOed lite on your startup disk, you can use it to edit and print ASCII text, Teach, AppleWorks Classic and AppleWorks GS word processor files from inside any desktop program that properly supports NDAs.

To use EGOed lite, you must install it on a IIGS System Software v6.0 (or later) startup disk with at least 40K of free space.

Note: You will *not* be able to print from EGOed lite or any other desktop program when using an 800K, System 6.0 boot disk. (There isn't enough room for all of the required drivers and control panels.)

If you want to save even *more* space, you might want to consider using Autopilot (from ***GS+*** V4.N1) as a replacement program launcher. With Autopilot installed on the minimal System 6.0.1 boot disk, initial free space goes up from 26K to 163K! You can then use Autopilot to autolaunch the Finder from a second 3.5-inch disk drive and still have plenty of room on your boot disk for lots of system extensions. For more information on Autopilot, refer to the "Autopilot v2.0" article in ***GS+*** V4.N1 or give us a call.

Self-Extracting Archive

We use GS-ShrinkIt v1.1 to compress the *source code* and related files on the ***GS+*** Disk into a *self-extracting archive*. To extract the files from the archive, simply double-click on the **GSP.V6.N2.SEA** program on your backup ***GS+*** Disk. ***You do not need to have a copy GS-ShrinkIt in order to use any of the programs or other materials on this GS+ Disk!***

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

However, you will gain better control over the files you wish to extract if you have GS-ShrinkIt v1.1. If you do not have GS-ShrinkIt v1.1 and you would like a copy, check with your local user group or give us a call here at *GS+* Magazine and we will try and help you locate a copy.

What's on the Disk

There are eight items in the root directory of this disk:

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. Please try to read this file before using the *GS+* Disk.

Documentation

This folder contains the EGOed lite documentation file and the complete *GS+* Glossary. The EGOed lite documentation is a Teach file which can be read using Teach, EGOed lite, or any other TextEdit based editor. The *GS+* Glossary file is a plain text file containing all of the terms defined in the past installments of the "*GS+* Glossary."

GSP.V6.N2.SEA

This is a self-extracting archive (SEA) containing the source code and related files for all the programs contained on this *GS+* Disk. The archive also contains the Miscellaneous Library. Technical information, such as the Miscellaneous Library documentation is supplied in the archive as well. To extract the files from the archive, simply double-click on this file from the Finder. You will then be presented with a dialog asking you where you want the files extracted to. Note that if you try to extract *all* of the files from this archive at one time, they will *not* fit on an 800K disk!

Icons

This folder contains Finder icons used by the various programs on the *GS+* Disk. This folder also contains the **FType.GSPlus** file type descriptors described in the "Elucidation" article.

Installer

This is the Apple IIGS Installer. The installer requires System Software v6.0 or later. Run it to install the other programs on this issue's disk. For more information on using the Installer, be sure to read the example on the previous pages, and refer to your owner's manual.

Programs

This folder contains the Copy Icon, EGOed lite, Elucidation, Geeker, and Ultimiter I programs. Use the Installer provided on your backup *GS+* Disk to automate the installation of these files. EGOed lite requires System 6 to operate. All the other programs on this disk require System 6.0.1 to operate.

Scripts

This folder contains all of the scripts that are used by the Installer to install the files from this *GS+* Disk.

Talk.To.GSPlus

This folder contains our feedback form, a troubleshooting guide, a problem form, and our writer's guide.

The feedback form is a plain ASCII text file. Fill it out and send it in to let us know what you thought of this issue.

The troubleshooting guide contains tips on how to resolve some of the more common problems you may experience while trying to use the programs on your *GS+* Disk. If you are having a problem, *please* read this file before you go to all the trouble of filling out a problem form! But, if the troubleshooting tips don't help, *please* fill out the problem form and send it to us! These are Teach files, you may use EGOed lite or the Teach application to view them.

The writer's guide is a Teach file that explains what you need to know to write for *GS+* Magazine—you may view it with EGOed lite or the Teach application. *GS+*

How to Get System 6.0.1

Everyone should have a copy of System 6.0.1. Fortunately, we have a license to distribute it to our magazine-and-disk subscribers as a part of their subscription. Unfortunately, we can't afford to mail all five of the disks that System 6.0.1 takes up to every magazine-and-disk subscriber. However, we still want to make it easy for you to get System 6.0.1. So, if you are a subscriber to *GS+* Magazine with the companion *GS+* Disk (sorry, but we can *not* distribute System 6.0.1 to our magazine-only subscribers), send us the following items and we will send you System 6.0.1:

1) Five (5) *blank* and *formatted*, 3.5-inch diskettes to our P. O. Box address (which is shown on the back of your magazine). We are asking for "blank and formatted" disks because formatting takes time that we don't have, and it's a great way to tell if a disk is good before you send it to us. *If you send us a bad disk, we aren't going to replace it.*

2) A *self-addressed* return disk mailer with enough postage on it to mail the five disks back to you. (Foreign subscribers without access to United States postage may include International Postal Coupons instead. See your local post office to obtain these.) *If you don't provide a postage-paid, self-addressed return mailer, your disks will be considered "gifts" and will be used for backups.*

3) That's all. Don't send any money. We don't want any money for this.

How Else Can You Get it?

If you are a magazine-only subscriber, here are some other ways to get System 6.0.1.

Your Apple dealer. Bug them until they get it in for you. The retail price is $39, but that includes manuals. The part number is #A0077LL/A. For the name of your local Apple dealer, call (800) 538-9696.

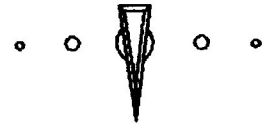
Your user group. Take your own disks and they should only charge you a small copying fee. Some user groups may have it already copied for you and available for a nominal charge. (Note that some user groups make these services available only to their members. Of course, you do plan on joining, don't you?) If you need to know where your local user group is, call the Apple User Group Connection at (800) 538-9696 extension 500.

Resource Central. You won't have to bug them, they have it in stock, and in no less than two different "flavors." For just the disks (item number DA-006), the price is $34.95. For the complete end-user package, including manuals, the price is $49.95 (item number DA-0013). Take your pick, and then give Resource Central a call at (913) 469-6502.

And, of course, if you have a modem, you can download it from your favorite online service. The total download time is about 5 hours. *GS+*

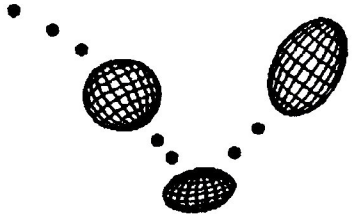
Introducing *Animasia 3-D*™

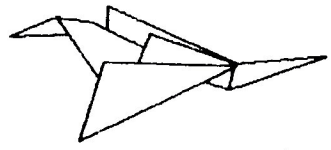
Animasia 3-D is an exciting new desktop animation application for your Apple IIGS. With it, you can create instructive, dramatic, or experimental type animations. Animations can be played by *Animasia 3-D*, recorded to VCR, or included in HyperStudio and HyperCard GS stacks. Three-dimensional objects are the things that make up animations; objects have height, width, and depth, like a chair or a bird. You animate objects just by moving the object to a new position at a new time. Create the starting and ending positions and the in between transitions are filled in automatically.

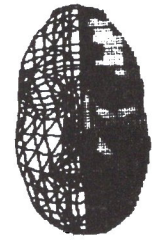


Animasia 3-D is . . .

- **Intuitive** – Elements are directly changeable by pointing and clicking the mouse, conforming to the guidelines published by Apple Computer. In other words, "What you see is what you get."
- **Familiar** – Begins with a basic set of tools, such as those found in drawing and painting applications.
- **Accommodating** – Advanced features are hidden so that beginners can work at their own pace.
- **Forgiving** – Thirty levels of undoable actions protect you from making costly mistakes.
- **Compatible** – Works with all applications, like AppleWorks GS, Platinum Paint, HyperStudio, HyperCard GS, Switch It!, The Manager, Twilight II, etc.
- **Standardized** – Uses existing standards, such as exporting Apple Preferred Format pictures, creating Paintworks animations, and importing AutoCAD "DXF" type object files.
- **Powerful** – Create oval, rectangle, and free-form objects, which can be built upon to create pipes, rings, spirals, cones, and lathed shapes. Cut, copy, paste, move, rotate, size, deform, bevel, align, hide, and lock any object.
- **Structured** – Assign parent-to-child object relationships to aid in character animation.
- **Visible** – Multiple cameras show any perspective of your 3-D world.
- **Lighted** – Brighten your 3-D world with an unlimited combination of four types of lights: Ambient, Directional, Radial, and Spotlight.
- **Animated** – An event-based timeline lets you precisely adjust the timing of any one of an object's 12 attributes.
- **Fast** – Full-screen animations play back at 30 frames per second by taking advantage of the IIGS' unique "Fill Mode" video.
- **Capable** – Does more than Cartooners and Fantavision combined.
- **Supported** – Assistance is available 5 days a week and on-line through GEnie (Animasia@genie.geis.com).
- **Fun** – Creating 3-D worlds, images, and animations is one of the hottest trends in the computer industry.
- **Amazing** – People are already talking about it: ". . . aspects of the user interface . . . convinced me that *Animasia* was probably the most user friendly 3-D program I'd ever seen." --Jerry Kindall, Editor "II Alive"







While *Animasia 3-D* has capabilities rivaling software costing ten times as much, *Animasia 3-D* is **only $99**! Check or money orders accepted. Add $3.50 S&H. Overseas orders add $5 for air mail shipping. Florida residents add 6% sales tax. School purchase orders are accepted.

Available beginning December 10th

System requirements: Apple IIGS with System Software 6.0.1, 2 MB of memory, disk drive, and a color monitor. 4 Mb of memory, a hard disk, and an accelerator are recommended.

Animasia • 3324 Vishaal Drive • Orlando, FL • 32817 • USA
407-380-9932, M-F, 10 AM - 4 PM ET

Reviews

AUGE CD #1

Price: About $55
(See advertisement elsewhere in this issue for complete pricing and ordering information.)

AUGE
c/o Udo Huth
Leipziger Str. 16 a
38329 Wittmar
Germany

Internet: U.Huth@GEnie.geis.com

Not copy protected
Requires System 6.0 or later, a SCSI interface and a CD-ROM drive.

Reviewed by Steven W. Disbrow

Even with the recent release of DiscQuest and the DiscQuest Encyclopedia (see reviews in *GS+* V5.N5 and V6.N1 respectively), we still get calls from folks wanting to know if there are any other CD-ROMs they can use with their IIGS. Well, now there is.

The AUGE CD #1 is a collection of both IIGS and Macintosh public domain and shareware software. (Note that I said "*IIGS* and Macintosh," not "*Apple II* and Macintosh." There is some 8-bit software on this disc, but it all seems to be IIGS-specific.) Don't let the fact that there's Mac stuff on this disk bother you—there's plenty of IIGS stuff too. In fact, according to the Finder, there is about 204 *megabytes* of IIGS material on this disk. Actually, there's quite a bit more than that on the disk, because everything that's on it is compressed using GS-ShrinkIt. (So, if we assume an average compression rate of between 40% and 50%, that means there's probably around 400 megs of IIGS material on this disk. Not bad at all.) Of course, GS-ShrinkIt is also included on the disk, so that you can actually get to all this neat stuff.

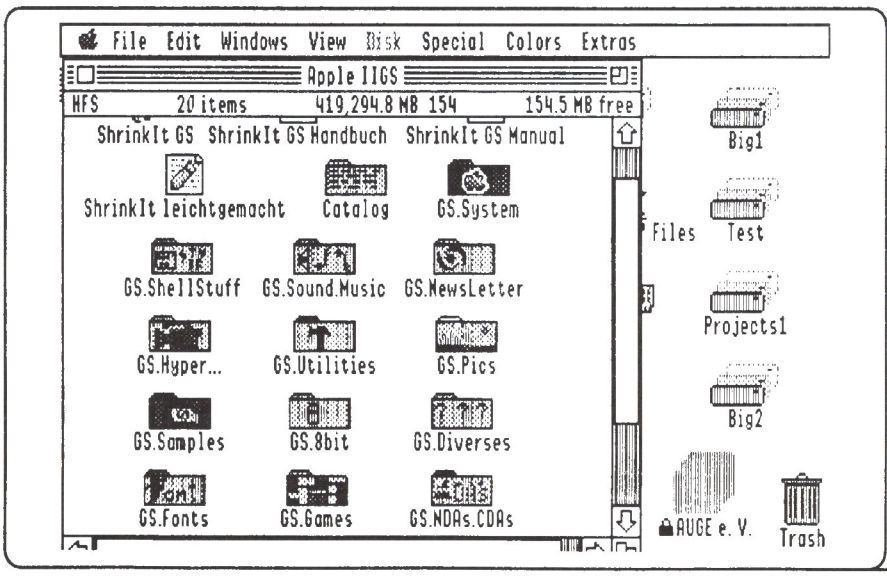
So What?

While the fact that there is over 200MB of IIGS material on this disk is nice, I'm sure just about everyone out there has at one time or another felt "burned" after purchasing a collection of shareware and public domain software that, to put it politely, stunk. Fortunately, in the short time I've had to peruse this disk, I think I can say that this isn't the case with *this* collection. Even if you just take a quick look at this disk, you'll find a lot of useful stuff. What kind of stuff? Well how about:

• System 6.0.1 - Until now, if you wanted to get System 6.0.1 on CD-ROM you couldn't, because it didn't exist on one. You'll find all six disks of it here.

• Technical Notes - The last good place to find a complete set of File Type and Technical notes was on the System 6 Golden Master CD-ROM that Apple put out. Unfortunately, that disk is no longer available. Even worse, some of the notes on it were out of date. We were not able to check every single note, but the AUGE CD #1 seems to contain all of the most up-to-date versions of all the technical notes put out by Apple.

• Drivers - That's right, this disk contains shareware drivers for things like the Epson LQ printer, the LaserJet IIp printer and floptical drives. Of course, I should warn you that these are *not* the commercial drivers available from companies like Vitesse and Tulin, and they aren't likely to perform quite as well. (In fact, the IIp driver doesn't support graphics printing at all!) But, if you have a need for one of these drivers, these beta and shareware versions are better than having nothing at all.



• Programming utilities - If you are a programmer, this will probably be your favorite part of the disk. Included are such favorites as Bryan Pieterzak's ZakPak, and some oldies but goodies like Randy Brandt's "ROSE" editor for ORCA. Along with the utilities there is also lots of source code that you can dissect to your heart's content.

So, What Else?

Well, besides these immediately cool/useful things, you'll also find a heaping helping of:

• Sound and music files - Of course, beauty is in the ear of the beholder with these files, but there are a *lot* of files for you to listen to. There seem to be files in almost every IIGS music and sound format (MIDIsynth, SoundSmith, raw sounds, rSoundSamples, HyperStudio sounds, etc.), so, you might have a problem playing *everything*.

• Stacks - Frankly, I had no idea there were so dang many HyperStudio and HyperCard stacks out there in the IIGS universe. There are almost 100 stacks on here (including one that takes you on a "tour" of the InterNet), so if "HyperMedia" is your favorite buzz word, you should easily find something that you like.

• Fonts - Dozens of TrueType and bit map fonts are included on the disk. (However, only you know if that's something that you will be useful for you. Personally, I try to stay as far away as possible from shareware and public domain fonts. Why? Because professional TrueType font collections are usually cheaper and are almost always higher quality.)

• Pictures - If you like using your IIGS to look at pictures, you'll find plenty to keep you entertained here. You got your GIFs, your screen shots and your 3200 color pictures. Don't have a GIF viewer? Keep looking around on this CD and you should find a couple of them!

• Utilities and extensions - No IIGS shareware collection would be complete without a good selection of utilities (DiskTimer, Cheap Paint and stuff like that) and a whole slew of NDAs, CDAs, control panels, and Finder Extras. There are literally hundreds of these things on the AUGE CD! The stuff you will find ranges from the original NoDOS 16 (the first IIGS program I ever wrote—I released it as freeware), to a strangely altered copy of the famous "Makin Copies" Finder Extra.

• Demos - If you've heard of a program, and thought you would like to try it, you probably wished that you had a demonstration version of it. Well, if it's a IIGS program and it ever *had* a demonstration version, chances are you'll find it here. Just taking a quick look at the demos on the disk, I found demos for DreamGrafix, the original Salvation (now called Salvation: Bakkup), HoverBlade and Universe Master. (Of course, the legendary Nucleus demo is also on here, along with just about everything else the FTA put out.)

More Good News

More than once in this review, I've noted the sheer *volume* of stuff that this disk contains. I've also noted that the quality of the programs seems to be very good. In fact, going through this disk (and frankly, I've really only scratched the surface), I get the feeling that the people that put this together spent a *lot* of time weeding out all the "this was my first program—it doesn't crash often" type stuff that you might find if you just started downloading stuff at random. Actually that's not that big a surprise now that I stop and think about it. The AUGE is actually a well-known user group in Germany, and a lot of this material appears to come directly from the "disks of the month" that they provide to their members. (I don't know this for sure, the documentation is all in German.) So, this material was probably "tested" by lots of their user group members before it ever made it onto this CD.

Another great thing about this CD is that all of the material on it seems to be the most recent available. That's not to say that it's all 1994 stuff (the most recent date I saw on anything was in fact December of 1993), but it's a lot more recent than the stuff on the 1990 GEM Apple II CD put out by Wayzata (see review in *GS+* V4.N2).

What's Not to Like?

Well, OK, it is a bit of a pain that the documentation is in German.... But my German readers don't complain that *GS+* Magazine is in English, so I'm not going to complain too much about that. (Joe reads a bit of German, but the word for "security measures" doesn't seem to appear anywhere on the CD.)

What is a problem is that there *is* a small amount of duplication in the contents of the disk. For example, while looking through the demos, I found two demos for DreamGrafix. I suspect that they are for different versions of DreamGrafix, but it would have been better if only one made it onto the disk. Fortunately, this isn't much of a problem, as I've only found two files that have duplicates elsewhere on the disk.

Another problem is that everything on the disk is displayed, in the Finder, by its small icon, so a lot of the stuff appears *way* on the right hand side of the directory windows. Just be sure to either view the contents of each folder by name or remember to completely scroll around in and investigate each folder.

OK, so those are really just nit-picks. About the only *serious* problem with this disk is that once in a blue moon you might run across a corrupted ShrinkIt archive. But, these are few and far between, so I wouldn't really worry about them too much. (And somewhere out there is a utility that can deal with corrupted archives, but I didn't run across it on the AUGE CD.)

Mr. Rubber, Meet Mr. Road

The bottom line here is that this is a *very* good collection of IIGS public domain and shareware software that spans the history of the IIGS (well, until at least the end of 1993). It is much more current than the GEM CD, not to mention less expensive. (How does it compare to the CD that DigiSoft Innovations is working on? Well, that one doesn't exist yet, so I can't tell you for sure.)

Actually, if you have a CD-ROM drive, this disk is worth having just to have a permanent backup copy of System 6.0.1. If you are programmer, it's worth having just for the complete and current set of technical notes, not to mention all of the ORCA utilities and source code.

So, if you've been complaining about the lack of IIGS-specific CDs you should get this disk. It's so useful, you won't even remember that there is a bunch of Mac stuff on it too. ***GS+***

The SimpleScript Workbook
By Virginia & Paul Raymer

Price: $39.95 ($29.95 direct)

Requires HyperStudio for the IIGS

Roger Wagner Publishing
1050 Pioneer Way, Suite P
El Cajon, CA 92020
(619) 442-0522

Reviewed by Steven W. Disbrow

If you've ever wanted to create your own IIGS programs, you might have been put off by the fact that you actually had to learn a complex programming language like C in order to get anything done. However, with the advent of multimedia authoring systems like HyperCard and HyperStudio, this isn't necessarily so. Using one of these systems, you can create "applications" that are almost as powerful and flexible as those a professional programmer can create, and you can usually do it faster. However, most folks that use a program like HyperStudio barely scratch the surface of what it can do, simply because they are afraid to take the plunge and learn how to use the programming language that is built into HyperStudio: SimpleScript. If, however, you *are* willing to spend the time learning how to use SimpleScript, you can build HyperStudio stacks that are more versatile and more powerful than they would be otherwise.

That's where *The SimpleScript Workbook* comes in. *The SimpleScript Workbook* (which I will simply call "the Workbook" from now on) is a beginner's guide for using, and understanding, the SimpleScript language.

Working With the Workbook

The Workbook itself is divided into sections, each of which covers a single SimpleScript language command. Within each and every section is a complete SimpleScript program that demonstrates the use of that command. All you have to do is type in the program and execute it.

Of course, by presenting complete programs in each section, the Workbook puts itself in the position of having to use commands that haven't yet been discussed, but which are necessary to make the sample programs work. (For example, you need to know the command to ask the user for her name before you can use the `Draw Text` command to draw it on the screen.) Fortunately this isn't a real problem, because each line of every program in the Workbook is cross referenced to show which section of the Workbook covers that line of the program. Consider the following example:

```
(34)  Move To 10, 10
(38)  Draw Text "Hello!"
(8)   End of Script
```

By looking at this listing you would know that the "Move To" command is discussed in section 34, the "Draw Text" command is discussed in section 38, and the "End of Script" command is discussed in section 8. (Note that you *do not* type these numbers into your SimpleScript programs, they are just there for cross reference within the Workbook.) This system makes it *very* easy to move through the Workbook, finding exactly what you want to find, when you need to find it. (There is also a complete listing of all the commands and their section numbers just after the table of contents. Since the Workbook is made up of loose leaf pages in a three ring binder, this can be a handy reference to pull out.)

The Workbook has an Index, which can also come in very handy when you need to find something in a hurry.

The Workbook also has a bunch of SimpleScript hints and tips scattered throughout it. These discuss everything from reserved words to how to use the pull-down menus to create your SimpleScript programs. These tips are also listed in the table of contents for easy reference.

In other words, *The SimpleScript Workbook* is a very well-organized piece of work. It can be a bit confusing at first, but after working through a couple of the sections, I found it easier and easier to use the extensive cross referencing system. The only real problems I found with it were a couple of rare places where page or section references were simply left out. However, there were only two or three of these occurences, so they don't really detract from the overall excellence of the Workbook's organization.

So, What Will You Learn?

Since we've looked at how information in the Workbook is organized, let's look at the information itself.

As I mentioned earlier, just about every SimpleScript command is covered in the Workbook. Each command is explained by actually showing how it works within a complete SimpleScript program. This is very, very good.

Unfortunately, some of the programs have mistakes in them. Most of these errors seem to have been caused by the examples being copied and pasted from one another, without the new copies being correctly revised to reflect the fact that a different command (or a variation of the same command) was being discussed. Fortunately, there aren't really that many of these errors, and, with a bit of programming experience, they are easy to spot. (This leads to a bit of a problem, because the Workbook is very much intended for people with absolutely *no* programming experience. I'll come back to this in a bit.)

The main focus of the Workbook is learning how to write SimpleScript programs, period. So, if you are an absolute beginner with HyperStudio, you *must* learn how to create cards, buttons and stacks *before* you can even think about cracking open the Workbook. Of course, this single focus does mean that you *will* learn a lot about scripting, but you will learn very little, if anything, about any of the other aspects of HyperStudio.

Simple is the Truth!

Finally, I need to address the sample programs themselves. Almost every single one of the examples in the Workbook is mind-numbingly simplistic. If you are a beginner, this will be a blessing of the most divine kind. If, however, you are a programmer with just about any amount of experience, you will probably have trouble staying awake while going through the Workbook. There is one advanced sample in the back of the book (a "text editor on a card" called MicroWord), but that's about all the Workbook has to offer to the experienced programmer. (The Workbook also makes a fairly good reference of all the commands in the SimpleScript language, but if that's all you will be using it for, it's a bit pricey.)

Get to the Point Man!

The bottom line here is that *The SimpleScript Workbook* is a *very* good product for HyperStudio users that want to learn how to take more control of their stacks by using SimpleScript. It's also a good book for people that want to learn the basics of programming without leaving the safety of the HyperStudio environment. Just don't expect to learn anything other than how to write a SimpleScript program. Of course, that's exactly what this book is *intended* to teach you, and it does it very well indeed. *GS+*

Ultima I: The First Age of Darkness
IIGS version by Bill Heineman
Graphics by Scott Everts
Music by Tony Gonzalez
Dungeon art by Scott Campbell

Price: $39.95

Not copy protected
Requires 1 MB RAM and System Software v5.0.4 or later. Installation on a hard disk requires approximately 655K of space.

Vitesse, Inc.
13909 Amar Rd. Suite 2
P.O. Box 929
La Puente, CA 91747
(818) 813-1270

Reviewed by Robert A. Ribaric

The peaceful realm of Sosaria has fallen to the evil forces of Mondain the wizard! Your ally, Lord British, desperately needs your help to rid the land of this scourge. You must travel the countryside on your quest to slay the wicked Mondain. Along the way, you will encounter waves of the evil one's army of creatures. Be ever vigilant as you pass through castles, cities, dungeons, the high seas, and even outer space! Don't miss anything, because you'll need all the help you can get to complete your mission.

Do You Remember?

It seems to me that the original Ultima adventure came out a long, long time ago. However, this masterpiece was never available in a IIGS-specific version. It doesn't really matter whether you've played this game before or not, because this one's a must-buy! Here at the *GS+* Magazine offices, we all remember playing Ultima I, even though not one of us recalled how to solve it. Besides that, there are many new features that make this an almost new experience. The graphics and sound have been updated to take full advantage of the IIGS's capabilities. For these reasons I was very excited to hear that Vitesse was releasing Ultima I: The First Age of Darkness. The nostalgic side of me really enjoyed this "blast from the past," while part of me was interested in the new, advanced features. Joe expressed a similar view. God bless Bill Heineman! First he brought us Out of This World (see review in *GS+* V4.N2), now this. I also hear that something even more exciting is on the way, but I'll leave it to Professor Gumby to tell you about it. Anyway, I love games, so of course I loved reviewing Ultima.

The Play

You can choose from four races and four professions for your character. The races include human, elf, dwarf, and bobbit (the latter seems to be proficient in the use of cutlery). I assume everyone knows what a human is, but elves tend to be weaker and more agile, dwarves are smaller and stronger, while bobbits seem to have a more profound wisdom. Professions include: Fighter, thief, cleric, and wizard. Fighters are obviously stronger and more skilled with weapons than other classes, and thieves are more clever and dexterous. Clerics use their advanced wisdom to cast spells, while wizards use their intelligence and mastery of magic. Some players tend to tailor a character after themselves, while others go for who they'd like to be. It is up to you, so mix and match! You can even specify whether your character is male or female.

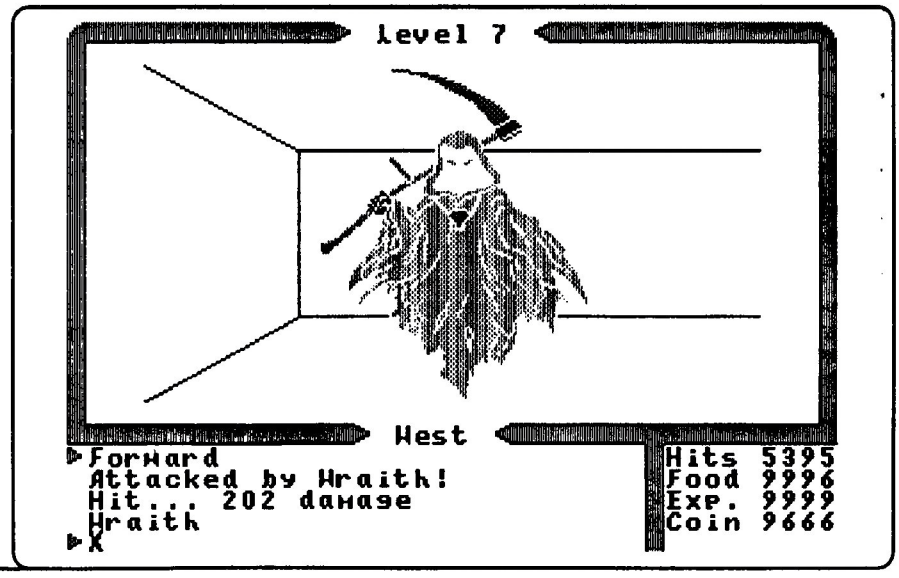
The castles and towns of Sosaria are sources of the supplies your character will need for his or her journey. You start off with a small amount of coinage, so an initial shopping excursion to a city is possible. Food is mandatory, and it can be purchased just about anywhere. There are also many stores that offer armor and weapons, but you may have to look around for the kinds you want. Once you amass some cash, you can even acquire transportation such as a horse, cart, boat, or space shuttle! Finally, inns and taverns offer drinks and gossip. The local kings offer refuge in their castles, and will even help you out in return for tribute or service. Be careful while in towns and castles, as the numerous guards don't take too kindly to trouble-makers. They'll even stick up for that annoying jester!

Strategy

Well, what exactly is it that you're supposed to do? Survive, for one thing. Self-preservation takes a lot of effort at first. When you gain enough confidence, exploring the underworld of dungeons is a good place to start your quest to vanquish Mondain. There you'll battle monsters for great amounts of wealth. The underground world is populated by beings like baldrons, carrion creepers, liches, wraiths, and zorns. Lesser creatures such as gremlins, mimics, and gelatinous cubes also abound. Some even play tricks with your mind! Watch out for mind whippers, wandering eyes, and invisible seekers. You might even be asked by one of the kings to slay a particular beast in return for his favors. Visiting all the different rulers is one way to learn what needs to be done in Sosaria. Another source of wisdom is your local tavern. Much can be learned through a bartender's chatter. You'll soon discover a chain of events that must be unleashed to achieve your ultimate goal. The use of magic can really get you out of a bind during this pursuit. You can acquire spells to fight with, move you around, and open things. As your skill levels rise, you'll get better with spellcasting. Of course, a lot of money is also required to overcome obstacles. I mean, how can you travel to outer space without the proper funding? Anyway, I don't want to give it all away, so I'll let you figure out the rest.

How It Went

When Ultima I arrived, I sat down for a few hours and just played. It was fun at first to just explore, fight, and buy things. Soon, I wanted to begin my quest. Joe was also interested for reasons that will be made clear a little later. We both set out to solve the game separately. I was slowly making progress, but Joe seemed to be doing a lot better. He didn't die all the time like I did. Something was up! I decided to get tips from him about where to go and what to do. This helped a little, but even with the game's resurrection feature, death really got in the



way. Eventually, I ended up pretty frustrated while Joe finished the whole thing!

It turns out that Joe used his programming knowledge to assist his greedy agenda. While I was struggling to stay alive, he was working on a way to edit his character's stats. He ended up with maxed-out skill levels, money, experience, and hit points. He even gave himself every possession the game offered. Sorcery, I tell you! I was beginning to think that Joe was the evil Mondain himself. I knew what had to be done. He had to die. So, one quiet afternoon at work, I slipped into his office and . . . WAIT A MINUTE! Perhaps this new knowledge could be used for good. I mean, we're only trying to save the world. What could be more noble than that? I decided Joe should live long enough to share his secret of immortality and infinite wealth. When this was done, I quickly completed my quest and saved humanity. All in a day's work! Besides, Steve wouldn't have paid me to take the time necessary to finish the game regularly. [Best of all, Joe's cheat program is on this issue's *GS+* Disk for you to use! See "Ultimiter I" elsewhere in this issue for more infomation - Mondain]

My Opinion

Well, even using Joe's cheat program from this issue's disk, Ultima I: The First Age of Darkness was a very addictive and time-consuming game. I can't believe that the latest sequel (Ultima Ten Zillion: The Quest For More Hard Disk Space) is something like ten times bigger! I'd never have the time, nor the patience to undertake such an endeavor. I'll leave that for the truly bored. (I hear Joe got up to Ultima IV!) This first one was just about the right size for me. However, other things could use improvement besides the size. For instance, the city and castle screens were pretty cheezy, as was the outer space mode. These are obviously left over from the original Apple II version. And whose idea was it to combine medieval and outer space scenarios? I'd have just left it all on the Earth. These things are addressed in later sequels, however. I am glad that Heineman and his pals dressed up the sound and graphics a little. The monsters were very cool compared to what I remember from the ancient 8-bit version. The digitized sounds also kept things interesting. The repetitive songs tended to annoy, but those could be turned off. They were neat for a while, though.

One big problem for me was the keystroke for saving the game. Why did they pick "Q" instead of "S" to do this? To make matters worse, *Command-Q* is the keystroke to quit the game, so I sometimes ended up inadvertently quitting when I wanted to save my position! (There is no confirmation when you quit the game.) I can't begin to tell you what it is like to spend hours slowly winning the game, only to have it all dissappear in an instant! I also found the space combat to be rather awkward, but this might have just been part of "the challenge." Don't get me wrong, I really enjoyed the majority of the play. After playing it, I can see why fantasy role-playing games are so popular.

All these things having been said, I can honestly endorse Ultima I for the IIGS. It was one game I thought I'd never see again, but I am glad I did. In today's endless sea of hand-to-hand combat games, one that requires a little intellectual effort is very refreshing. It seems that everything nowadays just involves martial arts. Now, Ultima does require the slaying of a few beasts, but it isn't the focus of the whole game. It is more of a mystery that needs solving. There's also an over-abundance of princesses that need to be rescued! Ultima comes from an age where games were written with thought and somewhat original storylines. I can see where kids like to occasionally pull the spine out of a opponent or show him his own beating heart, but I still prefer games with a noble mission. Hopefully, you do too.

GS+

Ultimiter I

By Josef W. Wankerl

The peaceful realm of Sosaria has fallen to the evil forces of Mondain the wizard! Your ally, Lord British, desperately needs your help to rid the land of this scourge. However, you are failing desperately. Every fifty moves you run out of food and die. Just when you think you have enough money to return to the surface from a dungeon excursion, a Wandering Eyes monster appears and all your hard work goes down the tubes as a resurrection is attempted. And, to top it all off, that pesky Jester keeps pilfering weapons from you! It looks like you need a little divine intervention.

Back when I was still in school, the Ultima games were "The" games to play. I remember playing Ultima during lunch and every waking hour after school. It was a race: A race to see who among the select group of Ultima players could solve the game first. Of course, as with love, war, and game playing, all is fair—including cheating. The budding programmers in our Ultima sect, in a mad fever to complete the game first, started taking the Ultima games apart at the source to find out its secrets. I had pages of notes on what bytes I could change to give me more hit points and spells . . . all I needed was a good disk editing program. When Ultima IV came out, I even wrote a nice little program to assist editing my characters in a "user friendly" manner. When Ultima I: The First Age of Darkness came out for the IIGS (see review elsewhere in this issue), I immediately had flashbacks of what fun it was to poke around and enhance your character, so I took it upon myself to create a nice character editor.

Imagine the look of surprise and despair on poor Rob's face when he realized that my character was fifty times more powerful than his! Of course, for all its powerful abilities, my character still couldn't win the game unless I knew what to do and where to go. The reason why the Ultima games are so popular is that you have to develop your character from a weakling to a tower of strength, and in the process you discover little clues to help you solve the game. You take great pride in knowing that, through your game playing skill, your character has grown. Of course, if you can't get started, the game can be really frustrating, so a little help can be a great confidence booster in its own right.

Editing Characters

OK, so your character isn't doing so well and you want to give it a little boost up in the world. How do you do it? Well, first you launch the Ultimiter I application. Next, you choose the Open menu item from the File menu and find and load your saved game. You'll then see a window describing your character. You can directly type in new values for your character's strength, agility, stamina, charisma, wisdom, and intelligence. You can also type in the number of aliens your character has killed in outer space (the Enemy Kills field), the number of hit points your character has, and the amount of experience your character has acquired. To the right of your character's attributes, you'll see an inventory list—a list of items that you character has in its possession. To change the value for an inventory item, double-click on the item in the list, or simply select the item you want to change and press the return key. A dialog will then appear letting you type in a new value for the inventory count. Once you're satisfied with your changed character, select the Save menu item from the File menu to make the changes stick.

If you want to create an entirely new character, you can choose the New menu item from the File menu. You character will start out just as if you had created a new character from the Ultima I game, except you have the option of sprucing your character up a bit before the game starts.

Special

As if simply changing your character's attributes one-by-one isn't enough, I've provided a host of other goodies you can do to your character under the Special menu. First off, you can choose the Set Attributes menu item to change your character's name, race, gender, and class. The next five items in the Special menu are quick ways to increase your character's attributes and inventory. The Maximize Stats menu item will set each of your character's strength, agility, stamina, charisma, wisdom, and intelligence settings to 99. The Maximize Gems menu item will make sure your character has 99 of each gem type in the inventory. The Maximize Armour menu item will make sure your character has 99 of each armour type in the inventory. The Maximize Weapons menu item will make sure your character has 99 of each weapon type in the inventory. The Maximize Spells menu item will make sure your character has 99 of each spell type in the inventory. You do get the picture now, right?

The next menu item, Set Location, is one of my favorites. When you choose the Set Location menu item, a dialog appears letting you choose the location for your character. This means you no longer have to randomly wander around the Ultima world looking for a particular castle, city, dungeon, or sign post—you can magically "teleport" right to it. However, when you use the Set Location menu item, any craft you have boarded will disappear. This is to protect you from being on a frigate and then moving right on top of Lord British's castle—the frigate would then be stuck there forever. Since my favorite craft is the air car, however, I usually want to take it along with me when I set my location. So, if you hold down the option key when you select your new location, your craft will be magically transported along with your character.

The final menu item in the Special menu is the Launch Ultima I menu item. This menu item was suggested by Rob during the testing stages of Ultimiter I and is among the most useful features in Ultimiter I! When you choose it for the first time, you'll be prompted to find the Ultima I application. Then, Ultima I will be launched and the frontmost character window you had in Ultimiter I will automatically be set to play when Ultima I is finished loading. When you quit Ultima I, you'll be taken back to Ultimiter I and the character you had just been playing will be automatically opened for you so you can make more changes. From then on, every time you choose the Launch Ultima I menu item, you won't be prompted to find the application, since it will be remembered, and the game will be launched again. This is an excellent method to keep your game going strong without a lot of hassle finding characters and launching different applications repeatedly. (When you quit the Ultimiter I application, however, the location of the Ultima I game application is forgotten, so the next time you launch Ultimiter I and choose the Launch Ultima I menu item, you'll have to locate Ultima I again.)

Even with the help Ultimiter I provides, the Ultima I game is still a lot of fun to play, and is perhaps even more so now that you know death isn't so horrible. Now if only I could write a cheat program to give my apartment 9,999 food and my bank account 9,999 dollars . . . . *GS+*

Geeker

By Josef W. Wankerl

One day I was playing around in the Finder and I decided to quit. The familiar quit dialog appeared on my screen and a tiny bell started ringing inside my head. I had just had lunch, so I knew it wasn't Taco Bell, and so I started to ponder the significance of the bell. It then dawned on me that I use the Finder's "Quit to previous application" shutdown option ten times more than I use any of the others. In the technical documentation for the Finder, it was pointed out that there are special preferences that can be changed in the Finder so that the default quit option can be changed. However, there weren't any programs out there which could actually change those special "geek" preferences, as the Finder engineers lovingly called them. (Because only really-really geeky people would ever want to change them, supposedly.) So, being one of the founding members of the fraternity of geeks, I naturally decided to take it upon myself to allow the geek preferences to be changed. The masterful result of my tiny bell is called, ever so lovingly, Geeker. (Or you can call it by its full technical name: "What do you get when you cross a geek with the Finder?")

Those Extra Preferences

There are five different Finder geek preferences you can change with Geeker. You can change all the default window positions (I count that as one thing that can be changed), the default folder color, the default comma character, the default quit setting, and the default trashcan position.

To change any of the settings, you first have to launch the Geeker application. (Geeker can even be run from your backup *GS+* Disk if you wish.) When Geeker finishes loading, choose the Open Finder menu item from the File menu. Once you've opened the Finder, you'll see that the Finder trashcan icon appears on your desktop. So, the first thing you might want to do is to change the trashcan position. To do this, simply click on the trashcan and drag it to its new location on the desktop. Too easy, huh?

OK, so now you might want to change some window positions. There are eight default window positions. To change a position for a window, you choose the window you want from the Windows menu. The window will then appear on the desktop and you can drag it around and resize it until you have it where you want. You can choose the Open All menu item from the Windows menu to open all of the default Finder positioning windows at once. (You can also choose the Close All menu item from the File menu to close all the open windows.) I've rearranged my windows so they open from the bottom left hand corner of the screen toward the top right hand corner—but you can change yours any way you want to.

The rest of the geek preferences are changed by choosing menu items from the Special menu.

The first menu item in the Special menu, the Folder Color menu item, brings up a dialog which lets you choose the default color for folders. Simply choose some new colors from the pop-up menus in the dialog and the folder icon will change to show you your new selections. When you have selected the colors you want, click the Accept button to make the change permanent.

The Comma Character menu item brings up a dialog which lets you choose the default comma character. The comma character is used when formatting numbers. For example, if there are five thousand files in a directory, the count would read "5,000" if the comma character were a comma. However, if you change the comma character to a period, the count would read "5.000" instead. To change the comma character simply type the character you want to use and click on the Accept button.

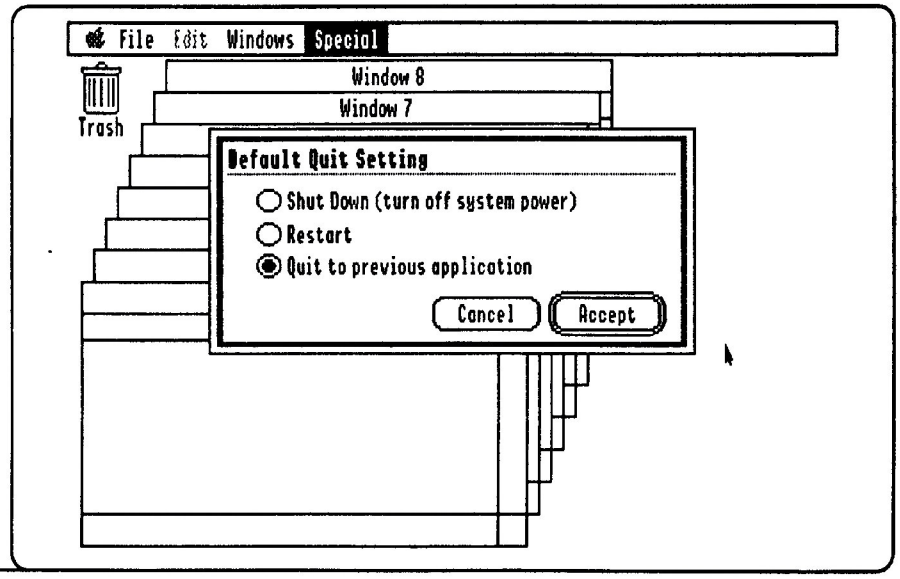
The third menu item, Quit Setting, lets you choose the default setting for the quit dialog. Of course this was the main reason why Geeker was written, but being a geek, I couldn't just write a program that would just change the quit setting and nothing else. Anyway, to change the quit setting, simply click on the setting you want to be the default and then click on the Accept button.

The last menu item, Return To Defaults, changes all the geek preferences back to their original values.

To make all of your geeky changes stick, you have to choose the Save menu item from the File menu. This actually writes the preferences out to the Finder. Note that the geek preferences are actually stored in the Finder's resource fork, not in some separate preferences file. The only other menu item with significance that I haven't discussed is the Close Finder menu item from the File menu. Choosing the Close Finder menu item is the counterpart for Open Finder. If you have two copies of the Finder lying around and you want to make changes to both of them, you can open the first one, make some changes, save the changes, close the first one, then open the second one and make changes.

Five or Six (to Four?)

If you happen to know what all of the geek preferences are, you'll notice that there are a few that Geeker doesn't allow you to change. One of those is the default settings for the "normal" Finder preferences. But, you can change these preference settings by using the Finder's Preferences menu item, so Geeker really doesn't need to be able to change them as well. There are also two window positions (for the Trash window and the Clipboard window) and the default desktop pattern preferences that Geeker



doesn't allow you to change. The reason for this is because the Finder doesn't actually honor those settings. However, if you hold down the option key when you start the Geeker application, then Geeker will allow you to edit those preferences, even though the Finder ignores them. The two window positions can be changed by choosing the Trash and Clipboard menu items from the Windows menu, and the default desktop pattern can be changed by choosing the Desktop Pattern menu item from the Special menu. (Remember, these menu items are not present unless you hold down the option key when Geeker starts up.)

Programmers Take Note

For the most part, writing Geeker was very straightforward. It's just a simple desktop application that has a few windows you can move around. The preference dialogs were simple enough to write as well. However, the real fun part was writing the code that allowed the trashcan to be positioned. I did this by installing a custom desktop drawing routine. The routine is called by the Window Manager whenever the desktop needs to be redrawn. First it draws the desktop, taking into consideration any custom pattern or picture in the message center, and then it draws the trashcan on the desktop if the Finder is opened. Whenever a hit is detected on the desktop, it is checked to see if it was over the trashcan, and if so, a grafPort containing the visible region of the desktop is switched in and `DragRect` is called to allow the dragging of the trashcan. Once the dragging is done, the desktop is redrawn to draw the trashcan in its new position. You can find information on how to write a custom desktop drawing routine yourself by looking in the *Apple IIGS Toolbox Reference*, volumes 2 and 3, and also by reading IIGS technical note #98, which is provided for you on your *GS+* Disk. Of course looking at some actual sample source code is great too, so the Geeker source code is heavily commented.

Even though Geeker is a fairly simple application, when we were testing it a strange problem arose. I called `Desktop` with a `deskTopOp` of 8 to tell the system that new drawing information was present before setting the new drawing procedure (actually the same draw procedure in order to force the desktop to redraw), as is outlined in IIGS technical note #98. That worked fine without a custom desk message in the message center, but after calling `Desktop` for the first time *with* a desk message in the message center, the desktop information (as returned with a `deskTopOp` of 4) was screwed up, so getting and setting the desktop drawing information would send the system off into the weeds. My solution was to first get the drawing information, then tell the system about the new drawing information, and finally set the new drawing information (i.e. call `Desktop` with a `deskTopOp` of 4 then 8 then 5 instead of 8 then 4 then 5). (If that description was too confusing, check out the source code.)

Another nasty problem cropped up when trying to change the default desktop color. As it turns out, the geek preferences resource stores the default desktop pattern in two rectangle structures. Since one rectangle is four words, and one word is two bytes, it turns out that the pattern is taking up sixteen bytes (since 2 rectangles times 4 words times 2 bytes equals 16). If you've read anything at all about QuickDraw II, you'll know that patterns take 32 bytes, *not* 16. However, I conveniently overlooked the mismatch in pattern sizes, and treated the saved preference as 32 bytes instead of 16. This caused some horrendous crashes when memory past the end of the geek preferences was overwritten .... And what was worse was the problem was only intermittent, since memory past the end of the preferences will not always be the same. Eventually the problem was narrowed down to the "for" loop that sets the new pattern in the geek preferences. That just screamed of having memory overwritten somewhere, but silly me, I looked at the code, and sure enough I was only moving 32 bytes—the size of a QuickDraw II pattern, which is what I wanted, right? Eventually it dawned on me that the geek preferences pattern was shorter than I had originally assumed. Now it's true that in 640 mode, only the first 16 bytes are used for a pattern unless bit 15 in the current port's `arcRot` field is set as outlined in IIGS technical note #6 (which is included on your *GS+* Disk), however the QuickDraw II pattern structure is explicitly 32 bytes. Of course the Finder can probably get away with making assumptions such as the window manager port always having bit 15 clear in its `arcRot` field since it's part of the System Software, however that's still a bad excuse in my book. If you work with a pattern, you work with 32 bytes—period. (Just for laughs, try setting bit 15 in the Window Manager's port `arcRot` field and look at the goofy patterns that result.) Anyhow, I had to bow down to the Finder and switch from using a 32-byte pattern to a 16-byte pattern inside the default desktop changing function. I'm still not very happy about it, though.

Who You Calling a Geek?

Chances are you're not as big a geek as I am, but I'm sure you'll still have lots of fun playing with Geeker and seeing exactly how funny looking you can get the Finder to be! *GS+*

Errata

In the product information accompanying the article "II Scary: Halloween Fun With Your IIGS", the phone number for *The MIDI Buying Guide* is missing a digit. The complete and correct phone number is (508) 531-6192.

If you find an error in *GS+* Magazine, please let us know about it! Give us a call at (615) 332-2087 or write to us at one of the following addresses ....

GEnie: Diz
Delphi: GSPlusDiz
Internet: Diz@genie.geis.com

Snail Mail:
GS+ Magazine
P. O. Box 15366
Chattanooga, TN 37415-0366

Apple II Software

APDA Moves West

Byte Works is now the exclusive distributor for Apple II APDA products. Produced by Apple's staff for Apple II developers, these products range from slick 4-color books to loose-leaf pages. Some items are in stock and some are printed on demand. Unless you check with us first, please allow 3-4 weeks for APDA orders.

ORCA/Modula-2

Niklaus Wirth designed Pascal for learning to program. When he wanted a systems programming language (like C) with great structure and safety (like Pascal) he created this highly acclaimed language—now available for the Apple IIGS. It's also the only GS language that supports multiprocessing directly.

Apple IIGS, 1.75M, hard drive.

Quick Click Calc

If you thought the spreadsheet in AppleWorks GS was a good start, you'll love our finish! Quick Click Calc is a true GS spreadsheet, featuring Publish & Subscribe, encryption, true split screen, unparalleled formatting, and a great business graphics package—all for a remarkably low price!

Apple IIGS, 1.125M

Apple IIGS Productivity

GS-21	Quick Click Calc	NEW	$60.00

Apple IIGS Programming Languages

GS-02	3D Logo	$60.00
GS-03	HyperLogo	$60.00
GS-04	ORCA/M Assembler	$75.00
GS-05	ORCA/Pascal Compiler	$95.00
GS-06	ORCA/C Compiler	$95.00
GS-07	ORCA/Modula-2 Compiler	$95.00
GS-08	ORCA/Integer BASIC	$30.00

Apple IIGS Programming Products

GS-09	Design Master	$40.00
GS-10	ORCA/Disassembler	$30.00
GS-11	ORCA/Debugger	$30.00
GS-12	Talking Tools	$35.00
GS-13	ORCA/SubLib Source	$25.00
GS-14	Merlin to ORCA	$25.00
GS-15	Utility Pack #1	$25.00
GS-16	Prog. Ref. for 6.0 & 6.0.1	$50.00

Apple IIGS Self-Study Courses

GS-17	Learn to Program in Pascal	$50.00
GS-18	Toolbox Prog. in Pascal	$75.00
GS-19	Learn to Program in C	$50.00
GS-20	Toolbox Prog. in C	$75.00

Other Apple IIGS Products

GS-01	Ugly Duckling Talking Storybook	$50.00

Apple II Programming Products

II-03	ORCA/M Assembler	$60.00
II-04	MON+ Debugger	$25.00
II-05	ORCA/M O/S Source	$25.00
II-06	Floating Point Libraries	$25.00

Other Apple II Products

II-01	Crypto (File Encryption)	$10.00
II-02	Byte Paint	$10.00

Upgrades (with proof of purchase)

ORCA/M 2.0.4 from...	NEW	
any Apple IIGS Assembler		$62.50
ORCA/M 1.x		$50.00
ORCA/M 2.0 or 2.0.1		$10.00
ORCA/M 2.0.2 or 2.0.3		$7.00
ORCA/Pascal 2.1 from...	NEW	
any Apple IIGS Pascal		$75.00
ORCA/Pascal 1.x		$60.00
ORCA/Pascal 2.x		$7.00
ORCA/C 2.0.3 from...	NEW	
any Apple IIGS C		$75.00
ORCA/C 1.x		$60.00
ORCA/C 2.x		$7.00
ORCA/Debugger 1.1.1		$10.00
ORCA/Integer BASIC 1.0.1		free† or $5
ORCA/SubLib Source		$10.00
3D Logo 1.0.2		free† or $5
HyperLogo 1.0.1		free† or $5
Prog. Ref. for System 6.0.1		$10.00

† Updates marked "free or $5" are free with any order, and cost $5 to cover postage and handling if ordered separately.

What Makes A Computer Tick?

Find out with our complete introductory programming courses. Learn to Program in C and Learn to Program in Pascal both cover the basics with a hands-on, friendly style and lots of examples. These courses also get you ready for toolbox programming.

Apple IIGS, 1.75M, hard drive recommended

Apple II APDA Products

APDA-01	Apple II Filecard Toolkit	$19.00
APDA-02	Apple II System 4.0.2	$14.00
APDA-03	Apple II Desktop Toolkit-Pascal	$30.00
APDA-04	Apple II SuperPILOT	$69.00
APDA-06	Apple IIGS System 6.0	$24.00
APDA-47	Apple IIGS System 6.0.1	$24.00

ORDERING To order directly from the Byte Works, please send your name, the shipping address, what you are ordering, and the total price. You can pay with VISA, MasterCard, or by personal check if the check is drawn on a U.S. bank for U.S. funds. C.O.D. is available in the U.S. at a slight additional charge. Schools and businesses listed with Dunn & Bradstreet may order with purchase orders.

SHIPPING For any order shipped to locations in the U.S. and Canada, include $5 for shipping and handling. If the product will be shipped outside the U.S. or Canada, you may specify how you would like the product shipped (air mail or surface) and we will charge you the actual shipping charges rounded up to the nearest dollar. We will also be happy to compute shipping charges for your order in advance.

30 DAY GUARANTEE In addition to any other guarantees that may apply, you may return any Byte Works product within 30 days for a full refund of the purchase price.

Gets the Job Done

Apple II Tech Notes

Where do you go for answers to detailed programming questions? According to the experienced pros giving advice on the online services, the Apple II Tech Notes! These essential references are used by the best Apple II programmers—people who make a living programming. Now you can learn their secrets firsthand! Get the full set with APDA-46, or get just what you need.

Write it Yourself!

Our toolbox programming course is available for C or Pascal. It's self-contained—all you need is the course and the language. Step-by-step, you'll learn how to write your own professional quality programs for the Apple IIGS. You'll also get a great library of four disks filled with examples of toolbox programs.

Apple IIGS, 1.75M, Hard Drive

APDA-07	Apple IIc Tech. Ref.	$30.00
APDA-08	Apple IIc Memory Expansion Card Ref.	$15.00
APDA-09	Apple II Memory Expansion Card Tech. Ref.	$15.00
APDA-10	Apple II High Speed SCSI Card Ref.	$15.00

HyperLogo GS

Here's the amazing graphics and artificial intelligence language built right into your favorite multimedia authoring tool! HyperLogo controls HyperStudio stacks, just like SimpleScript, but offers more commands, the power of artificial intelligence, and true 3D graphics!

Apple IIGS, 1.75M recommended, HyperStudio 3.0 or better.

APDA-11	Video Overlay Card Development Kit	$19.00
APDA-12	AppleShare Programmer's Guide for the Apple II	$30.00
APDA-14	XREF Apple II, Books and Notes	$20.00
APDA-15	Apple II GSBug and Debugging Tools Ref.	$30.00
APDA-16	MPW IIGS Assembler	$100.00
APDA-17	MPW IIGS C (Limited Supply!)	$150.00
APDA-19	MPW IIGS Tools	$50.00
APDA-20	Apple IIGS Source Code Sampler	$20.00
APDA-21	Apple II High-Speed SCSI Card Utilities	$30.00
APDA-22	DOS 3.3 User's Manual	$20.00
APDA-23	DOS 3.3 Programmer's Manual	$20.00
APDA-24	Apple II Pascal 1.3	$69.00
APDA-25	Desktop Toolkit (ProDOS)	$30.00
APDA-26	IIGS Firmware Reference 1 MB IIGS Update	$12.00

3D Logo

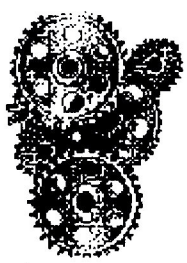
Create true 3D drawings that literally appear in front of the monitor. Make a movie of a molecule rotating in space. Build simple desktop programs without learning toolbox programming. Do it all with a language so powerful it's used in universities, and so simple it's used in elementary schools!

Apple IIGS, 1.125M

APDA-27	GS/OS Device Driver Reference	$29.00
APDA-30	HyperMover	$15.00
APDA-31	Apple IIGS System 6.0 Release Notes (disk)	$4.00
APDA-33	Apple IIGS System Disk 5.0.4	$24.00
APDA-34	HyperCard IIGS Developer's Kit	$15.00
APDA-35	Apple II Appleshare Setup	$15.00
APDA-37	APW & MPW Interfaces for System 6.0.1	$20.00
APDA-40	File Type Notes	$25.00
APDA-41	Classic Apple Notes	$20.00
APDA-42	IIGS Notes	$30.00
APDA-43	Misc. Notes	$10.00
APDA-44	Apple II Pascal Notes	$4.00
APDA-45	HyperCard IIGS Notes	$3.00
APDA-46	Apple II Tech Notes (Full Set; APDA40-45)	$65.00

Addison-Wesley Books

AW-04	Apple IIGS Hardware Ref.	$26.95
AW-08	Apple IIe Tech. Ref.	$24.95
AW-09	Applesoft Tutorial	$29.95
AW-05	Prog. Intro. to the IIGS	$32.95
AW-06	Technical Intro. to the IIGS	$9.95
AW-10	Applesoft BASIC Programmer's Ref. Manual	$22.95
AW-07	Apple IIGS Firmware Ref.	$24.95
AW-11	ProDOS 8 Tech. Ref.	$29.95
AW-12	ImageWriter LQ Reference	$22.95
AW-13	LaserWriter Reference	$19.95
AW-14	Understanding Computer Networks	$9.95
AW-15	Planning and Managing AppleTalk Networks	$18.95



Byte Works, Inc.
8000 Wagon Mound Drive N.W.
Albuquerque, NM 87120

Phone (505) 898-8183
AOL e-mail to MikeW50
GEnie e-mail to ByteWorks
Internet e-mail to MikeW50@AOL.COM

Elucidation

By Josef W. Wankerl

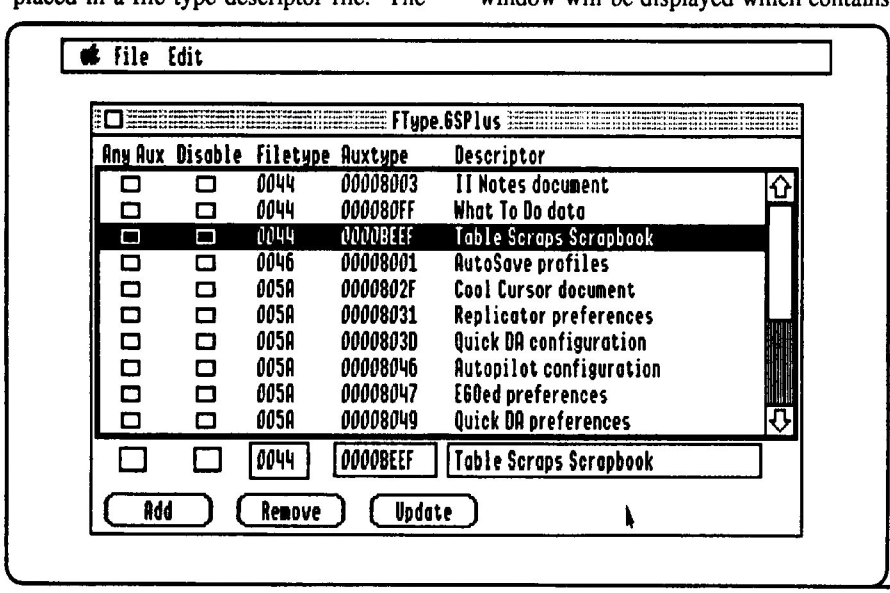
Have you ever wondered how the Finder knows exactly what a file is when you select the "Icon Info" menu item? There always seems to be a nice text string which describes, with amazing accuracy, what the file is. In order to do this, the Finder relies on special files, called file type descriptor files, which reside in your ***:Icons** folder. (Yes, the file type descriptor files must reside on your boot disk in order to be searched. The ***:Icons** folder is the only folder which will be searched. Any **Icons** folders on other disks will not be used to look for file type descriptor files.) The file type descriptor file which comes with System 6.0.1 is called **FType.Apple** and it contains all of the descriptions for all the defined file types Apple had assigned when System 6.0.1 came out. Whenever the Finder needs to know what a file is, it looks through all the file type descriptor files until it finds a match, then it displays the description string which corresponds to the matched file. Now that Apple appears to be out of the IIGS system software game, the Apple-supplied file type descriptor file does not know about all the new IIGS file types that have been assigned since System 6.0.1 was released. The result is that the Finder displays a generic "Unknown" string for these newer file types. So, in order to combat this, I wrote a simple little program which allows you to create your own file type descriptor files.

About Descriptors

To tell the Finder (or for that matter, *any* program which uses file type descriptor files—the Finder isn't the only program that can) about new file types, an entry is placed in a file type descriptor file. The entry basically consists of a file type, an auxiliary type, some special flags, and a string describing any files that match the file type and auxiliary type. When a program needs to know what a file is, is looks through all the file type descriptor entries until it finds a file type and auxiliary type match. There are a few special cases though, and they are handled with the special flags. Currently, five flags are defined, but only two of them are supported. The first supported flag allows a match to be based only on the file type and any auxiliary type will be allowed. (For example, every file with a file type of $0004 is considered a text file no matter what its auxiliary type is.) The second supported flag means that the file can be made inactive by setting bit 15 in its auxiliary type. (For example, new desk accessories can be made inactive.) For an in-depth discussion of the file type descriptor file, you should read the corresponding file type note, FTN.44.xxxx, which is provided in the self-extracting archive on your *GS+* Disk. So, now that you know what makes up a file type descriptor, let's take a look at how to actually create some descriptions with the Elucidation application.

FType.Apple

The first thing you might want to do when you run the Elucidation application is to take a look and see what Apple's file type descriptor file looks like. To do this, choose the Open menu item from the Elucidation File menu, then find the **FType.Apple** file in the **Icons** folder of your boot disk. After a bit of work (it takes some time to process all the entries in large file type descriptor files) a window will be displayed which contains all of Apple's file type descriptions. You can browse through them, and you can even make changes if you want. However, it is not a good idea to make changes to the **FType.Apple** file. If you want to override the descriptions in the **FType.Apple** file, you should make a *new* file type descriptor file and add your corrected description to the new file. Files created with the Elucidation application will *always* be looked at before the **FType.Apple** file, so your modifications will take precedence over the Apple supplied descriptions. (File type descriptor files are searched based on their auxiliary type. The order that file type descriptor files are searched is described with great detail in file type note FTN.44.xxxx, which is provided in the self-extracting archive on your *GS+* Disk. Elucidation always creates file type descriptor files with an auxiliary type of $0200. If you want to have your file type descriptor file ordered differently, you can use any program, such as our own NoDOS from *GS+* V4.N4, which allows you to change the auxiliary type of a file.)



Making Your Own

Once you've gotten a feel for what a file type descriptor file should look like, you can start making your own. To start, you choose the New menu item from the File menu to create a new file type descriptor edit window. To add a new description, you put the appropriate information in the edit area at the bottom of the window and then click on the Add button. To remove a description, select the description you want to remove in the list and then click on the Remove button. To change a description, select the description you want to change in the list, make your changes to the description in the edit area at the bottom of the window, and then click on the Update button. If all you want to do is toggle one of the check box values, you can simply double-click on the check box in the list and the value will be changed. (Note that if you turn on the match any auxiliary type check box, the auxiliary type will automatically be set to zero.)

I used Elucidation to create a file type descriptor file which contains information about all the file types that have been assigned to *GS+* Magazine. The file, **FType.GSPlus**, is included on your *GS+* Disk, and you can install it in your system using the supplied Installer script. If you ever need to play with file type descriptions, Elucidation is invaluable! *GS+*

Copy Icon

By Josef W. Wankerl

Once again, I've managed to write another program revolving around the system clipboard. (Programs in the past few issues which dealt with using the system clipboard have been Clip On and Sun Dial in V5.N6, and Table Scraps and Scrapie in V6.N1.) With Copy Icon, you can select any icon the Finder displays and copy it to the system clipboard.

Using Copy Icon

Copy Icon is a Finder extension which appears in the Finder's Extras menu. When one or more icons are selected in the Finder, the Copy Icon menu item is enabled. When you choose the Copy Icon menu item, the selected icons are copied to the system clipboard. You can then launch your favorite icon editor and paste those icons into a new document. (Note that although Copy Icons will copy multiple icons to the system clipboard, many icon editors were written to only see the very first icon on the clipboard, so don't be too disappointed if you can't access all of your copied icons. You can always copy and paste your icons one by one if you have to. Our Table Scraps scrapbook new desk accessory can be a big help in this situation.) In addition to putting icons on the clipboard, Copy Icon also places a picture of the first selected icon on the clipboard as well. You can then go into any application which can paste pictures (like Platinum Paint) and paste the picture of the icon. That's all there is to know in order to use Copy Icon.

What You Can do With It

Being able to copy icons from within the Finder can be useful for a number of reasons. One such reason would be that you want to slightly change the appearance of an icon the Finder displays. All you have to do is copy the icon to the clipboard using Copy Icon, paste the icon into your favorite icon editing program, make your changes, and then save the icon out to a new icon file. Without Copy Icon, you'd have to browse through all the icon files until you found the icon, then edit it. You can also use Copy Icon and your favorite icon editor to easily create new icon files that contain only the icons that are needed for each of your disks. Finally, if the icon you want to modify is contained in an invisible **Desktop** file, you would never be able to find and edit the icon by using one of the older icon editors. However, using Copy Icon, you can make an old-style copy of that icon and very easily edit it with an old icon editor.

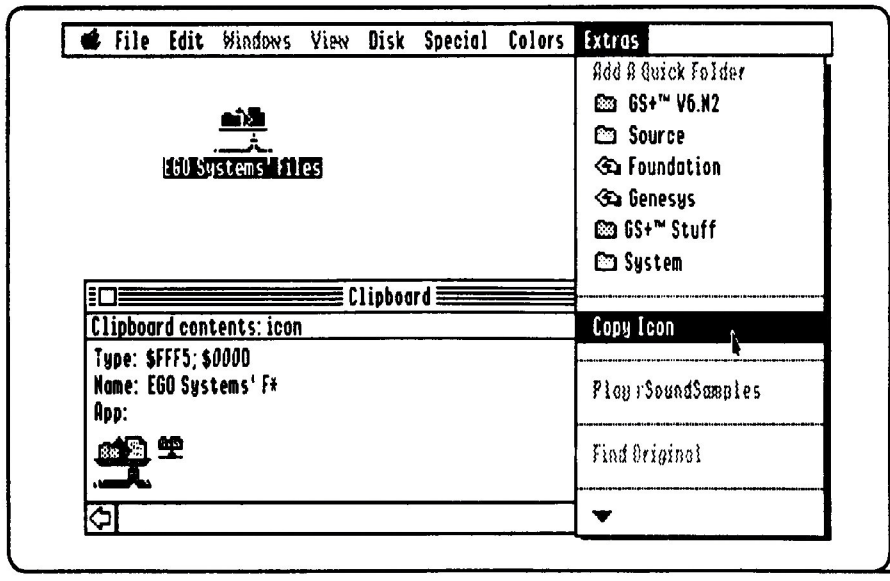
Another reason you might use Copy Icon would be to paste an icon image into a paint program, word processor, or page layout program. Using icons in your documents can greatly help to illustrate a point, especially if you are writing documentation for a new program.

Reading the Source Code

Most of the Copy Icon source code is straightforward. The majority of the code deals with keeping up with the Finder environment. There are only three routines which actually deal with copying things to the system clipboard: `CopyIcon`, `CopyPicture`, and `CopyIcons`. When the Copy Icon menu item is chosen from the Finder's Extras menu, the `CopyIcons` routine gets called. The `CopyIcons` routine gets all the selected icons, clears out the current system clipboard contents, and then loops through each selected icon. For each selected icon, the `CopyIcon` routine is called to place a copy of the icon on the system clipboard. Because the `PutScrap` Toolbox call simply appends data to the clipboard, adding all the icons is made easier by repeatedly calling `PutScrap` instead of building one big scrap and then adding it to the clipboard once all the icons have been looked at. The format for the icon scrap type is described by IIGS Technical Note #99 and by IIGS File Type Note $CA, which are present in the self-extracting archive on your *GS+* Disk.

For the very first selected icon, the `CopyPicture` routine is also called. The `CopyPicture` routine places a picture of an icon on the system clipboard. The format for the picture scrap type is a QuickDraw II picture. Actually creating the picture from an icon was a little more difficult than I had first imagined. To create a picture, I thought I could just call `OpenPicture` to start the picture definition, call `DrawIcon` to draw the icon in the picture, and then `ClosePicture` to finish the picture definition. It turns out, however, that the `DrawIcon` call does not add anything to a picture definition! What I had to do instead was open a new off-screen grafPort, draw the icon to the off-screen grafPort, open a new picture, copy the pixels from the off-screen grafPort back onto themselves using the `PPToPort` call (which *is* recorded in the picture), and finally close the picture. After the picture is closed, the handle which was returned from the `OpenPicture` call then contains the actual QuickDraw II picture data, so it is simply copied to the system clipboard. Since putting items on the system clipboard actually puts a copy on the clipboard, the original picture handle is disposed of since it is no longer needed.

That's pretty much everything to Copy Icon. I hope you find that being able to simply copy icon images by selecting them from the Finder can be very useful. *GS+*



What's New?

Compiled by Steven W. Disbrow

"Ask and ye shall receive." Apparently this old saying is true, because I've been asking for press releases, and I've been getting them. So, let's take a look at what's come to us over the last couple of months. (I know, some of this is old news, but we didn't have room for "What's New?" in the last issue. Sorry!)

News That Doesn't Byte

Just in case you missed their big ad in our last issue, here's some important news from the Byte Works. First of all, they are now the official home of all the Apple II development products that were produced by APDA.

The second announcement from the Byte Works is that, after over a year of waiting, they have decided that Apple Computer, Inc. will never deliver the promised "Apple Tools" update to them. (This was supposed to include updates to GSBug, REZ, and other Apple-created IIGS development tools.) So, they have cancelled all pending orders for this product. This, of course, stinks. But, it certainly isn't the fault of the Byte Works!

Third, the Byte Works recently announced new updates to ORCA/M, ORCA/Pascal and ORCA/C! The latest version numbers for these products are: ORCA/M - 2.0.4, ORCA/Pascal - 2.1 and ORCA/C - 2.0.3. Each update is only $7.

Finally, the Byte Works announced that they have lowered their direct-purchase prices on almost all of their products! For example, ORCA/Pascal used to cost $150. Now, if you buy it direct from the Byte Works, it's only $95! To get complete information and price lists, contact:

The Byte Works, Inc.
4700 Irving Blvd. N.W. Suite 207
Albuquerque, NM 87114
(505) 898-8183
GEnie: ByteWorks

Oh so Blue (Disk)

Did you know that Apple recently discontinued the Apple II SuperDrive Controller card? Well, it doesn't really matter, because SHH Systeme has just released the BlueDisk controller card for the IIGS. This card gives you the same capabilities as the old SuperDrive Controller—and then some.

Like that other card, the BlueDisk card lets your IIGS work with High Density disks for both the Mac and PC. However, unlike the SuperDrive card, the BlueDisk controller lets you use inexpensive MS-DOS floppy disk drives! If you shop around, these drives can be had for under $100! (Compare that price to Apple's $350 price tag for a SuperDrive!) As if that weren't cool enough, the BlueDisk also lets you use the new 2.88MB "Extra Density" floppy drives and disks on your IIGS! But hey, there's more! The BlueDisk controller also . . . .

- Is compatible with the PC Transporter
- Allows you to format and use ProDOS and HFS disks (up to 2.88MB)
- Includes Peter Watson's MS-DOS utilities (SHH Systeme even pays the shareware fee!)
- Handles up to two drives per card

The BlueDisk controller is available *now* for only $139 (plus $14 shipping). Since the dollar is getting weaker every day, you should contact SHH Systeme for the latest pricing information.

SHH Systeme
Dipl. Ing. Joachim Lange
Bergstrasse 95
82131 Stockdorf
Germany

Phone: + 89 + 8577040
Internet: JLANGE@TASHA.MUC.DE

Keep In Touch

A couple of issues ago, we ran an announcement that Joe Kohn's Shareware Solutions II had taken over publication of the Contacts GS new desk accessory. Well, it was recently announced that the new version of Contacts GS is finally available, along with a new manual and a new AppleWorks enhancement called "TimeOut ContactsMover."

The Contacts GS NDA allows you to create and maintain a database of names, addresses and phone numbers from inside any IIGS program that supports new desk accessories (i.e. programs like the Finder, AppleWorks GS, etc.).

Using TimeOut ContactsMover, you can easily import the information you type into the Contacts GS desk accessory directly into AppleWorks. And, if you want to later use that data in Contacts GS again, TimeOut ContactsMover lets you easily export the data out of AppleWorks.

If you had previously purchased Contacts GS from Simplexity Software, you can upgrade to this new version for only $10 (plus $3 for shipping in the U.S. or $5 for shipping to anywhere else). (Note that if you are upgrading, you must send in your original disk.) If you don't already own Contacts GS, the price is $35 (plus the shipping costs shown above). To get more information, contact:

Shareware Solutions II
166 Alpine Street
San Rafael, CA 94901-1008

Internet: joko@crl.com

Do You GNO/ME?

As mentioned in this issue's "Writer's Block" EGO Systems has begun carrying the complete line of Procyon products. These products include:

- GNO/ME ($89) - A Unix-like shell for the Apple IIGS that provides (among other things), pre-emptive multitasking. (See review in *GS+* V5.N3.)
- Pick 'n' Pile ($20) - An infuriating and addictive "Tetris-like" game that's a blast to play. (Reviewed in *GS+* V4.N5.)
- Splat! ($39.95) - A source code level debugger for the ORCA programming languages. (Reviewed in *GS+* V5.N4.)
- Switch It! ($39.95) - An application switcher utility that lets you have more than one IIGS application active and switch between them instantly.

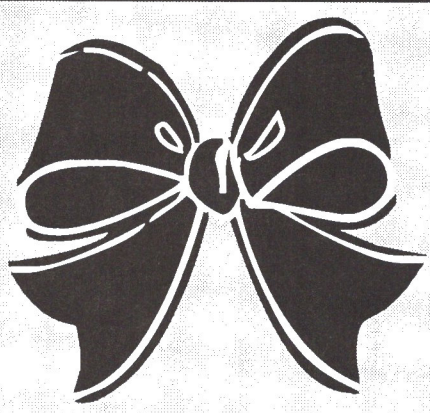
You can order these products from us just like you would order any of our other products: Just give us a call or send us a letter and we'll take care of the rest. The only other thing to tell is that if you are ordering from overseas, you should include $10 extra to cover shipping on GNO/ME, add $3 for shipping on Pick 'n' Pile, and add $4 for shipping on either Splat! or Switch It!

For more technical information on any of these Procyon products, check out the reviews we've done, or contact Procyon directly by sending e-mail to "bazyar@netcom.com". Or, give us a call here at (615) 332-2087.

Hey You!

Do you have a new Apple IIGS product or service that you want the world to know about? If so, let us know and we'll let our readers know too! Send your press release to us at:

GS+ What's New?
P. O. Box 15366
Chattanooga, TN 37415-0366



To: All Apple IIGS Users

From: Seven Hills Software

Super Holiday Specials!

Thank you for your support in 1994; we wish you the best in 1995! Your purchase of our products lets us continue publishing great software for your IIGS.

Spectrum™ ——— $64.98
System 6 *See GS+V5N5* $129.95
Incredible IIGS-specific telecommunications program that gets you connected to the world! Comes complete with a powerful English-based scripting language and Online Displays (emulations) for ANSI, ProTERM Special, VT100, and others. Many more features.
Spectrum makes a great gift for any modem user!

Kangaroo™ ——— $24.98
System 5 or 6 *See GS+V4N1* $49.95
Navigating the Standard File (SF) dialog boxes has never been easier! After choosing "File/Open" you can instantly hop to any folder or file you've used recently. Or use the "sub-menu" feature to completely bypass the SF dialog! Also lets you rename, delete, etc., right in the SF dialog.
Use it for a week and you'll wonder how you ever managed without it!

The Manager™ ——— $34.98
System 6 *See GS+V4N4* $69.95
Do you frequently switch between two IIGS applications, such as the Finder and GraphicWriter III? If so, The Manager will save you tons of time because it keeps multiple applications open at the same time, and simply clicking in an inactive window instantly re-activates the program!
An amazing piece of software—dazzle your friends and family!

GraphicWriter III™ — $74.98
System 6 *See GS+V1N5* $149.95
The best choice for desktop publishing on the Apple IIGS. Create professional newsletters, eye-catching fliers, imaginative instructional materials, business forms, and practically anything else. Includes free disk of fonts and two free clip art disks!
GraphicWriter III, Express, and a DeskJet with Independence... the perfect gift bundle!

Holiday Bonus
Drive Cleaner GS
software only for
FREE!
with the purchase of any item in this ad - when you place your order just tell us you saw this ad in GS+!

Super Menu Pack™ — $14.98
Key caps/font menu utility $29.95

ShoeBox™ ——— $29.98
Home info organizer *GS+V3N5* $59.95

Font Factory GS™ — $19.98
Bitmap font editor $39.95

Independence™ ——— $19.98
HP printer drivers *GS+V2N4* $39.95

Drive Cleaner GS™ — $17.48
3.5" disk drive cleaner kit $34.95

Express™ ——— $24.98
System 5 or 6 *See GS+V3N5* $49.95
Stop waiting for your printer! Express quickly saves your print jobs to disk then returns computer control to you while it prints your documents in the background. Multiple files can be spooled to disk. Spooled files can remain on disk to be reprinted in the future without even launching the application that was used to print the document.

TransProg III™ ——— $19.98
System 5 or 6 *See GS+V4N3* $39.95
Create a custom menu that appears in the menu bar of IIGS specific applications—just choose an item to launch, the current application quits, then the new application is launched directly. Change slot settings on-the-fly, establish default "data paths," change the system speed, and so on.
If you like our Quick Launch Finder Extra, you'll love TransProg III!

SuperConvert™ ——— $19.98
System 5 or 6 *See GS+V3N3* $39.95
Convert a huge variety of picture formats into images you can use on your IIGS! Includes a powerful "remap" command that lets you set the image type, palette type, rendering method, and size for the final picture. Converted images can be printed from SuperConvert with a variety of options, including the ability to print wall-size posters. Many more "extras."

Hurry—holiday specials end 12/22/94!
Checks-MOs-Visa-Mastercard-Discover

All products require an Apple IIGS with 1.25MB memory. Hard drive recommended.

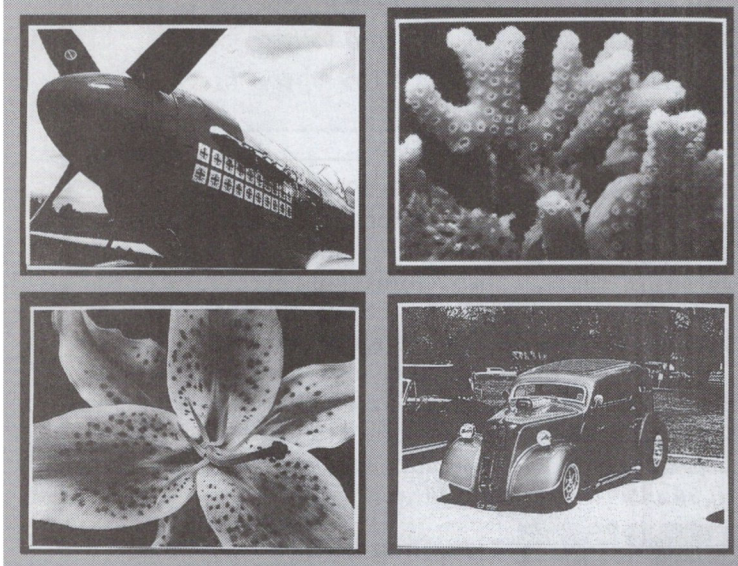
Seven Hills
SOFTWARE
2310 Oxford Road
Tallahassee, FL 32304-3930

America Online, GEnie: *SevenHills*
CompuServe: *75300,1743*
Internet: *sevenhills@aol.com*

(904) 575-0566, M-F, 9am-5pm ET
FAX (904) 575-2015 anytime

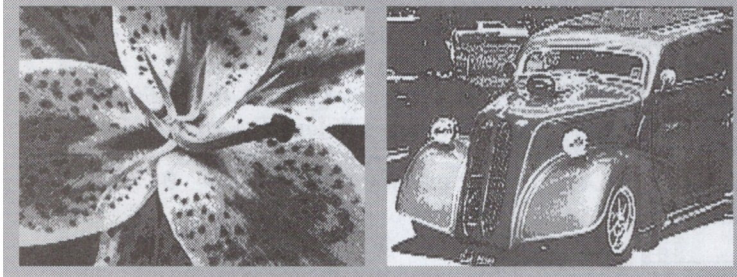
Brilliant Color on your Apple IIGS?

YOU BET!



Actual unretouched display photographs of an Apple IIGS with a Second Sight video card and a VGA monitor.

Same images displayed on a standard Apple color monitor without a Second Sight video card.



- **Uses standard VGA or Apple Color Monitor**
- **Display a palette of 256 colors from 16M**
- **Supports Apple II Text and Graphics modes**
- **Includes graphics display software for TIFF, GIF, PICT and KODAK Photo CD images**
- **Ideal for use with DiscQuest, DiscQuest Encyclopedia and CD-ROM text and images**

Finally, state-of-the-art video for your Apple IIGS. Whether you're using discQUEST™ to view HiRES graphics from CD-ROM or typing a text document in GS/OS it will look better than ever. Use your existing color monitor, add a new VGA or both! For your IIGS, it's Second Sight!

SECOND sight™

High-Resolution 8-bit color video card for Apple IIGS computers

For more information call:

SEQUENTIAL SYSTEMS

1-800-759-4549

America's Leading Apple II Source

Working With the Toolbox Part 14: The TextEdit Tool Set

By Josef W. Wankerl

To follow up on my Font Manager coverage in the last issue, I thought that TextEdit would make a nice tool set to pick on for this time. If you need to display a large area of styled text, you can use the LineEdit tool set's `LETextBox2` tool call. However, if you want to *edit* styled text, you'll probably turn to the TextEdit tool set. Before we dive in to using TextEdit, there are a few basic concepts you'll need to get straight:

Concepts

For simple text editing, I can't think of any other tool that beats TextEdit. However, the TextEdit tool set does have its limitations. You should know from the start that the TextEdit tool set is *not* meant to be used as a word processor, although it has many features that are found in word processors. As its name implies, the TextEdit tool set is mainly for editing styled text. The boundaries between a text editor and a word processor are very fuzzy and difficult to define, but here's the way I see it: A text editor's main concern is the text—the actual characters that make up a document. On the other hand, a word processor has to worry about text as well as how the text will fit on a printed page. The reason you would use a word processor would be to make a document that would ultimately be printed. The reason you would use a text editor would be to create or edit some text, perhaps for later importation into a word processor or page layout program. The text that a text editor works with may not ever make its way to the printed page. That's the main difference: A word processor document is made with printing in mind, a text editor document is made with only the text content in mind. The TextEdit tool set could very well be made into a decent word processor tool, but don't hold your breath waiting for this to happen. Some of the main limitations on the TextEdit tool set which prevent it from being a word processing tool are:

- The entire document must have the same justification (left, right, center, or full)

- The entire document must have the same margins and tab stops

- There is no way to insert a forced page break into the text

- Horizontal scrolling is not supported (which would be necessary to see all of the text that would fit on the page horizontally, but would not necessarily fit on the screen at one time)

Reading Up

First off, you should sit down with your copy of the *Apple IIGS Toolbox Reference: Volume 3*. The TextEdit tool set is described in its entirety in this volume. Trying to use the TextEdit tool set without this information is to play a guessing game that you'll almost certainly lose. You'll also want to be familiar with the Control Manager chapter in volume 3 as well, since it describes how TextEdit controls work. The *Programmer's Reference for System 6.0* and *6.0.1* don't contain any new information about the TextEdit tool set, just a few clarifications. Of course, if you're doing any kind of programming with the IIGS Toolbox, you should have every last bit of reference material you can lay your hands on.

Basics

There are two ways to use TextEdit in your programs. One way is by exclusively making calls to the TextEdit tool set. The other way is to create a TextEdit control. Using a control is, by far, easier than doing the job yourself, so everything in this article assumes you're using a TextEdit control. I can't think of a good reason why anyone would decide to avoid TextEdit controls in favor of doing all the housekeeping work themselves. For example, without using a TextEdit control you'd have to call `TENew` to create a new control, call `TEActivate` and `TEDeactivate` to keep the active state for the TextEdit record consistent with the desktop (i.e. if the window a TextEdit record in is not the frontmost window, the record should be inactive), call `TEUpdate` whenever the record needs to be updated, call `TEClick` when a mouse down event is detected in the record, call `TEKey` when a key is pressed, and call `TEIdle` to keep the cursor flashing consistently. Using a TextEdit control lets you comfortably ignore all the details of keeping up with the control state and just worry about the task you're actually trying to accomplish by using the TextEdit control. That's about the only difference between using a TextEdit control and a TextEdit record created by `TENew`. Since just about every TextEdit call takes as a parameter a handle to a TextEdit record, you can pass either a handle to a record created by `TENew` or a handle to a TextEdit control as that parameter.

The TextEdit tool set does a lot of work to let you edit styled text. You'd think that for such a complex tool set, there would be a lot of complex calls you could make. This, however, is just not true. There are relatively very few TextEdit tool set calls and they are easy to make. Basically, there are three things that you will do with a TextEdit control: Put styled text in, get styled text out, and modify the styled text that's already in the control. Before we can discuss how to do this, it's helpful to know a little bit about some of the data structures that the TextEdit tool set uses to keep up with styled text.

TextEdit Data Structures

There are a number of data structures that the TextEdit tool set uses to associate style information with text. The highest level structure is called the `TEFormat` structure. The `TEFormat` structure defines the ruler and style changes for the entire TextEdit document. Contained in the `TEFormat` structure are a bunch of other information structure pieces. The first piece is the `TERuler` information. Currently the TextEdit tool set can only handle one ruler per document. The `TERuler` structure controls the settings for the left margin, left indent for new paragraphs, right margin, text justification, extra line spacing, and tab stops. The second piece is the style list information. The style list information contains a list of every single unique text style that is contained in the TextEdit document. Each unique style is represented in a `TEStyle` structure. The `TEStyle` structure has settings for the font, size, style, foreground color, and background color. The last piece is an array of `StyleItem` structures which specify which `TEStyle` structures apply to which areas of text. The `StyleItem` structure has two fields: One field for the number of characters a style applies to, and one field to reference the `TEStyle` structure containing the style to apply to the text. This may all sound a bit confusing, so to help illustrate the point, I'll give an example of how this works:

First off, the ruler information is straightforward and can be ignored for this example. The fun part is the interaction between the text, the `StyleItem` structures, and the `TEStyle` structures. For example, let's look at the styled text:

"The **TextEdit** tool set is groovy."

Important TextEdit Data Structures

TEFormat (page 49-31 of TB Ref vol III)

Offset	Field Name	Size
$00	version	Word (2 bytes)
$02	rulerListLength	Long (4 bytes)
$06	theRulerList	Array of TERuler's
$xx	styleListLength	Long (4 bytes)
$xx	theStyleList	Array of TEStyle's
$xx	numberOfStyles	Long (4 bytes)
$xx	theStyles	Array of StyleItem's

TERuler (page 49-39 of TB Ref vol III)

Offset	Field Name	Size
$00	leftMargin	Word (2 bytes)
$02	leftIndent	Word (2 bytes)
$04	rightMargin	Word (2 bytes)
$06	just	Word (2 bytes)
$08	extraLS	Word (2 bytes)
$0A	flags	Word (2 bytes)
$0C	userData	Long (4 bytes)
$10	tabType	Word (2 bytes)
$12	theTabs	Array of TabItem's
$xx	tabTerminator	Word (2 bytes)

TabItem (page 49-59 of TB Ref vol III)

Offset	Field Name	Size
$00	tabKind	Word (2 bytes)
$02	tabData	Word (2 bytes)

TEStyle (page 49-41 of TB Ref vol III)

Offset	Field Name	Size
$00	fontID	Long (4 bytes)
$04	foreColor	Word (2 bytes)
$06	backColor	Word (2 bytes)
$08	userData	Long (4 bytes)

fontID (page 8-6 of TB Ref vol I)

Offset	Field Name	Size
$00	famNum	Word (2 bytes)
$02	fontStyle	1 Byte
$03	fontSize	1 Byte

fontStyle bits (page 8-5 of TB Ref vol I)

Bit	Meaning
0	Bold
1	Italic
2	Underline
3	Outline
4	Shadow
5-7	Reserved

StyleItem (page 49-55 of TB Ref vol III)

Offset	Field Name	Size
$00	length	Long (4 bytes)
$04	offset	Long (4 bytes)

Now, I'll define two styles for this text: Style A will be Times 10-point plain, and style B will be Times 10-point bold. If I want the word "TextEdit" to have the bold style and everything else to have the plain style, the array of `StyleItem` structures would be: Style A applied to 4 characters (which corresponds to "The "), style B applied to 8 characters (which corresponds to "TextEdit"), and style A applied to 20 characters (which corresponds to " tool set is groovy."). Note that every single character, including spaces and punctuation, has a style associated with it.

Luckily, you rarely have to deal with these structures yourself—you just make calls to TextEdit and the TextEdit tool set will manipulate the data appropriately.

Getting Text In

Before you can work with text, you need to put some text in a TextEdit control. There are a number of ways to accomplish this. The first way is to put references to the text and style information in the TextEdit control template. When the control is created, the referenced text and style information will be used to set up the initial text. Another way to is create an empty control and then use the `TESetText` call to place the text in the control. (You can make a `TEInsert` call as well, but `TEInsert` does not clear out the previous contents of the control and it does not change any ruler information.)

Getting Text Out

If you have some text in a TextEdit control and you want to see what it is, you use the `TEGetText` call. The best way to make the `TEGetText` call is to have it allocate the handles for the text and style information that it returns to you. Once you have the text and style information, you are free to manipulate it in any way you want. (If you had `TEGetText` allocate handles for you, you have to dispose of them when you are done with them.) If you have some text selected in the control, you can set the `onlyGetSelection` flag bit so the `TEGetText` call only returns the text that is selected. However, if you get the style information as well, then the style is for the *entire* text, not for the selection only. (If you want to get the text and style for the selection only, there is a Miscellaneous Library call you can make [`TEGetSelectText`] that will do this for you.)

Other Operations

One of the most influential parts of a TextEdit control is the current selection. The current selection specifies the characters of text that certain operations are performed on. You can get and set the selection using the `TEGetSelection` and `TESetSelection` calls. (And, of course, the user can set the current selection by selecting some text.) Once the selection is properly set up, you can copy it to the system clipboard (`TECopy`), clear it (`TEClear`), cut it (`TECut`), replace it with text from the system clipboard (`TEPaste`), replace it with any other text (`TEReplace`), find out what styles are applied to it (`TEGetSelectionStyle`), and even change its style (`TEStyleChange`). Most of the operations are self-explanatory, however the last two dealing with styles could use a bit more explanation.

The `TEGetSelectionStyle` call returns a flag word specifying the common style elements for all the selected text and a list of all the unique styles (i.e. the unique `TEStyle` structures) in the selection. It also returns a separate `TEStyle` structure that contains all of the common style elements. So if the flag said that the font size and foreground color were common to everything in the selection, all you have to do is look at the common `TEStyle` structure to find out what the size and foreground color is for the entire selection.

The `TEStyleChange` call allows you to change any aspect of a style in the current selection. The `TEStyleChange` call requires a flag word which specifies what parts of the style information should change and what parts should be left alone. Changing the font attributes can be done by completely replacing the attributes with a new attributes byte or by toggling individual bits in the byte. Toggling individual bits makes it easy for you to implement a Style menu where, for example, choosing the Bold style menu item toggles the bold attribute. Replacing the attributes completely makes it easy for you to completely change the attributes when you choose the Plain menu item or choose a completely different font via the Choose Font dialog.

Rulers

As mentioned before, the `TERuler` structure controls the settings for the left margin, left indent for new paragraphs, right margin, text justification, extra line spacing, and tab stops. You can get and set ruler information using the `TEGetRuler` and `TESetRuler` calls. Note that the `TESetText` tool changes the ruler information as well. Text justification can be left, right, center, or full justification. There are three types of tab stops that can be set: No tab stops, evenly spaced tabs, and absolutely placed tabs. With no tab stops, there is no tab stop data. With evenly placed tabs, tabs stops are evenly placed by a certain amount of pixels you set (i.e. there is a tab every 10 pixels). With absolutely placed tabs, an array of absolute pixel positions for tab stops are required to identify the tab stop positions. All tabs are left justified tabs, meaning that text will be left justified at the tab position. Right, center, and decimal tabs are not supported by TextEdit.

You should pay special attention to the margin values. If the right margin goes past the right-hand side of your TextEdit control, then the text that goes past the right-hand side of the control will never be able to be seen on-screen because TextEdit controls do not support horizontal scrolling.

Miscellaneous

There are a few miscellaneous calls you can make to TextEdit that cause special things to happen. One of my favorites is the `TEScroll` call. The `TEScroll` call causes the content area of the TextEdit control to scroll. You can scroll to an absolute character position, an absolute line, an absolute number of pages down, or a relative pixel position. You can also have the scroll destination appear at the top or center of the control (except for absolute page down and relative pixel positions—they always appear at the top of the control).

Two calls that I have never used in all my years of working with TextEdit are the `TEPointToOffset` and `TEOffset-ToPoint` calls. They convert absolute pixel positions to character offsets in the text. Although I have never used them myself, it's always nice to know that they're there for you in case you find some obscure use for them.

You can take the text in a TextEdit control and arbitrarily paint it into another grafPort by using the `TEPaintText` call. This is useful if you need to display the contents of your control and not allow any editing. This call can also be used to print the text—just open a printing grafPort with the Print Manager and then paint the text into the printing grafPort.

If you ever need to find out some statistics on a TextEdit control, you can make the `TEGetTextInfo` call. The `TEGetTextInfo` call will tell you how many characters are in the control, how many lines the characters take up, how much memory is used to describe the style information records, how much memory is used to describe the text and style information, and finally how many

rulers are defined. The number of rulers should always come back as one unless you're using some as-yet unwritten futuristic version of the TextEdit tool set that supports multiple rulers.

User Fields

The `TERuler` and `TEStyle` structures, in addition to their regular fields, also have a user data field. You can use this field to store any four-byte custom data just like you would use a `refCon` field for a window or control. Note, however, that by using the field, you are actually creating a new unique ruler or style structure. This means, for example, that you could have two `TEStyle` structures with the same font and color information but different user data fields and they would be considered unique by TextEdit.

More Information

Even though I have described a lot of what you can do with TextEdit here, there is still even more information! To detail it all would be rather silly since the *Apple IIGS Toolbox Reference: Volume 3* has everything in exacting detail. Some things that I don't touch on here are custom filter procedures and hook routines and looking at the TextEdit low-level data structures (which you should never have to do unless you're writing the be-all, end-all, high-speed, text-editing application).

Teach Files

One thing you might have to deal with when using the TextEdit tool set is the Teach file. A Teach file contains styled text information which can be placed into a TextEdit control. The description of a Teach file is documented in file type note FTN.50.5454, which is present on your *GS+* Disk. Of course, the most important information is the text itself. The entire text is simply stored in the data fork of the Teach file. All other information for a Teach file is contained in the file's resource fork. The style information (the `TEFormat` structure) is contained in the `rStyleBlock` resource with the resource ID of $00000001. Since the style information contains a ruler, and since the ruler contains margin settings, you have to correctly set the width of the TextEdit control so the text is displayed properly. A Teach file is saved with a window position information resource type $7001 with the resource ID of $00000001. The window position information assumes that you are using a TextEdit control with a grow box so that the TextEdit control takes up the entire content region of a window. Saved within the window position information resource is a window position (top and left coordinates for the window) as well as window size (width and height) information. One final resource which may or may not be present in a Teach file is a `rPrintRecord` resource with the resource ID of $00000001. You should take advantage of the `rPrintRecord` resource if it is present, but you should not count on it being there.

Is it Teach or is it EGOed?

Once again, I've provided a demo program on your *GS+* Disk. The demo program shows you how to take advantage of the TextEdit tool set. I built upon the demo program from the Font Manager (using Font, Size, and Style menus) to create a nice text editing application similar to the Teach program provided by Apple and to our own program EGOed. The TextEdit demo program is probably the most functional program I've ever written for the "Working With The Toolbox" series, however, it is still a demo. It is not meant to replace Teach or EGOed. There are a *lot* of things that a good text editor does that the TextEdit demo program ignores. For one thing, it can't print. I actually started to put the ability to print in the demo, but then I realized I hadn't covered the Print Manager in this series yet! So now I have to think about what tool set to cover for the next issue ....

Housekeeping

The TextEdit demo program is probably a bit different than most applications you've seen before. Sure, on the outside it appears to be a typical desktop application. However, on the inside there are a couple of very interesting tricks that are being done. The first thing you may notice different is that the main event loop is using `GetNextEvent` as well as `TaskMaster`. To use `GetNextEvent` and `TaskMaster` in the same event loop, you have to set the `tmNoGetNextEvent` bit in the taskFlags field so that `TaskMaster` uses the event already contained in the event record instead of calling `GetNextEvent` itself. This technique didn't work every time (ROM 03 machines especially didn't like the `tmNoGetNextEvent` bit) until System 6.0.1, so if you do this in your programs, you will have to require System 6.0.1. In addition to simply calling `GetNextEvent`, you also have to call `SystemTask` so that desk accessories have a chance to perform their run actions. And finally, if `GetNextEvent` returns a false value (telling you that your application should not handle the event) then you should *not* call `TaskMaster`. The only exception to this rule is when `GetNextEvent` returns a null event. (I had to find all this information out the hard way. Apple documented the `tmNoGetNextEvent` bit, but never bothered to document the supporting tasks you have to code to actually use the bit. The hard way was actually a combination of trial and error as well as using GSBug to trace through `TaskMaster`—what fun!)

Using `GetNextEvent` in your event loop makes your application much more flexible than using `TaskMaster` alone. In the TextEdit demo program, some menu item states are set whenever a mouse down event in the menu bar region or a key down event with the Command key modifier is noticed. After the menu item states are set, `TaskMaster` is called to handle the event. If the event entailed making a menu selection, the menu item states were properly set prior to the `TaskMaster` call, but after the initial low-level event. Maintaining certain variable menu item states (such as the Paste menu item state, which is dependant upon a text scrap being in the system clipboard, the Save menu item state which is tied to the TextEdit control's dirty bit, and items corresponding to the current text selection) can be very time consuming if they are performed *every* time through the main event loop. You would see a marked decrease in response time to events, particularly when typing. So, to speed things up, the variable menu item states are set only when an event might cause a menu item to be displayed or chosen. Menu item states that are constant for a particular window (such as the Close item—it's *always* available when a window is open) are still handled in the familiar `CheckFrontWindow` routine.

Also contained in the event loop is the `HandleDiskInsert` call. This call watches for disk inserts, and if the inserted disk is not initialized, it will prompt the user to let them format the disk or eject it. The `HandleDiskInsert` call can do more than simply watch for uninitialized disks, but normally that's all you'll ever really need it for. It's simply a good practice to put the `HandleDiskInsert` call in all of your main event loops.

Another nifty routine you might like is the one that maintains the proper cursor appearance. Every time through the event loop, the `CheckCursor` routine is called. It checks to see if the cursor is over a TextEdit control, and if it is, it sets the I-beam cursor. If not, it sets the arrow cursor. If the frontmost window is a desk accessory window, nothing is done to the cursor since desk accessories should be allowed to manage their own cursor appearance. Note that the cursor is not

changed if the correct state is already set. To do this, at application startup time, the addresses for the arrow and I-beam cursors are found. When it's time to change the cursor, the address of the current cursor is checked against the address of the desired cursor, and if they're the same, no changes are made.

The only special thing done in the main event loop to keep up with the TextEdit controls is (. . . drum roll please . . .) absolutely nothing! That is, of course, unless you count switching to an I-beam cursor when the cursor is positioned over the TextEdit control. Everything else is handled for you by `TaskMaster`. `TaskMaster` takes care of everything from typing characters, to scrolling, to choosing special menu items (such as Cut, Copy, Paste, and Clear).

The TextEdit Stuff

Even though the demo program doesn't do *everything* a real text editor should, it does enough that you can get the feel for how to use the TextEdit tool set. If you want, you can build upon the TextEdit demo foundation and create a nice text editor for yourself. Some things that should probably be added are:

• Printing (I'll most likely explain how to do this in the next "Working With The Toolbox" installment, so you shouldn't worry about this too much yet.)

• Add a Windows menu which keeps track of all the application windows

• Every time a new window is opened, it should get a new "tiled" position on the desktop

• Every time a new window is opened, a count should appear after the window title (The first window should be "Untitled" without any number. The second window should be "Untitled 2". The third window should be "Untitled 3" and so forth.)

• Documents other than Teach files could be opened and saved

• Support for the message center could be added to allow the Finder to automatically open and/or print documents (This includes looking in the message center when the application starts up as well as adding a `rBundle` resource to the application.)

I could have added these features to the demo program myself, but that was going a bit beyond simply using the TextEdit tool set. If you feel up to it, adding the preceding features would probably be a good exercise. I'm sure you can think of many other features you'd like a good text editor to have as well. Of course, if you get stuck, don't hesitate to send in your questions. I'll answer them as best I can.

That's All Folks!

Using the TextEdit tool set isn't very difficult. The source code for the TextEdit demo program covers a lot of TextEdit related material. If you find you need to do more than the TextEdit demo program does, you should read up on the Toolbox references and experiment. If you had trouble following this article, or the Toolbox references, let me know and I'll attempt to clarify.

The next "Working With the Toolbox" installment will be a nice discussion of the Print Manager. After that, I think I will have run out of "major" tool sets to cover. I never thought I'd see the day when I could type that! However, that doesn't mean that I've covered every tool set. There are still quite a few left to take a look at. However, with the major tool sets out of the way, I don't really have a direction to take the remainder of these articles. So, to help me out, you can write in and tell me that you'd like to see a discussion of the <insert tool set name here> tool set and I'll work on it. I look forward to seeing your requests! *GS+*

GS+ Classifieds

Readers can place an ad in the *GS+* Classifieds for only $5. This price buys you 25 words in one issue of *GS+* Magazine. Additional words are just 25 cents each. The *GS+* Classifieds are a great way to contact thousands of other IIGS owners.

The deadline for inclusion of a classified ad in the next issue (Volume 6, Number 3) of *GS+* Magazine is December 12, 1994. Simply send your ad along with your name, address, phone number, number of issues to run, and payment (made payable to "EGO Systems") to *GS+* Classifieds, P. O. Box 15366, Chattanooga, TN 37415-0366; or call us at (615) 332–2087, Monday through Friday between 9 a.m. and 5 p.m. Eastern Time, to place an ad with your MasterCard or VISA. You can also FAX us your classified ad by calling (615) 332-2634.

For Sale: Apple IIe Cards
AE RAMworks III 1 meg memory card: $100
AE Color Link Video card: $50
Clock card: $15
IIe Mouse card: $75

Contact: David McClay
614-259-4606

Wanted to Buy
Will pay cash or US check for: "HOMETOWN, U.S.A." Miniature railroad building program.

Contact: Guideon SPIRYTUS
1, rue de Marnes
92410 Ville d'Avray
France
Tel/Fax 33 1 4750 8139

Music, MIDI & Your Apple IIGS
Apple IIGS MIDI Value Pack: Interface, Notation Software, Sequencing Software, Cables, Book - $275
Or, Call for MIDI Buyer's Guide: 240 pages - $5

Contact:
Sound Management Electronic Music Products
P. O. Box 211
Lexington, MA 02173
(800) 548-4907
FAX: 617-860-7325

The Big *GS+* Moving Sale!
Apple Memory Board (0K) - $25
256K SIMMS, 150ns (Perfect for OctoRAM & older Macs) - $5 each!
FutureSound Stereo Card or GoldStar 1200 Baud Modem - $20 each
Star Saga One adventure game! Never played! - $20
Photonix II Disk Copier, TeleWorks Plus, List Plus, Writer's Choice Elite, DrawPlus - $10 each
Lots more stuff available! Call for complete list!
Free shipping on orders for more than $10!

Contact:
GS+ Magazine
P. O. Box 15366
Chattanooga, TN 37415-0366
(615) 332-2087

GS+ Classified Ad Order Form

Ad copy: _____
_____
_____
_____
_____
_____
_____

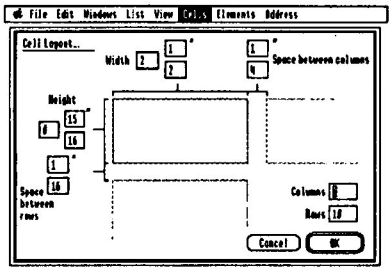
Number of issues to run: _____ Number of words: _____ Total enclosed: $_____

Name: _____ Phone: (_____)_____-_____

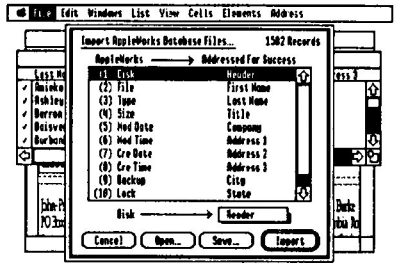
Address: _____

City: _____ State: _____ Zip: _____

Addressed For Success™

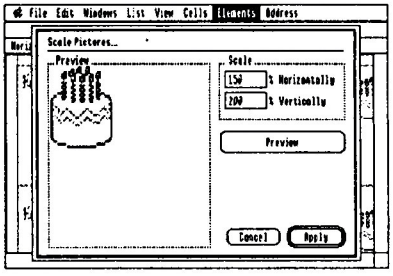


Includes Dozens of Pre-made Label Templates and Lets You Easily Design Your Own Label Templates



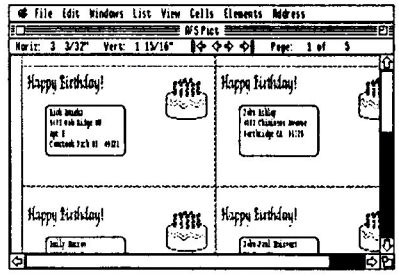
Imports ASCII Text Files and AppleWorks Classic v3.0 and v4.0 Databases

Works With All of Your IIGS Fonts
Performs Bulk Sorts For Reduced Postage Costs
Prints on Any IIGS Compatible Printer
Prints Postal Bar Codes
Lets You Select "By Example" the Records You Want to Print
Lets You Easily Specify a Return Address For All Your Labels



Lets You Print up to Three Different Graphics on Each Label

Allows You to Individually Scale Each Graphic For the Best Printing Results



You can do all of these things, and much more, with Addressed For Success™! Addressed For Success is the *only* desktop-based, label design and printing application available for the IIGS. It's powerful, versatile, and extremely easy to use. Best of all, Addressed For Success is on sale for only $30! So, whether you need to print a dozen labels or 1,200, Addressed For Success is exactly what you need to make yourself, and your labels, look great! But hurry, this sale ends on 12/22/94!

To order Addressed For Success v1.1, send a check or money order for $30 in U.S. funds to: EGO Systems, P. O. Box 15366, Chattanooga, TN 37415-0366. If you prefer to order by credit card, you can use your Visa or MasterCard by calling us *toll-free* at 1-800-662-3634. Outside of North America, please call 1-615-332-2087. Or, FAX us at 1-615-332-2634. TN residents add 7.75% sales tax. System Requirements: An Apple IIGS with System 6.0 or later, 2MB of RAM and at least one 3.5-inch disk drive. A hard drive and 4MB of RAM are *strongly* recommended! Addressed For Success was written by Michael Lutynski. Addressed For Success is a trademark of EGO Systems. AppleWorks is a trademark of Claris.

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

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

Forwarding & Return Postage Guaranteed