

\$3.00
(magazine only)

\$8.00
(magazine + disk)

**March
April
1990**

**Volume 1
Number 4**

The *First* Apple IIGs Magazine + Disk Publication!

Happy Easter!

"If
You
Think
I'm
Gonna
Dress Up
Like A Rabbit For
This Issue's Cover...

You're Nuts!!!"

In This Issue:

- Beginner's Guide to the Finder - All About Icons
- How To Write a CDEV
- Random IIGS Programming Notes - An EGOed Update
- Architecture on Your IIGS

Reviews:

- InnerDrive Vs. Vulcan
- Salvation
- ORCA/Disassembler
- Computer Eyes
- Jam Session
- Ancient Land of Ys
- Tunnels of Armageddon

**Plus Much, Much
More!**



WRITER'S BLOCK

First of all, let me apologize for this issue being so late. We try hard to avoid these delays, but it's just Noreen and me here, and this is at least a ten-person job! We are making changes to streamline our operation, so that, hopefully, this sort of thing won't happen again after the next issue.

Yes, that's right, the next issue will be late, as well. But it won't be quite as late, and we have a good reason... AppleFest. In the next issue we will be bringing you a first-hand report of this spring's AppleFest in Somerset, New Jersey (see Rumors, Wishes & Blatant Lies on page 19 for more information).

Before I get into the heavy stuff, let me tell you that we have switched BBS software. We are now running ProLine BBS software (ProLine is published by the Morgan Davis Group). The phone number is the same, but the name of the board is now *pro-gsplus*. We are hooked into the ProLine network through pro-europa in Houston, TX (thank you Paul!). So, if you are on UUCP, BitNet or any of those other Nets (can you tell I'm still new to all this network stuff?), you can reach me by sending some EMail to: diz@pro-gsplus.cts.com.

Well, the reaction to our decision to take ads from mail-order firms has been very positive. In fact, quite a few of you even said that you trust us enough for us to begin taking ads from the actual product manufacturers. OK, you talked me into it.

If you represent a company with a IIGS specific product to sell, give us a call and we will try to set you up with an ad. At this point in time, our ad rate is \$50/page.

But, don't get all excited yet. When I say that, "we will try to set you up with an ad," I mean that all ads will be put through the following meat-grinder:

1) If the product is one we are not familiar with, we will request a review copy. **NO BETA COPIES WILL BE ACCEPTED!** Only finished, available versions of products may be advertised in *GS+*.

2) We will only print an ad for a product if we find that, within reason, the product performs as advertised. As examples, we would not have accepted an ad for the original version of HyperStudio; we might have accepted an ad for the original AppleWorks *GS*.

3) If the product does not perform as advertised, the ad will be refused and we will let our readers know about it.

4) We will only take ads from mail-order firms that we are familiar with. If we are not familiar with a particular mail-order firm, we will contact the Better Business Bureau for information on the history of the company. If everything checks out OK, we will run the ad on a trial basis.

5) If we are given a sufficient number of complaints about a mail-order firm, we

will pull the advertising and let our readers know about the complaints.

If this all seems just a tad, harsh... it is. Why? Well, as you may remember, I never wanted to have to print ads in the first place. But, the realities of a rather soft IIGS market have forced my hand. But, by taking ads, we open ourselves up to the same criticisms that certain other magazines (that shall remain nameless) have been subjected to. I want to make sure that when you read a review in *GS+*, you know that it is an unbiased, honest review. And, when you see an ad in *GS+*, I want you to know that the products or services offered in that ad are among the best the IIGS market has to offer.




Diz






"Look thoughtful, you idiot!"

CONTENTS

FEATURES

	Beginner's Guide to the Finder - Part 3: All About Icons....	4
	How to Write a Control Panel Device (CDEV).....	7
	Brush with Greatness - Architecture.....	13
	Random IIGS Programming Notes - EGOed Update.....	16


REVIEWS

	InnerDrive vs. Vulcan.....	25
	Salvation.....	27
	ORCA/Disassembler.....	29
	Computer Eyes	31
	Jam Session.....	32
	Ancient Land of Ys.....	34
	Tunnels of Armageddon.....	35

CLASSICS

Where in the World is Carmen Sandiego.....	37
--	----

DEPARTMENTS

	Writer's Block.....	inside front cover
	Letters.....	2
	Programmer's Queue & A.....	3
	Back Issue Information.....	6
	Rumors, Wishes & Blatant Lies.....	19
	How to Use the GS+ Disk.....	20
	Icons.....	21
	Classifieds.....	24
	Product Updates.....	33
	Contest 3.....	36
	The MoleHill.....	39
	Writer's Guide.....	40

Steven Disbrow • Publisher, Editor

Noreen Ribaric • Associate Editor,
Layout

Jeffrey H. Walker • Contributing Editor

FingerPrint GSi was used to freeze the screen so that the screen photographs could be taken. This issue's cover was created by Michael Quinn using DeluxePaint II and Computer Eyes GS.

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

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

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

GS+ is published bimonthly by:
EGO Systems
4126 Mountain Creek Road #20
Chattanooga, TN 37415
(615) 870-4960

If you have a submission for GS+,
send it to:

GS+ Submissions
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415

Subscription rates for one year
(six issues) are:

Magazine only - \$15.00
Magazine + Disk - \$36.00.

Canadian and Mexican orders add \$1.
Other foreign orders add \$5 for surface delivery
or \$25 for air mail.

Quantity discounts available for schools
and Users' Groups.

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

GS+ Subscription Services
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415
or call (615) 870-4960

GS+ is produced on an Apple IIGS using
AppleWorks GS v1.1, GraphicWriter III,
and an Apple LaserWriter IINT.

No part of this magazine or its companion
diskette may be reproduced without the
written permission of EGO Systems.

pro-gsplus BBS (615) 875-4607
2400 Baud/8 data bits/no parity/1 stop bit

LETTERS

Dear Steve,

I enjoyed the GS+ magazine [November/December 1989] and I especially enjoyed the NDA Program [EGOed] and source code. While I haven't looked at it really carefully, your use of lots of good comments make the program very useful. Too bad you didn't use the "good" Pascal compiler (ORCA/Pascal), which I use.

While the magazine itself was OK and I am sure very useful and interesting for many IIGS users, for myself the information it contained was generally not that useful, at least in this one issue. I am familiar with the Finder and do not find myself having very much time to play games. I am also not much into drawing and other artistic sort of things. My use of the IIGS is much more productivity and programming based.

However, based mainly upon your NDA and the accompanying source code I am still interested in subscribing. However, I was wondering what your cancellation and refund policy would be if after 3 or 4 more issues I find that I am not getting what I thought I would be getting from the magazine/disk. Thanks.

Steven Peterson
Edina, MN

There are several reasons that TML Pascal II was used to create EGOed. The first reason is that I was doing a review of TML Pascal and, well, the best way to review a compiler is to use it to write a program. Another reason is that, at the time EGOed was written, ORCA/Pascal did not have the proper interface files for the Text Edit Tool Set or any of the other System 5.0 Tools that I needed. Also, ORCA/Pascal is just so SLOW in it's compile-and-link cycle that I probably would not have been able to get the program done in time even if the interfaces were available.

Our refund/cancellation policy is that we will refund the cost of any issues not yet mailed. I hope that the content of the last

issue and this one, both of which are very heavily productivity and programming oriented, will keep you from cancelling your subscription. Please understand that while we intend to be a magazine for "power-users," every "power-user" starts out as a beginner. It's our job to help those beginners become "power-users." Also, you can never tell what you might learn from reading something written for beginners. I learn something new every time I write one of those beginners articles, and there have been quite a few "power-users" that have called me up to say that they learned something new while reading them.

Gentlemen,

...Would it be possible to receive your first two issues (or however many there are)? I am especially interested in obtaining the second issue, but would like to have a copy of all, if possible.

Mrs. Marion B. Ingram
Cantonment, FL.

Back issues are available for all issues of GS+! For more information, see page 6.

Steve,

I would like you (and your staff) to know that I have enjoyed the first two issues of GS+ Magazine very much. Of particular interest was the System 5.0 compatibility chart and Beginner's Guide to the Finder. One question: Do you think you could begin a column that would explore programming on the GS from the "Ground Up?" By this I mean for someone like myself that has had NO previous programming experience whatsoever? Perhaps you could start the column out by suggesting books, articles, etc. that readers could obtain and then discuss some of the aspects of the material in the next issue. I think the 2 month publication format of GS+ (and disk) would lend itself quite well to such a column in that it would give readers enough time to read the material, experiment with it, discuss it in GS+ and then move on to something else. I would

really like to be able to program this machine but feel somewhat frustrated in that I don't know where to start!

Ron Hochevar
Bloomington, CA

Well, hopefully, last issue's beginner's programming tutorial, "Rotator," helped somewhat. If not, turn to page 3 and take a look at the first installment of our new department, Programmer's Queue & A. Hey, is that quick service or what?

If you have a comment, question, or criticism, we want to hear it!
Write to:

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

Next Issue...

FEATURE ARTICLE:

- What REALLY Happened At AppleFest!

FOR BEGINNERS:

- How to Set up a Startup Disk

PROGRAMS:

- Fractals GS v2.0
- Mouse Position

REVIEWS:

- CMS 45MB Removable Hard Drive
- GraphicWriter III
- Hunt for Red October
- Skate or Die

IIGS Classics:

- The Bard's Tale

**PLUS MUCH,
MUCH MORE!**

PROGRAMMER'S QUEUE & A

Queue:

What books, articles, newsletters, etc. would you recommend for those us who aspire to learn to program on the IIGS?

Ron Hochevar
Bloomington, CA.

A:

My, my, my... this looks like a simple question doesn't it? Well, it is, but the answer is a bit tricky. First, pick a language that you want to work in (Assembly, C, Pascal, etc.) and learn that language! Learning to program the IIGS is tough enough without having to learn a language at the same time. The best way to learn your first or a new computer language, is to learn it on the job or take a class in it. If neither of these methods is practical, home study is a close second. The most important thing to remember when you are learning to program is to concentrate on the concepts and not to get bogged down in the particulars (syntax) of the language that you are using. Once you learn the concepts of programming, no matter which language or computer you learn them with, you can use that knowledge to program any computer in any language from that day on.

So then, knowing that it's the concepts that we should be focusing on, what books are good for the beginning programmer? If you have decided to work in Pascal (and even if you will be working in another language) I highly recommend the book *Oh! Pascal!* by Doug Cooper and Michael Clancy from W.W. Norton & Co. This wonderful book not only teaches you Pascal, it goes out of its way to teach basic programming concepts. It's also very funny and features as one of its example programs, the classic, "Hunt The Wumpus" (which, I believe, is being made into a movie starring Sean Connery). If it's C that you want to learn, well, I don't know of many good C books that teach programming concepts. Most C books assume that you have Computer Science Degree. The ORCA/C manual

recommends the book *C Primer Plus* from Howard W. Sams & Co., Inc. as, "An excellent primer for beginners," so you may want to look at a copy of that. One book that you must have is *The C Programming Language* by Brian W. Kernighan and Dennis M. Ritchie. This book, published by Prentice-Hall, is lovingly referred to as, "The C Bible." (If anyone knows of a good concepts book for the C language, please let me know about it.)

For those of you disturbed enough to be starting out in Assembly Language, congratulations. While not the easiest thing in the world to master, an understanding (even a vague one) of the concepts of Assembly Language will make you a much better programmer. However, finding a good concepts book on Assembly Language is about like trying to find a good concepts book on C. But, the one book that everybody seems to love is, *Programming the 65816* by David Eyes and Ron Lichty, published by Brady Prentice Hall Press.

Now that I've beat that to death, which books do I recommend for actually programming the IIGS? Well, at the very least, you need the *Apple IIGS Toolbox Reference*, volumes 1 and 2. *The Programmer's Introduction to the Apple IIGS* would be good for someone unfamiliar with the concepts of event-driven programming, pull-down menus, windows and the like. These are concepts that you must learn, in addition to the basic concepts of programming, if you wish to write IIGS desktop programs. Once you learn them though, you can easily learn to program any machine that uses these concepts (Macintosh, Atari ST, Amiga, OS/2 machines, etc.). Another book to get would be *Exploring Apple GS/OS and ProDOS 8* by Gary B. Little. All of these books are published by Addison-Wesley.

After you have buried yourself under this mountain of books, the last resource you should call on is the one at the top of your neck. Write programs. Lots of them. While it's very easy to start working on a program, the trick lies in finishing it. Remember that it's normal to get frustrated when trying to

finish a program. The secret is to take your time and not give up!

Queue:

Where can I find good examples of ORCA/Pascal code? Do you plan to publish anything written in ORCA/Pascal? I like your EGOed text editor very much, but I would like to convert it to ORCA/Pascal. What would be the corresponding ORCA/Pascal routines for TML Pascal II's RefDescriptor(), Ref(), ResultBuf255Hndl, StuffHex(), etc.?

Jean-Patrick Hine
Plaisance Du Touch, France.

A:

Well, since everyone seems to be asking for ORCA/Pascal programs, you will find some good examples of ORCA/Pascal code in the next couple of issues of *GS+*. On the whole though, TML Pascal II and ORCA/Pascal are quite a bit alike, and you should be able to translate from one to the other without much trouble. (Note that this was not the case with the original TML Pascal!) However, for the specific items that you ask about, I can give some fairly quick answers.

Ref() and RefDescriptor() are examples of how TML Pascal II handles *type casting*. In EGOed, these type casts are mainly used to have the compiler treat Integer variables as Reference values for Toolbox calls that use Resources. Type casting works exactly the same way in ORCA/Pascal. The catch is, for these type casts to work in ORCA/Pascal, you will need to have the System 5.0 Toolbox interface files for ORCA/Pascal. If your version of ORCA/Pascal does not have these interface files, you can get them from Byte Works for a small update fee. For more info, look in the index of the ORCA/Pascal manual under "type casting" and "interface files." For info on the new interface files, call Byte Works at (505) 898-8183.

(continued on page 24)

BEGINNER'S GUIDE TO THE FINDER

Part 3: All About Icons

By Steven W. Disbrow

Over the last two months, quite a few of our readers have called or written to tell me how much they either liked or hated these articles. In talking with the folks that like the articles, one subject kept coming up: Icons. "How do they work?", "Why don't mine work?", "How do I use an Icon Editor?" So, in this final installment of the Beginner's Guide To The Finder, we will be answering those very questions, and a few others. To begin, the most popular question was...

"What are these 'Icon' things anyway?"

Icons are small pictures representing the different devices (disk drives, CD ROMs, printers, etc.) that you have attached to your IIGS. Icons are also used to represent the files stored on those devices (to keep things simple, I will call all of the things that Icons can represent, *items*).

"Where do these Icons 'live'?"

Icons are kept in files which are called, appropriately enough, "Icon files," or just "Icons." An Icon file can contain any number of Icons, and you can have as many

Icon files as you can fit on a disk. However, for the Finder to recognize them, Icon files must be in a folder called **Icons**, and, the **Icons** folder must not be inside another folder.

"How do Icons work?"

When you first put a disk in the disk drive, the Finder looks in the **Icons** folder of the disk to see if there are any Icons. If there are, the Finder then uses these Icons to represent the files stored on the disk. Each of the Icons stored within an Icons file contains within itself, several fields that tell the Finder exactly which device, file, or files this Icon should be used to represent. All of these fields must match up with the corresponding attributes of an item before a particular Icon can be displayed for an item.

"What are these fields?"

In no particular order, the main fields that make up the innards of an Icon are:

File Name - The File Name field is used to specify the name of the item that this Icon will represent. The Finder will display an Icon for an item only if the name of the item and the name specified in this field match.

In order to get the Finder to display an Icon for more than one item, this field should contain the *wildcard* character '*'. For example, if the File Name field of a particular Icon were "Kit*," any item whose name started with "Kit" (i.e. "Kitty," "Kitten," etc.) would be treated as a match. If the File Name field contains only the wildcard character, every item, regardless of its name, would be considered a match.

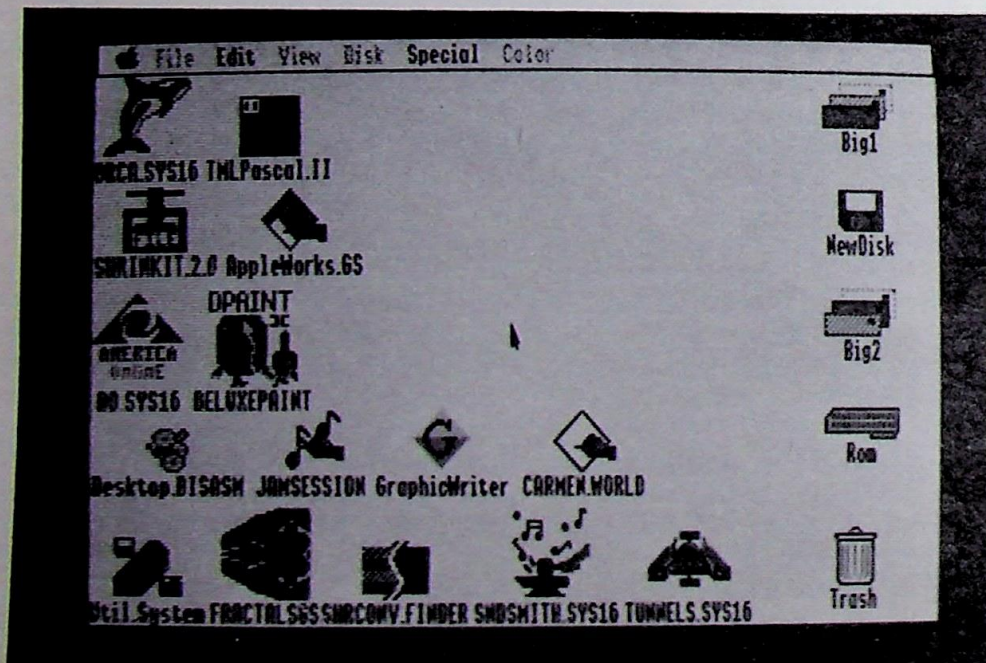
File Type - This is a number (usually shown in hexadecimal) that is used to specify the File Type of the items that this Icon can represent. If this field is zero, it will match with every item, regardless of its file type. In other words, zero is a file type wildcard.

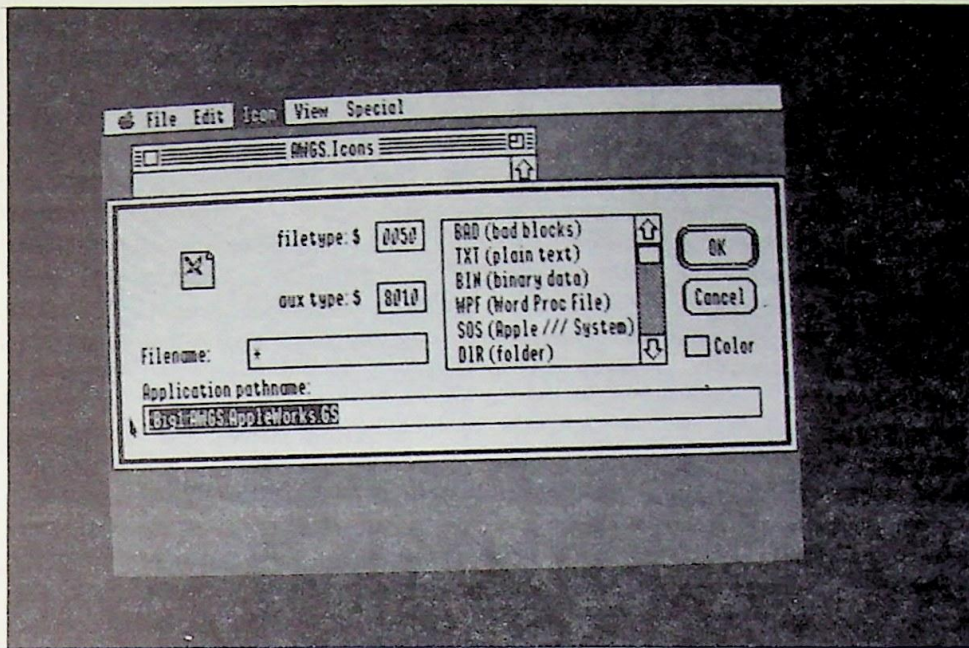
Auxiliary File Type - This is a number (usually shown in hexadecimal) that is used to specify the Auxiliary File Type of the items that this Icon can represent. If this field is zero, it will match with every item, regardless of its auxiliary file type. In other words, zero is an auxiliary file type wildcard. (See the "File Types and Auxiliary File Types" sidebar on page 5.)

Application Pathname - This field tells the Finder the *pathname* (see the "What's a Pathname?" sidebar on page 6) of the application it needs to run when the user double-clicks on an item represented by this Icon.

"What happens if an item matches more than one Icon?"

Since the Finder searches disks for Icons in the order that the disks appear on the Desktop, the Finder only displays the first Icon that matches an item. Subsequently matching Icons are ignored. So, for my Desktop, as shown at left, the Finder will first look for Icons on **Big1**, then on **NewDisk**, then **Big2** and finally **Rom**. Since **Big1** is my startup drive, the Icons from **Big1** have precedence over all other Icons. The Icons on **NewDisk** will have precedence over those on **Big2**, and **Rom**..., and so on, and so on. If you place another disk in a drive, any Icons on that disk will be subsidiary to the Icons on disks that have already appeared on the Desktop.





The Finder establishes this Icon precedence each time you run the Finder (i.e. when you first start up the computer or when you return to the Finder from another program.) So, if the order of your disks has changed, the precedence of the Icons on those disks will change the next time you run the Finder. Due to this fact, it is a good idea to keep your favorite Icons on your startup disk.

"At my Users' Group meeting, I saw someone double-click on an AppleWorks GS word processor document. When he did this, AppleWorks GS started up and the document was automatically loaded so that he could begin working with it. How does this work? Also, why does the Finder sometimes tell me that 'An application can not be found for this document,' when I double-click on an item?"

Let's answer this by example. On my system, AppleWorks GS is kept in a folder named AWGS on my hard disk, which is named Big1. So, on my system, the complete pathname for AppleWorks GS, and the Application Pathname that is stored in the Icon used to represent my AppleWorks GS documents is: :Big1:AWGS:AppleWorks.GS. So, when I double-click on an AppleWorks GS

document, here (basically) is what the Finder does:

First, the Finder looks at the Application Pathname field inside of the Icon for that item. If this field is empty, the Finder reports back that, "An application can not be found for this document."

In this example, the Application Pathname is not empty, so the next thing the Finder needs to do is to verify that the file :Big1:AWGS:AppleWorks.GS actually exists.

If this file does not exist, the Finder reports back that, "The application AppleWorks.GS can't be found for this document." Along with this message, you are given two buttons to choose from, Cancel and Try Again. The Cancel button is used to cancel the operation and the Try Again button gives you an opportunity to insert the disk containing the application.

If the file :Big1:AWGS:AppleWorks.GS does exist, the Finder starts up that application and passes it a message telling it the pathname of the document that the user double-clicked on. AppleWorks GS then uses this message to load the document so that you may begin working with it immediately.

"Well, that's all very wonderful for the Finder and AppleWorks GS, but why won't my Icons work like that?"

The number one reason that the "double-click on the Icon, the Application runs, and the document is loaded" feature does not work is that very few applications support this feature. For applications that do support this, the main cause of problems is that the application does not have the same pathname as the Application Pathname that is defined in the Icon. For example, if you keep your copy of AppleWorks GS on a disk named GSWorks, its pathname is :GSWorks:AppleWorks.GS. If this is not the Application Pathname defined in each of the Icons for AppleWorks GS documents, the Finder will not know where to look to find AppleWorks GS.

"So, if this pathname stuff is all screwy, how do I fix it?"

You use an Icon Editor.

"Care to elaborate on that?"

Basically, an Icon Editor is a IIGS application that allows you to examine and

File Types and Auxiliary File Types

File Types are numbers that GS/OS uses to identify the general nature of the contents of a file. Text files, for example have a File Type number of 4. Since few of us can even remember our zip code, Apple has defined a set of three character strings that are often used to refer to a File Type. To continue this example, the three character string for text files is "TXT."

Auxiliary File Types are numbers that can be used for just about any purpose that a programmer wants it to be used for. However, it is most often used in conjunction with the File Type to more exactly identify the contents of a file.

For a more complete list of File Type numbers, their three character strings and a utility that will let you change File Types and Auxiliary File Types, see "NoDOS" in the September/October 1989 issue of GS+.

"What's A Pathname?"

Every file has a name. This is called a *filename*. Now, suppose you have a LOT of files. These files may or may not have anything in common, so it would be nice to be able to store them on a disk in a logical fashion, grouping related files together. To accomplish this, GS/OS lets you store related files in *folders* (also called *directories*). These folders may contain files or they may contain other folders, allowing you to further organize your files. A *pathname* then, is a name made up of the name of the disk the file is on, the names of each of the folders that you must open to get to a particular file, and the name of the file itself. Each of these names is separated by a colon (:). Example: if you have a diskette named *Data*, and on that disk is a folder called *Sales*, and within that folder is a file named *January.90*. The complete pathname of the file *January.90* would be: `:Data:Sales:January.90`.

change both the appearance and attributes of the Icons contained in Icon files. Two of the more popular Icon Editors available for the IIGS are IconEd and DIcEd, both of these are ShareWare and can be downloaded from any of the larger information services (America Online, CompuServe, GENie), or our own pro-gsplus BBS. You may even be able to find a copy at your next Users' Group meeting, contact your club's librarian to see if either of these programs is available.

Once you have an Icon Editor, you can use it to change the fields discussed above so that your Icons will work correctly with your particular IIGS setup. If you are feeling artistic, you can also use an Icon Editor to change the appearance of the individual Icons. It does not take a great deal of artistic talent to do (as I prove each issue), and it lends a whole new meaning to the term "personal computing."

As for how to actually use a particular Icon Editor... the important thing is to get a basic

understanding of how Icons work and refer to the documentation that came with your particular Icon Editor. Then, sit down and play with the Icon Editor. In a very short time, you should be able to modify any Icons that you might have.

Well, that about does it for our Beginner's Guide to The Finder. If you have made it through all three articles, you should have a fairly good idea of how the Finder works, what it can do, and how to make it do it. If you are still completely lost, here are a couple things you can do:

- Read your manuals. That's what they're for!
- Ask about it at your next Users' Group meeting. That's what Users' Groups are for!
- Write or call us with your questions. That's what we're for!

The best thing you could do though, is to sit down at your IIGS and explore on your own. That's what it's for!

BACK ISSUE INFORMATION

September/October 1989 (Volume 1, Number 1)

\$4 magazine only
\$5 disk only
\$6 magazine + disk

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

PLUS: Icons, Rumors, and more!

November/December 1989 (Volume 1, Number 2)

\$4 magazine only
\$8 disk only
\$9 magazine + disk

- **Beginner's Guide to the Finder** - Part 1: The Basics
- **EGOed** - An NDA text editor (TML Pascal II source code on disk)
- **Update Info** - AWGS v1.1, HyperStudio v2.0, System 5.0.2
- **Brush with Greatness** - Tips on drawing faces (pictures on disk)
- **Reviews of TML Pascal II, Source Code Library II, Cutting Edge Keyboard, Battle Chess, Dark Castle, Dungeon Master, Neuromancer, Laser Force**

January/February 1990 (Volume 1, Number 3)

\$4 magazine only
\$6 disk only
\$9 magazine + disk

- **Beginner's Guide to the Finder** - Part 2: Mousing Around
- **Rotator** - A beginner's desktop programming tutorial and program w/source code written in ORCA/C
- **Winning Arkanoid II Levels**
- **Brush with Greatness** - Space graphics (pictures on disk)
- **Reviews of HyperStudio v2.0, Graphic Disk Labeler, Programmer's Online Companion, Xenocide, Keep the Thief, Life & Death, The Three Stooges**

ALL ABOUT CONTROL PANEL DEVICES

What you need to know to use and write CDEVs

By Steven W. Disbrow

What is a CDEV?

With the release of System Software v5.0, Apple Computer gave IIGS owners access to a whole host of new capabilities. One of the more significant of these enhancements was the introduction of the New Desk Accessory (NDA) version of the Control Panel. As supplied on the System Software v5.0 System Disk, the Control Panel NDA pretty much duplicates the capabilities of the older, text-based, Classic Desk Accessory (CDA) version of the Control Panel. That is to say, both Control Panels give you access to utilities that allow you to control various aspects of your IIGS. The difference is that the Control Panel NDA is designed so that its capabilities can be enhanced by the use of Control Panel Devices (CDEVs). In this article, we will be referring to a new CDEV that, when added to the Control Panel NDA, will give us control over the color of the IIGS desktop. The CDEV itself and the source code that creates it, are on your GS+ disk. For more information on the locations of these files, refer to "How to Use the GS+ Disk" on page 20. From this point on, when I refer to "the Control Panel," I'm speaking about the Control Panel NDA.

How To Install And Use A CDEV

To add a CDEV to the Control Panel, you simply copy the CDEV file into the **System:CDEVs:** directory of your startup disk and restart the computer. To use the CDEVs installed in your system, select the Control Panel from the Apple Menu at the far left of the menu bar. When the Control Panel finally appears (it can take a while, depending on the number of CDEVs you have installed), use the scroll bar to find the icon of the CDEV you wish to use and click the mouse on that icon. The controls for that CDEV will then appear in the right half of the Control Panel window. To actually use a particular CDEV, you may, of course, have to refer to the documentation for it. However, the Control Panel provides a

standardized help system that all CDEVs should follow. To see the help for the currently selected CDEV, click on the box labeled "Help" in the lower left hand corner of the Control Panel window. For simple CDEVs this should provide all the help you need. For a more complex CDEV, the Help feature may refer you to some other documentation. For specific instructions on installing the Desk Color CDEV, refer to "How to Use the GS+ Disk" on page 20.

If all you need to do is use CDEVs, that's really all there is to it! If, however, you want to know how to write your own, read on.

What Is A CDEV REALLY?

A CDEV is a GS/OS extended file with a file type of \$C7 (199 decimal). It should also be noted that CDEVs are made up entirely of Resources. The Data Fork in a CDEV file is always empty. This means that in order to write a CDEV, you must have a Resource Compiler, like Apple's REZ compiler or a very capable Resource Editor. Since there are no "very capable" Resource Editor's currently available (the TML Pascal II Resource Editor, for example, will not perform all of the tricks needed to create CDEVs), obtaining REZ is something of a necessity at this point. Fortunately, Apple has provided all of the extra tools needed to write CDEVs in a fairly inexpensive package called, Programming Tools & Interfaces for APW version 1.1. For information on how to get this and the other products mentioned here, see the product listing at the end of this article.

Structural Overview of a CDEV

As I mentioned above, CDEVs are made entirely of Resources. Those Resources can be anything that the CDEV might need to get its job done. Our Desk Color CDEV, for example, needs a specialized resource to hold the default Desktop pattern. However, there are three particular Resources that must always be present for a CDEV to work correctly. They are:

The CDEVFlags Resource - This Resource contains a series of bit flags and strings that tell the Control Panel exactly how this CDEV will operate and provide author and version information for the Control Panel's standardized help system.

The CDEVCode Resource - This is the "program" part of a CDEV. This Resource contains the code that, using an assembler or compiler, you create to handle the actual tasks that the CDEV will perform. This code must be in standard OMF v2 format, not in the newer ExpressLoad format. After you compile and link this code, you use the REZ language "read" command to include it in the CDEV as a resource.

The Icon Resource - This Resource contains the Icon that the Control Panel displays to represent the CDEV. These Icons can be created with either DIconEd or IconEd (by using the "Save As Source..." option of either of these programs), but they will require a bit of editing to get them in a format that can be used to create an Icon Resource.

```
type rCDEVFlags {
    unsigned word; /* Field 1. Flags for the CDEV. See text for details. */
    byte; /* Field 2. CDEV enabled flag. Zero means disabled. */
    byte; /* Field 3. CDEV version number. Programmer defined. */
    byte; /* Field 4. Minimum ROM required for this CDEV to run.*/
    byte; /* Field 5. Reserved. Must be set to zero. */
    rect; /* Field 6. Rectangle the CDEV will be drawn in. The */
        /* top and left coordinates must be 0. */
    pstring[15]; /* Field 7. Pascal string containing the CDEV's name. */
    pstring[32]; /* Field 8. Pascal string containing the authors name.*/
    pstring[8]; /* Field 9. Pascal string stating the CDEV's version. */
};
```

Figure 1.

Details, Details...

Now that we have the big picture, let's look closer and see exactly how each of these required Resources works.

In Figure 1 on page 7, we see the REZ language type declaration for a CDEVFlags Resource. Apple has not given these fields any real names yet, so I'll just refer to them as Field 1, Field 2, etc. Since Field 1 is the most complicated, let's look at the others first:

Field 2 - This field is used by the Control Panel to determine if the CDEV is active or not. If the value in this field is zero, the CDEV is not activated.

Field 3 - This is a programmer defined version number for the CDEV. It does not have to be the same as the version number defined in Field 9.

Field 4 - This field tells the minimum ROM version that this CDEV requires to run properly.

Field 5 - This field is reserved by Apple. Its value must be set to zero.

Field 6 - This is the rectangle that the CDEV's controls will be drawn in. The top left corner must be defined as (0,0). The Control Panel handles the necessary offsetting when the controls are actually drawn.

Field 7 - This is a Pascal style string that contains the name of the CDEV. This string is displayed under the CDEV's Icon in the Control Panel window and in the top of the CDEV's Help window.

Field 8 - This is a Pascal style string containing the name of the author of the CDEV. This string will be displayed in the CDEV's Help window. Note that there is no need for this string to include the word, "By" (as in, "Written By..."). The Control Panel automatically provides a "By" in the Help window.

Field 9 - This is a Pascal style string telling the version of the CDEV. It is displayed in

the upper right corner of the CDEV's Help window.

Field 1 is the field that actually contains the flags that give this resource its name. But, before we can understand these flags, we need to back up a bit and discuss how the Control Panel actually communicates with a CDEV.

Basically, the Control Panel acts as an NDA shell for each CDEV. That is to say, the Control Panel provides its CDEVs with a window to work in, and it handles almost all of the CDEV's interaction with the system and the user. When, for example, you click the mouse on one of the Desk Color CDEV's buttons, the Control Panel calls TaskMasterDA. TaskMasterDA then goes through its usual routine (highlight the button, track the mouse, etc.) to report back to the Control Panel which button, if any,

was actually selected. All of this happens and not a single line of code in the CDEV is ever executed! Which brings us back to the question of how the Control Panel communicates with a CDEV. This communication is carried out via messages which are passed from the Control Panel to the CDEV's CDEVCode Resource. These messages are made up of a message code and two long-integer values. At this point in time, Apple has defined eleven different message codes, but only ten of them are actually used. The two long-integer values, which have been named data1 and data2, have different meanings depending on the message code. So, to continue the Desk Color example, once the Control Panel determines that one of Desk Color's buttons has been "pressed," it calls Desk Color's CDEVCode Resource with a message that says, "Hey! One of your controls was just

Bits 15 to 11: Reserved for messages that have yet to be defined. These bits must be set to zero.

Bit 10: wantRun.

The Control Panel should call this CDEV with a RunCDEV message whenever the Control Panel receives a DARun action from the system. This can occur up to 60 times a second.

Bit 9: wantHit.

The Control Panel should call this CDEV with a HitCDEV message whenever one of the CDEV's controls is "hit."

Bit 8: wantRect.

The Control Panel should call this CDEV with a message of RectCDEV, before the CDEV is displayed.

Bit 7: wantAbout.

The Control Panel should call this CDEV with a message of AboutCDEV whenever the user selects the Help button at the bottom left of the Control Panel window.

Bit 6: wantCreate.

The Control Panel should call this CDEV with a message of CreateCDEV when the user selects the CDEV's icon from the

scrolling display at the left of the Control Panel window.

Bit 5: wantEvent.

The Control Panel should call this CDEV with a message of EventsCDEV before the Control Panel calls TaskMasterDA!

Bit 4: wantClose.

The Control Panel should call this CDEV with a message of CloseCDEV when the Control Panel is being closed.

Bit 3: wantInit.

The Control Panel should call this CDEV with a message of InitCDEV before the CDEV is displayed. The CDEV has already been passed the CreateCDEV message (if the wantCreate flag was set).

Bit 2: wantShutDown.

This flag has a name, but, Apple has not yet defined its use. It must be set to zero.

Bit 1: wantBoot.

The CDev.INIT startup file (not the Control Panel!) should call this CDEV with a message of BootCDEV when the system is being started up or restarted.

Bit 0: Reserved. This bit must be set to zero.

Figure 2.

'hit' by the user." In that case, data1 is a handle to the control, and data2 is the ID of the control.

Now that we know how the Control Panel communicates with CDEVs, what exactly do the flags in Field 1 of the CDEVFlags Resource do? All these flags do is tell the Control Panel which messages this particular CDEV has been programmed to handle. Each of these flags has a name that corresponds to the message it represents. If a bit has a value of 1, it means that the CDEV has been programmed to handle, and wants to be sent, that particular type of message. If the bit is set to zero, the CDEV does not want to be sent that message. These bits and the messages they flag are shown in Figure 2 on page 8.

The CDEVCode Resource is, as was mentioned earlier, a OMF v2 load file that has been converted into a Resource via the REZ language "read" command. In other words, it's just a program that's kept in a Resource. (Figure 3 shows how to declare this program in various IIGS languages.) And, since the Control Panel takes care of 90% of this program's interaction with the user, it can be a very small program. When the CDEVCode program is called by the Control Panel, it is passed an integer message code and two long-integers that contain additional information about the

message. The message code determines exactly what that additional information is. For some messages, the CDEV is required to return a long-integer result to the Control Panel.

From the earlier discussion of the CDEVFlags Resource, you might expect that the program that makes up a CDEVCode Resource will just be a big CASE (or SWITCH) statement. And, of course, you would be right. There are, at this point, a maximum of ten messages that a CDEVCode program will have to handle. "The Programmer's Aid Cut-Out" on pages 11 and 12 shows these messages along with information on the meanings of data1 and data2, when a particular message will be passed, and what a CDEV should do in response to each message.

After looking at the messages, you might be thinking, "These are some of the same things an NDA has to take care of." Exactly right. Remember, the Control Panel is an NDA itself, so it stands to reason that CDEVs will be very "NDA-like" in their operation. One major difference is that CDEVs seem to have it a bit harder when it comes to allocating memory and starting up interrupt handlers. The reason for this is that CDEVs are made entirely of Resources. When they are not needed in memory any longer, the Resource Manager removes them completely, CDEVCode and all.

The Icon Resource is simply...well...it's an Icon. As such, the easiest way to create a CDEV Icon is with an Icon editor, like IconEd or DcEd, that can convert the icon into an Assembly Language source code file. After the Icon editor creates the source file, you need to edit it a tiny bit to get it into a format that REZ can compile (and that can be easily read by humans). Rather than discuss that editing process here, I've included on the GS+ disk, a file containing the source code generated by DcEd (**DC.Icon.asm**). The edited code for the Icon can be found in the file **DeskColor.rez**. By comparing these two files, and by reading the comments in them, you should be able to discern the process involved in getting an icon incorporated into a CDEV.

Putting It All Together

With all these separate parts, you might think that a CDEV is a real pain to compile and link into a working program. Fortunately, the APW/ORCA environment allows us to pretty well automate what would otherwise be a fairly nasty process into a simple batch file (pardon my MS-DOS slang). The "Make File" shown in Figure 4 on page 10 contains all of the commands needed to compile and link the Desk Color CDEV.

Testing a CDEV

While there are a few liabilities that come from CDEVs being made up entirely of Resources, there is one advantage that becomes apparent during the development of a CDEV. Unlike just about every other type of IIGS system file, you do not have to reboot to install and test a CDEV. All you have to do is copy the CDEV into the ***:System:CDEVs:** folder, and then select the Control Panel from the Apple Menu. Note that this won't allow you to test the BootCDEV portions of a CDEV, but, it will allow you to test all other aspects of it.

The Desk Color CDEV

Now that we have all of the theory out of the way, let's take a look at a real example.

Assembly

The CDEVCode program is called in the same way a Toolbox routine is. Before the CDEVCode program is called, these parameters are placed on the stack in the following order:

```
Space for the result (4 bytes)
The message code (2 bytes)
data1 (4 bytes)
data2 (4 bytes)
The return address (3 bytes).
```

The CDEVCode program must return with an RTL instruction.

ORCA & APW C

```
pascal long CDEVProgram(message,data1,data2)
  int message;
  long data1,data2;
```

TML Pascal II & ORCA/Pascal

```
Function CDEVProgram(message : Integer; data1,data2 : LongInt) : LongInt;
```

Figure 3.


```

set echo 1
* Compile the ORCA/C source code.
compile +w +e DColor.cc keep=DColorCDev
* We don't need to keep the .root file, so, get rid of it.
delete DColorCDev.root
* Use linkIIGS to link the object file. The -x flag tells linkIIGS that
* it should not create an ExpressLoad file.
linkIIGS -o DColorCDev -lib 2/OrcaLib -lib 2/SysLib DColorCDev.a -x
* Now, compile the REZ file. The "rez=(-t $c7)" tells the REZ compiler
* that the resulting file should be given a file type of $c7.
compile +w +e DColor.rez keep=DeskColor.CDev rez=(-t $c7)

```

Figure 4.

While this section won't tell you how the Desk Color CDEV is written (that's what the source code is for), it will help you relate its operation to all of the theory we covered above.

The first time you boot up your IIGS with Desk Color installed, watch the bottom left corner of the "Welcome To The IIGS" screen carefully. Just about when the red thermometer is under the 'h,' you should see the Desk Color Icon appear at the bottom left of the screen. This means that the CDEV.INIT file has called Desk Color with a message of BootCDEV. When Desk Color gets this message, it reads in the default Desktop color from its Resource file and posts a message to the system that tells it to draw the Desktop in that color from now on.

When the "Welcome..." screen disappears, the Finder (or whichever program launcher you use) will appear. Desk Color initially sets the Desktop color to the periwinkle blue that you are used to seeing. To change the color, you now have to open the Control Panel and select the Desk Color Icon.

When you select the Desk Color Icon, the Control Panel calls Desk Color with a message of CreateCDEV. At this point, Desk Color creates all of the colored buttons that it uses to allow the user to select a new Desktop color.

After the buttons are created, the Control Panel calls Desk Color again, this time with a message of InitCDEV. None of the buttons have to be initialized, but the patterns that make up the possible Desktop colors have to be created. So, Desk Color uses this message as a signal to create those patterns.

When the patterns have been created and control is returned to the Control Panel, the buttons are actually drawn on the screen. At this point, a couple of things can happen. First of all, the user could pick another CDEV or close the Control Panel. Either of these actions seems to cause the Resource Manager to remove the Desk Color CDEV from memory. While this is not always a good thing, it does allow the Control Panel to access an almost infinite number of CDEVs.

If the user clicks on one of the colored buttons, the Control Panel calls Desk Color with a message of HitCDEV. When Desk Color gets the message, it does the following things:

- 1) Desk Color checks to see if it is safe to change the Desktop color at that moment. Some programs, like the Finder, use a custom Desktop drawing routine. Trying to change the Desktop while running inside one of these programs can lead to unpredictable results. In the Finder, all of the Icons disappear.
- 2) If it is not safe to change the Desktop, Desk Color makes a note of the new color and displays an alert window that lets the user know the situation. The Desktop will be changed to the new color when the next program is run or when the system is restarted. If it is safe to change the Desktop, that's what Desk Color does.
- 3) In either case, the new Desktop color is saved as a Resource in the Desk Color file so that it will be available when the system is restarted.
- 4) The last thing that can happen is that the user will pick the Help button in the lower

left corner of the Control Panel window. When this happens, the Control Panel calls Desk Color with a message of AboutCDEV and Desk Color then displays its help text.

The End

Whew! If all this seems like a lot to get a grip on, don't worry. Study the Desk Color source code carefully and you should be able to get the hang of it in no time. Once you get used to the fact that the Control Panel and Resource Manager do most of the work for you, cranking out CDEVs will probably become one of the easiest and most rewarding programming experiences you will have with the IIGS.

If you have any comments, questions, or corrections for this article, be sure to send them to us here at GS+.

Programming Tools & Interfaces for APW version 1.1 - \$50

Order Apple Part Number A0228LL/A
 Apple Programmers and Developers
 Association (\$20/Year Membership Fee)
 U.S.A. (800) 282-2732
 Canada (800) 637-0029
 International (408) 562-3910

ORCA/C v1.0 - \$150

The Byte Works, Inc.
 4700 Irving Blvd. NW, Suite 207
 Albuquerque, NM 87114
 Orders, Information and Tech Support:
 (505) 898-8183

Apple II File Type Note - FTN.C7.XXXX

This Technical Note (upon which this article was based) is the official Apple Computer, Inc. description of how to write CDEVs. Available from any of the more popular online services or our own pro-gsplus BBS.

TransWarp GS CDEV

Written by Dave Leffer, this Public Domain CDEV not only gives you full control over the speed of your TransWarp GS, it comes complete with ORCA/C source code. It, too, is available from any of the more popular online services or our own pro-gsplus BBS.

PROGRAMMER'S AID CUT-OUT

CDEV Message Codes

Message 1: MachineCDEV

Flag Required To Call: wantMachine

Meaning Of data1: Undefined

Meaning Of data2: Undefined

When Is This Call Made: Before CDEV is displayed.

What It Does: This message allows CDEVs that require a specific piece of hardware or software to check the system for the presence of that hardware or software. For example, a CDEV to control a Track Ball would respond to this message by checking to see if a Track Ball was actually attached to the system. As a result of this check, the CDEV returns a result that tells the Control Panel whether or not the CDEV should be displayed. If this return value is a zero, the Control Panel will not display the CDEV.

Message 2: BootCDEV

Flag Required To Call: wantBoot

Meaning Of data1: Undefined.

Meaning Of data2: Undefined.

When Is This Call Made: During IIGS Boot process.

What It Does: It is important to note that this call is not made by the Control Panel, it is made by the startup file, *:System:System.Setup:CDEV.INIT. While the CDEV is processing this message, CDEV.INIT draws the CDEV's Icon on the screen. This makes it easy for a user to tell which CDEVs are active. This message also allows the CDEV to perform any required startup tasks. As you might expect, very few Tools are available at this point in time! One very important Tool that is available, however, is the Resource Manager. Our Desk Color CDEV takes advantage of this by storing the default Desktop color in its Resource Fork and, during the boot process, loading that color into memory and then changing the Desktop to that color.

Message 3: ShutDownCDEV

Flag Required To Call: wantShutDown

Meaning Of data1: Not yet defined.

Meaning Of data2: Not yet defined.

When Is This Call Made: Not yet defined.

What It Does: Nothing. This Message is not yet defined.

Message 4: InitCDEV

Flag Required To Call: wantInit

Meaning Of data1: Pointer to Control Panel's window.

Meaning Of data2: Undefined.

When Is This Call Made: After CreateCDEV, but before the controls are drawn.

What It Does: This message tells the CDEV that its controls are about to be displayed and gives it a chance to set the controls to their initial values. The Time CDEV, for example, would read the system clock to find out what time it is, and then set its controls to that value.

Message 5: CloseCDEV

Flag Required To Call: wantClose

Meaning Of data1: Undefined.

Meaning Of data2: Undefined.

When Is This Call Made: When the Control Panel is being closed.

What It Does: This message allows a CDEV to perform any tasks it might need to carry out prior to being closed. This could include such things as disposing of memory, saving the values of controls, etc. Note that a CDEV does NOT have to dispose of the memory for its controls; the Control Panel disposes of this memory for the CDEV.

Message 6: EventsCDEV

Flag Required To Call: wantEvents

Meaning Of data1: Pointer to current (regular-style) Event Record.

Meaning Of data2: Undefined.

When Is This Call Made: Right before TaskMasterDA is called.

What It Does: Ordinarily, the Control Panel calls TaskMasterDA to handle all Events concerning the current CDEV. If, however, the wantEvents bit is set in the CDEVFlags resource, the CDEV is passed this message before the Control Panel calls TaskMasterDA. This allows the CDEV to handle the event on its own. Since data1 is a pointer to the Event Record, the CDEV can even change the Event Record if necessary. (This is a regular-style, and not

PROGRAMMER'S AID CUT-OUT

CDEV Message Codes

a TaskMaster-style, Event Record). After the CDEV finishes handling this message, the Event Record is passed on to TaskMasterDA, just as it normally would be.

Message 7: CreateCDEV

Flag Required To Call: wantCreate

Meaning Of data1: Pointer to Control Panel's window.

Meaning Of data2: Undefined.

When Is This Call Made: When user clicks on a CDEV's icon.

What It Does: When the CDEV gets this message, it knows that it is time to create its controls. The CDEV uses the pointer in data1 to create its controls in the Control Panel's window. The easiest way to create these controls is to define them as Resources and then, when the CreateCDEV message is received, use the NewControl2() call to create the controls.

Message 8 : AboutCDEV

Flag Required To Call: wantAbout

Meaning Of data1: Pointer to Control Panel's Help window.

Meaning Of data2: Undefined.

When Is This Call Made: When user clicks on Control Panel's Help button.

What It Does: This message tells the CDEV that the user has asked for help. The CDEV then draws its help text in the Control Panel's Help window. The Control Panel automatically draws the CDEV's Icon, version number, and author information. The Control Panel also creates and handles the "OK" button at the bottom of the Help window.

Message 9: RectCDEV

Flag Required To Call: wantRect

Meaning Of data1: Pointer to CDEV display rectangle (defined in CDEVFlags)

Meaning Of data2: Undefined.

When Is This Call Made: Before CDEV is displayed.

What It Does: When a CDEV receives this message, it can change the size of the rectangle that the CDEV will be displayed in. Consider, for example, a CDEV that would display the names of all disks online. Since disks can be inserted and ejected at any time and since not every system has the same number of drives attached, such a CDEV could use this message as a signal to find out exactly how many disks were online. It could then resize the display rectangle so that all of the names could be displayed without the list going past the bottom of the display rectangle.

Message 10: HitCDEV

Flag Required To Call: wantHit

Meaning Of data1: Handle to the control that was hit.

Meaning Of data2: ID of the control that was hit.

When Is This Call Made: When one of a CDEV's controls has been hit.

What It Does: This message notifies a CDEV that one of its controls has been hit. The CDEV may then use the information in data1 and data2 to take whatever action is appropriate for that control.

Message 11: RunCDEV

Flag Required To Call: wantRun

Meaning Of data1: Undefined.

Meaning Of data2: Undefined.

When Is This Call Made: Every time the Control Panel receives a DARun action.

What It Does: This message tells the CDEV that the Control Panel has received a DARun action from the system. This can happen up to 60 times a second. The CDEV can then perform any actions that it needs to. The Time CDEV for example, uses this to tick off the seconds in its display.

BRUSH WITH GREATNESS

By Michael J. Quinn

The theme for this issue's column is architecture. You should have two sets of pictures on your GS+ disk. One set shows the famous CitiCorp building and the other shows the even more famous home of Frank Lloyd Wright (one of the Wright brothers). As usual, there are several files in each picture set. Each file shows a step required to arrive at the completed picture.

The CitiCorp Pictures

Before we start, it should be noted that each of the pictures in this series is actually larger than the IIGS screen. In DeluxePaint II, you would use the up arrow and down arrow to scroll the picture. In Paintworks Gold or Paintworks Plus, you would use the Grabber tool (the one that looks like a hand) to move the picture around.

As you can see by the picture below, this is a drawing of the CitiCorp skyscraper. Computer graphics are very well suited for this kind of drawing. The main reason for this is that most buildings are primarily made up of straight lines and sharp right angles, with highly contrasting colors. All of these effects are easily accomplished with just about any paint program.

Colors

The first thing we need to do (as with any graphic on the IIGS) is to create the colors that we expect to use. For this picture, I have decided to use dark blues, bright yellows and several light shades of silver. The silver colors are created by starting with a bright white, then turning the brightness or "value" down until the color becomes a fairly light grey. As with most pictures, it is easiest to place the lighter colors toward one end of the color palette and the darker ones near the opposite end. For the best results, I usually choose two to four "base" colors to work with, then blend them together to get variations on those base colors. For this series of pictures, our base colors are: black, blue, yellow, and silver. This blending of colors becomes important when you want to

use a paint program's "smooth" tool to get rid of jagged lines, or when you need to use a gradient fill. In this series of pictures, just such a gradient fill has been used to create the sky in the background.

Basic Shapes

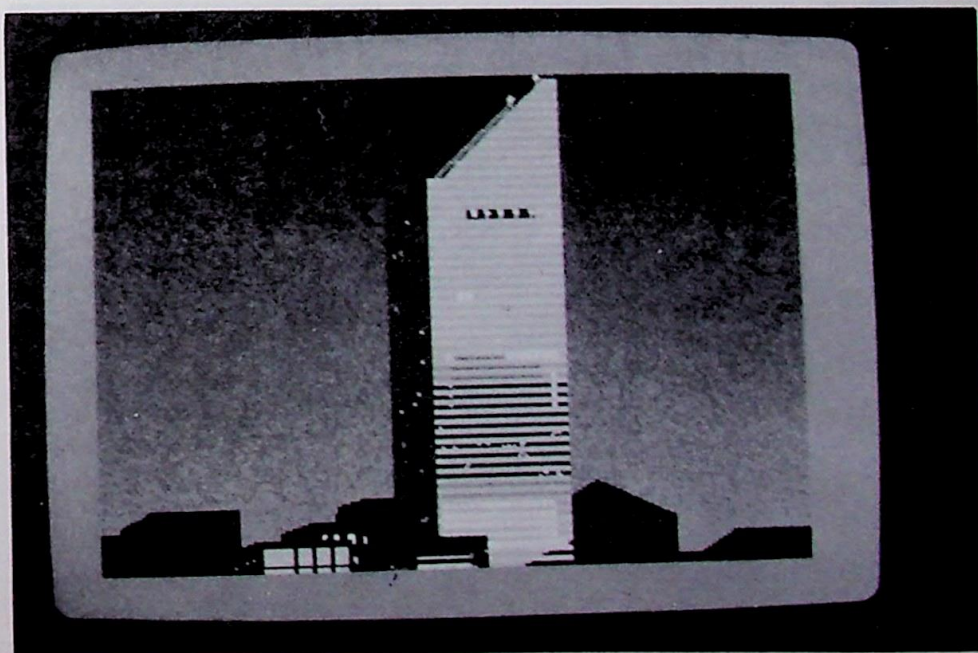
Now that everything is set up, we should start the actual drawing. A good method for creating art is similar to a practice used in creating computer programs: A "top down" design. What I mean by this is start with the general (i.e. a big tall rectangle for the building) and gradually add more and more detail until you have a sophisticated, detailed picture. This way, you don't have to worry about the "whole" thing at any one time.

Now, the first thing to do is choose a color for the background (I chose dark blue for this picture) and fill the screen with it (remember that if the tools are visible in your paint program, the Fill option will not fill under them. You have to either remove the tools and menu before filling the screen, or move the picture over and fill the areas that didn't fill the first time around). Next, I chose the solid box tool and made a medium-blue, tall rectangle that will be the building. To add dimension to it, I chose the darkest color I had created (black) then used the solid box

tool and created a black rectangle (about 25% the width of the first one) just to the left of the blue one. This creates the left side of building. We can worry about the slanted side later.

Front Detail

As you can see by looking at the final picture, there are a few yellow lights on the front and several horizontal, light blue, dark blue, and yellow lines. You must draw these in a specific order. It would be easiest to draw the horizontal lines first, since the little yellow lights are on top of the horizontal lines. Pick any place on the front of the building and draw a line one or two shades darker than the face of the building but NOT the same color as the background. Make sure you don't draw this line over into the black section. Make this line about two to four pixels thick. The thinner you make the lines, the more stories you can make the building appear to have. If your paint program has a pick-up brush tool, then choose it, pick up that line you just made, and keep pasting copies of it all over the front of the building about two to three pixels above and below each other until you have filled the face of the building. All of these lines should be placed an equal distance apart.



Now is a good time to take that gigantic wedge out of the top of the building. To do that, choose the color you filled the screen with earlier and use the filled polygon tool to make a triangle over the top left corner so that the building will come to a point at the top. Go ahead and make this triangle cover the black, left side as well. If you want, you could add a few tiny (2 pixels wide and 2 pixels tall) yellow squares on several of the darker blue lines to give the appearance that some office lights were left on. You could leave it as it is now, but we can make it even better. The glass on the front of the building will reflect other buildings' lights. The closer the glass is to the street, the more lights it will reflect (the top of the building won't reflect many lights since it's so much higher than the surrounding buildings). Choose a bright color (yellow, silver, etc.) and fill the windows on the lower floors to make them appear to reflect the lights of other buildings, or it may even appear to be reflecting the sunset that might be behind the viewer's head. You could put semi-dark vertical lines in these lighter lines to create the illusion of individual windows on the building.

The Sky

The sky might look fine the way it is now, but to me, it looks a little dull. If you look at my picture, you can see that the sky is blue at the bottom and fades to black towards the

top. This gives the illusion of the glowing city lights or maybe the light left over from a recent sunset. To do this, I used the gradient fill option on DeluxePaint II. I selected a color range of four shades, from dark blue to black, then filled the sky. If you don't have a gradient fill option, it will be very difficult to achieve this effect at this point in the drawing. (Note: Now would be a good time to save any work you have already done.)

One way make the sky like this, without using a gradient fill, would be to choose the lightest color you plan to use in the sky (in mine, it's the color closest to the bottom, dark blue) and make a square on the bottom left corner of the screen (far away from the building you've drawn). This square should be about one inch wide and a quarter of the height of the screen. Choose the next darkest color and make a square the same size just above the first one. When stacked on top of each other, these squares should reach to the center of the page. Now choose the next darkest color and make a third box on top of the second and about the same size as the others. Make one final box using the darkest color in the sky (dark-dark blue, or black) and put it above the third one. Keep the color the same and choose the spray paint tool and spray over the border of the top two squares until you can't make out the border. Choose the color of the box just

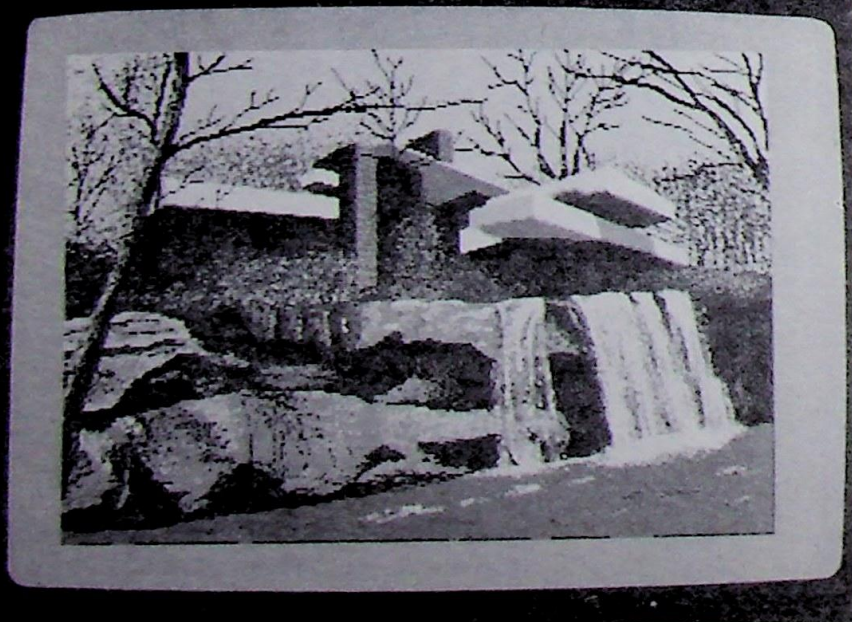
beneath the top box and spray paint a little more. Keep the same color selected and spray over the border of the next one down until the border is gone or hard to see, then choose the next lighter color and continue this process on all of the borders. Once you are finished, pick up that part of the screen with the lasso or pick it up as a brush and paste it over the rest of the sky.

Other Buildings

The picture is almost finished now. Make a few buildings near the bottom of the CitiCorp building using the solid black box tool. Just draw boxes at random over the bottom of the screen until you have a satisfactory sky line. All that remains is to add some lights to the buildings by placing light-colored squares or dots on them.

Frank Lloyd Wright

The second set of pictures on the disk show the steps in creating a completely different style of architectural drawing. Most of the detail in this picture is not in the actual building, but in the landscape around it. There are very few sharp angles or contrasting colors, so this type of drawing is a challenge on a computer. Choosing the right colors for this picture was not an easy task. As you can see by looking at the series of pictures on the disk, I changed the palette AFTER I had already started drawing. Remember, if you are not satisfied with the colors you have chosen, you can always change them. But, sometimes, if you make a drastic change, it can set you back in the drawing because some of the objects you've already drawn may not be the correct color. It is always best to have the "basic" pattern done before you start, that way, slight modifications won't be that harmful later in the development of the image. I knew what materials would be in the picture before I started and chose colors that would match those. The materials I decided would be in the picture were: sky, dirt, rock, concrete, and water. The biggest single color objects in the picture were the sky and the ground, so I chose a blue and a dark brown. For the concrete, I chose a light tan. I knew I'd need at least two or three shades of blue for the water, and three or four shades of tan for the concrete. I can use the same colors



for the concrete and the dirt so I decided to make the lightest color for the concrete and the darkest color for the dirt and spread them together to give me a few intermediate shades of brown (or tan). I also knew there would a lot of detail in this picture and that I couldn't think of every possible combination of colors, so I decided to have a couple of shades of grey too. Grey shades can almost always come in handy. The old scout motto, "Be Prepared" applies even to computer graphics.

I started by filling the screen with the color for the sky then I drew a dark brown "hill" line completely across the screen. I then filled the bottom half of the screen with the brown color. These two colored areas serve as the "foundation" for the rest of the picture.

The House

For the concrete structures in the house, I used the filled polygon tool with the lightest concrete color and drew the right-most sides of the structures. For the adjacent sides of the concrete blocks, I chose a slightly darker shade and drew a slanted rectangle touching the original light sides. This should give the effect of the sun beaming in from the right and highlighting the concrete. For the lower level of the concrete, I cast a shadow on it from the top block. This was done by selecting a couple of shades darker than the lightest one. The rest of the house could be done the same way.

The Rocks

Select a medium to dark shade of brown or grey and draw an outline of the rocks as I have done in the third picture, then fill it. (Make sure you use the solid-line freehand tool to draw the outer boundaries of the rocks so that there are no open spaces when you fill this object). Change your drawing color to black, and spray paint or draw some dark chunks near the bottom of the rocks. Then use the broken-line freehand tool or the spray can tool and put draw some black spots on the rocks. Next, choose some lighter shades of brown or grey, and spray paint near the top of the rocks. Continue this until you've reached the lightest shade of brown or grey, spraying less and less for each consecutive shade. Then, when you've reached the

lightest shade, just put a FEW dots near the upper right regions of the rocks.

Waterfall

I think that the waterfall is what makes this picture. To draw it, I chose the lightest blue shade that I had, a one-pixel brush, and the broken-line freehand tool. It's important that the broken-line freehand tool is used to do this or it will not look at all like it's supposed to. Draw a small horizontal line (a little bit wider than the waterfall will be) at the top of where the falls will be. Now put the pointer anywhere on that line, hold down the mouse button, and pull down, accelerating as you go. Continue this across the line and soon you should have a waterfall. Determine where the water falling hits the pool below and use the small spray can there; moving the mouse right and left, creating sort of a "sparkling waters" line. This should appear to be the waterfall splashing into the pool. Now, choose the smooth tool and go over the water in the falls to create some other shades of blue. If you don't have a smooth tool, just use a small spray can and spray some darker shades of blue, but don't overwrite the light shades at the top of the falls.

The Trees

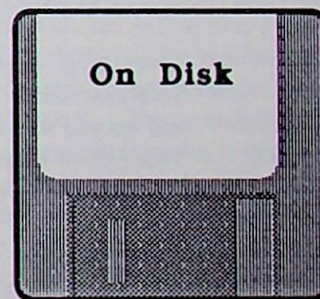
The trees are, by far, the easiest part of this drawing, and with only a little work, you can fill in the voids in your picture with lots of detail. If your paint program has a "stencil" option, use it. Lock out ALL the colors EXCEPT the one color you used for the sky (or if you created your sky like I created mine in the CitiCorp picture, lock all colors except those). This will make it easy to draw trees that appear to be protruding up from behind the house. If you don't have a stencil option, just be VERY, VERY careful and don't draw over what you've already created. Now would be a good time to save what you have already done.

Select a dark brown or black, a one-pixel brush, and the solid-line freehand drawing tool. Draw squiggly lines coming up from the top of the house and from the sides of the screen. To make the lines look more like branches, draw them starting low in the sky

at a LOW angle, then the further up you draw, increase the upwards angle. Draw smaller branches coming off the bigger ones with those smaller ones following the same rules with regard to angle and height.

If you want the trees to look closer, choose a thicker brush and draw the branches even longer. To draw the distant trees, just use the spray paint tool and draw a couple of lines inside the spray painted area. If you want this scene to appear to be in the springtime or summertime, create a green color and just spray paint some leaves over the branches. To make to bark on the trees, use a medium shade of grey and a broken free-hand draw tool and scribble on either the left or right edge of the branches (only do this on ONE side of all the larger branches). Then select a light shade of grey and do the same, but do not put as much light grey on the trees as you did dark or medium grey.

And that's all there is too it! As always, I hope that you enjoy these articles and find them helpful. If you do, and even if you don't, be sure to let me know by writing to me here at GS+!



On the GS+ disk, in the folder, BWG, there are two folders: CitiCorp and Frank.Lloyd. There are 5 pictures in the CitiCorp folder; CitiCorp1 thru CitiCorp5. There are 7 pictures in the Frank.Lloyd folder; Frank.Lloyd1 thru Frank.Lloyd7. These pictures are numbered to show the various steps Michael describes in the article, with the highest numbered picture being the final picture in the series. These are compressed pictures, so you will need to use a program that can read compressed pictures to view them.

RANDOM IIGS PROGRAMMING NOTES

By Steven W. Disbrow

Yes, friends, it's time for yet another update to our ever-popular NDA Text Editor, EGOed. This time around however, it's more than just a simple bug fix! EGOed's user interface has undergone a complete overhaul and several new features have been added to make this most useful of Desk Accessories even more indispensable. For information on the files in this update and how to install EGOed, see "On Disk" at the end of this article.

For those of you just joining us, EGOed is a New Desk Accessory (NDA) Text Editor that allows you to create, edit, and print plain text and AppleWorks Classic Word Processor files from within any application that supports NDAs.

The new features added in EGOed version 1.2 include:

- 1) All known bugs have (as near as I can tell) been fixed. This includes problems that folks were having with copy-protected games and with the Shut Down command in the Finder.
- 2) An honest-to-goodness menu bar. It not only looks better, it also leaves more room in the window for the text you are editing.
- 3) Any changes that you make in Page Setup are permanently saved. This sounds like a very minor thing, but think of it. You can set your page setup to Condensed print once and then forget about it.
- 4) The ability to read and write AppleWorks Classic files. It even reads version 3.0 files! (There are a few catches here, see below for more details.)

So, why a whole new article? Well, some of the things that have been added to EGOed

represent IIGS programming techniques that every IIGS programmer should be familiar with in order to remain on the "cutting edge". So, if you want to know how to put a real menu bar in a window or how to define and use your own custom resources, read on.

But First...

Before we get into the menus and custom resource discussions, we need to talk about what was needed to fix the insidious bugs that were crawling around in EGOed.

For the most part, EGOed has been very bug-free. However, several readers wrote in to say that, when EGOed was installed, some copy-protected games, notably both of the Arkanoid games and Tetris, would crash when you tried to run them. Another error that several folks reported, and that I was experiencing myself, was that, with EGOed installed, the Finder's Shut Down command would not always work properly. While

Figure 1. The EGOed Menu Bar

The File menu contains the following items:		The Edit menu contains the following items:	
Info...	Gives you information about the current file.	Cut	This item removes the current selection from the EGOed window and places it in the Clipboard. This item can be selected from the keyboard by pressing Control-X.
New	This item clears the current file out of memory and lets you start with an empty file.	Copy	This item copies the current selection from the EGOed window and places it in the Clipboard. This item can be selected from the keyboard by pressing Control-C.
Open...	This option loads in either a plain text file or AppleWorks Classic Word Processor file. No formatting is retained when reading in an AppleWorks file. Version 3.0 files can be read, but they might require a bit of "cleaning up".	Paste	This item places the contents of the Clipboard in the EGOed window at the insertion point. This item can be selected from the keyboard by pressing Control-V.
Save	This option will save the current edit file under the same name. If the file is an AppleWorks file, it will be saved as an AppleWorks file, but, all formatting will be lost!!! Be very careful when saving over AppleWorks files!	Clear	This item removes the current selection from the EGOed window. It is not placed in the Clipboard. This item can be selected from the keyboard by pressing clear on the numeric keypad.
Save As Text...	This item will allow you to save the current file as a plain text file.	Choose Font...	This item (formerly its own button), allows you to select a font from one of those installed in your *:System:Fonts folder.
Save As AWP...	This item will allow you to save the current file as an AppleWorks Classic Word Processor file.		
Page Setup...	This item will allow you to specify information about how you want the printed output to appear.		
Print...	This item will allow you to print the current file.		
About EGOed...	This item gives information about EGOed.		

these two problems might sound unrelated, they were both symptoms of a much larger problem.

When you use the Resource Manager, your programs gain a lot of power and flexibility. But, "With great power comes great responsibility," especially if you are wielding this power inside an NDA. One of the main benefits of using the Resource Manager is that it is exceptionally good at managing the memory that it uses. If you open and close your resource file correctly, the Resource Manager will automatically take care of all the memory associated with your resources, and, for the most part, you won't have to worry about it one bit (no pun intended). EGOed, as you might have guessed by now, was not doing things exactly right.

EGOed was opening and closing its resource file each time its window was opened and closed, but in an attempt to save a little time, it was only starting up and shutting down its tools (including the Resource Manager), once. Even though this does not sound like a mistake, it is. NDAs should always shut down any tools they may have started when they are through with them. In this particular case, not properly shutting down EGOed's tools seems to have caused several blocks of memory that were associated with the EGOed resources to remain allocated after EGOed was closed and, more importantly, after the currently running application (usually the Finder) was shut down. These hanging handles were what was causing the problems with the copy-protected games and the Shut Down command.

The solution is to always start up the needed tools when EGOed is opened and always shut those tools down again when EGOed is closed. The drawback to this is that EGOed must go through its start up procedure (check for needed tools, open the resource file, etc.) every time it is opened. This takes some extra time, but it's much quicker than a reboot.

New Features

With the bugs squashed, let's discuss the major change in this version of EGOed: the menu bar (see Figure 1). There are a couple

of reasons for this change, the most important of which is that someone finally told me how to do it! (Thanks to Robert B. Hess, a programmer for a major MICROcomputer SOFTWARE publisher and closet IIGS owner, and Joe Wankerl who can never leave well enough alone...) The concept of a window with its own menu bar is certainly not new, Microsoft Windows and OS/2 are built around it. As IIGS programs get more complex and if Apple and third-party vendors continue to refuse to give us either hierarchical menus or larger video displays, the ability to place menu bars in individual windows will become very important to keeping IIGS programs usable. Having said all that, let's look at just how this is done.

First you have to create the window that you want the menu bar to appear in. Next, create the menu bar so that it will appear in that window. To do this, all you do is pass the appropriate Window Pointer to the `NewMenuBar()` or `NewMenuBar2()` tool call. In EGOed, the code that does this is:

```
EGOedMenuBar :=  
    NewMenuBar2(refIsResource,  
                Ref(EGOedMenuBarRefID),  
                EGOedWinPtr);
```

All this says is that we want to create a menu bar from a menu bar resource and we want it to appear in the window pointed at by `EGOedWinPtr`.

Now that we have our window and our menu bar, we have to make sure that they both get drawn correctly. EGOed does this with the Content Definition Procedure shown in Figure 2. This procedure was associated with the EGOed window when it was first created and it is called whenever the EGOed window needs to be updated. As you can see by looking at Figure 2, this is a simple thing to do.

The last thing that must be considered is, how do we make this menu bar behave like a menu bar? Since it is not the System menu bar, TaskMaster won't handle any events associated with it... or will it? There are at least two ways to trick TaskMaster into helping us out with this. This first is to have an Information Bar in the window and draw the menu bar on top of that. When TaskMaster returns a code of `wInInfo` (mouse clicked in Information Bar), we know that the menu bar was hit and then we can call `MenuSelect()` to handle highlighting, pulling down the menus, etc. If you are not using TaskMaster, use the `FindWindow()` call to determine if the mouse-click occurred in the window's Information Bar.

Unfortunately, using an Information Bar in a window created from a resource is very, very tricky. When I tried it, the system would crash whenever I tried to move the window. Not exactly what I had hoped for. So, after some experimentation, I stumbled on a second method that probably would not work if EGOed were not a text editor. Because EGOed is a text editor, the Text Edit control that is used for editing text needs to take up as much of the window as possible. When the mouse is clicked in this area, TaskMaster and the Text Edit Tool Set handle everything without the need for any action on the part of the application. The remainder of the window (where the menu bar is drawn) is not covered by any controls, so when the mouse is clicked there, TaskMaster returns a value of `wInContent` (mouse clicked in the content region of window, but not in a control). When this happens we know that the mouse was clicked in the menu bar and we can call `MenuSelect()` to handle pulling down the menus, highlighting them, etc. That's all there is to it.

Figure 2.

```
Procedure DrawEGOedWindow;  
Begin  
    DrawControls(EGOedWinPtr); (* Draw the Text Edit Control *)  
    SetMenuBar(EGOedMenuBar); (* Switch to my Menu Bar *)  
    DrawMenuBar; (* and draw it. *)  
    SetMenuBar(Nil); (* Switch back to system bar *)  
End;
```


The second new feature in this version of EGOed that it automatically saves the currently defined page setup in a resource. If, for example, you choose Page Setup from the File menu, change the vertical sizing to Condensed and click on the OK button, this new page setup will be saved in the EGOed resource file and will be loaded each time you open EGOed. Being able to work with resources in this manner is something that all IIGS programmers should become familiar with very quickly.

Trying for the last 10 hours to work up an explanation for this, I keep reprinting the Resource Manager chapter of Volume 3 of the Toolbox Reference and the EGOed source code. So, rather than have a 90 page magazine and a lawsuit on my hands, I'll just refer you to the `DAOpen()`, `DAAction()` and `GetPrintRecord()` procedures in the EGOed source code. I've commented these sections heavily so you should be able to get a good feel for what is needed to define and use resources in this manner. It really is easy once you work with it a little.

The last new feature in this version of EGOed is the ability to read and write AppleWorks Classic Word Processor files. EGOed will even read in Version 3.0 files, but, you might have to clean them up a little if they contained any of those new Version 3.0 tabs.

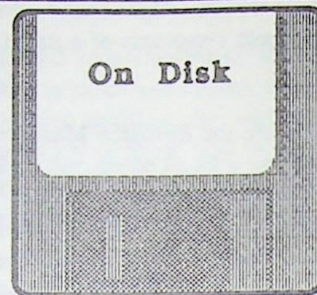
The main thing to remember when using EGOed to work with AppleWorks files is that no formatting is retained when either reading or writing AppleWorks files! If you load an AppleWorks file into EGOed and then save it, the resulting file will have no

formatting. If you save the file with the same name, you will lose the formatting! In other words, **BE CAREFUL!!!!**

The two procedures (`FixAWPFile()` and `MakeAWPFile()`) that do the conversion from AppleWorks to text and visa-versa are pretty much self-contained, so you should be able to easily use them in your own programs. If you want more information about how AppleWorks stores its files, be sure to get a copy of Apple File Type Note FTN.1A.XXXX. This note is available from any of the major online services and our own pro-gsplus BBS.

The End? No Way!

Over the last few months, EGOed has proven itself to be a very useful little utility. In the coming months, I plan on making it even more useful by rewriting it in C (so it will be smaller), fixing any new bugs that are reported and adding the ability to read AppleWorks GS files. If you have a request for a new feature you want to see, be sure and let me know about it.



This EGOed update is in the folder named `EGOed.1.2` on your GS+ disk. There are four files in this folder:

<code>EGOed</code>	This is the EGOed NDA.
<code>EGOed.p</code>	This is the TML Pascal II source code for EGOed.
<code>EGOed.r</code>	This is the TML Pascal II resource file for EGOed.
<code>EGOed.rez</code>	This is the REZ language file that is used to create the file <code>EGOed.r</code> .

EGOed works only with System Software v5.0 and later. This means that your system must have at least 512K of memory (but more is better). It **WILL NOT WORK** with System Software v4.0! This is because several of the tools required by EGOed (the Text Edit Tool Set and Resource Manager in particular) do not exist in System 4.0. To install EGOed, use the Finder to copy the file `EGOed` from the `EGOed.1.2` folder on your GS+ disk to the `System:Desk.Accs:` folder on your startup disk. **Do NOT rename the EGOed file!** Once you have the file copied, you must restart your IIGS to make EGOed available from the Apple menu.

MOVING?

Well don't forget to tell us! Simply remove your mailing label from a previous issue of **GS+**, affix it to a change of address form (available at your local post office) and send it to us at:

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

RUMORS, WISHES & BLATANT LIES

Yes Virginia...

There will be an AppleFest in May! At least that's what the folks that answer the phone at 1-800-262-FEST tell us. According to them, the reason that there has been no advertising published and no brochures distributed is that their printer screwed up. Well, at any rate, this year's AppleFest is scheduled to take place the first week in May, and the location has been moved from historic Boston to... um... spacious Somerset, NJ. The conference will be held in the Garden State Convention Center, and the hotel of choice (i.e. the closest) will be the Somerset Hilton. The costs for this three-day extravaganza are:

- Exhibits only: \$10
- Exhibits and Conferences: \$45
- Seminars: \$99 for the first Seminar, \$75 for each additional seminar.

Nuclear Regulatory Commission

Our contact at Applied Engineering, Mr. Busy Signal (Mr. BS for short), tells us that, pending NRC approval, the Romulan and Klingon hard drives will both be ready in time for AppleFest. A final price was not set for either drive, but, Mr. BS told us that potential buyers should seriously consider taking out that second mortgage while interest rates are still low.

More BS

Mr. BS also told us that AE will have two other new IIGS specific products to show at AppleFest. While he would not tell us what the products did, he did say that neither of these products (code named "9600 Baud Modem" and "Full Page Monochrome Display") would be released until the marketing people can figure out which names to steal from the Star Trek® Universe.

Apple Extends Warranty?

We could not get confirmation from any dealers by press time, but apparently Apple has finally conceded to consumer demand and increased the warranty period on their products from 90 days to 1 full year. It's about time!

ROM 04

Rumors continue to mount concerning the new IIGS that Apple supposedly has in development. Everyone seems to know exactly what the capabilities of this new machine will be, but no two stories ever agree. Some of the best we have heard include:

- 640x480 graphics with 256 colors on each scan line.
- A built-in SCSI port (finally!).
- A peppy little 7MHz 65816 processor.
- A Graphics Coprocessor.
- A built-in Stereo Output jack (This one is a Blatant Lie. A Stereo Output Jack would make far too much sense for Apple to actually include one.)

But, our favorite is the rumor that EVERY IIGS owner will be able to upgrade to this new machine for a minimal cost. According to this rumor, the reason that Apple would not let owners of ROM 01 machines upgrade to the ROM 03 machine was that the newer and more powerful ROM 04 machine was waiting in the wings. So, Apple may have actually been trying to SAVE us money!

Don't Call Us and We Won't Call You...

After months of trying, Apple has finally reorganized enough to force VP Jean Louis Gasee to announce that he will soon be leaving. Too bad. Does this mean he won't be at AppleFest? Awww....

HyperCard GS

About 900 people (give or take 850) have gone out of their way to let us know that HyperCard for the IIGS will be released in the next few days. (Probably while this is at the printer.) Among the features HyperCard GS is supposed to have is the ability to run Macintosh HyperCard stacks. This goes hand in hand with another rumor that says Apple will be releasing IIGS System Disk 6.0 in a few days as well. If this is the case, we can probably expect to see a Macintosh Hierarchical Filing System (HFS) File System Translator (FST) included in this new System Disk so that HyperCard GS can read those Macintosh HyperCard Stacks straight from Macintosh Disks!

Wow!

Have you heard the one about the 20MHz IIGS accelerator that's currently in development? Look for it in a month or so. Oh, you might as well hold your breath too.

¡Hóla, Amiga!

Apple Computers, Inc. claims the Apple II will be around for a long time. But Applied Engineering isn't taking any chances. Not only are they looking for Macintosh programmers, they are also looking for Amiga programmers.

Have you heard a good one lately? Share it with us here at GS+ and we'll, um, not print your name or turn you over to the cops. Send those rumors to:

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

The Top 10 Things To Do At This May's AppleFest

10. Avoid Paul Statt.
9. Spend yourself into the poorhouse.
8. Actively ignore all of the companies that bring Macintoshes.
7. Buy that swell Disbrow guy a drink.
6. Try to find someone from Claris.
5. Try to get a straight answer from an Apple Representative.
4. Find a rest room.
3. Try to sneak into seminars that you did not pay for.
2. Make "Busy-Signal" noises at the people from Applied Engineering.
1. Grab as many company buttons as you can, pin them to your shirt and pretend that you look cool.

HOW TO USE THE GS+ DISK

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

There are 8 folders on this issue's Disk. In alphabetical order, they are:

BWG

This folder contains two other folders: **CitiCorp** and **Frank.Lloyd**. These folders contain the pictures discussed in *Brush With Greatness*. The **CitiCorp** folder contains five pictures: **CitiCorp1** through **CitiCorp5**. The **Frank.Lloyd** folder contains seven pictures: **Frank.Lloyd1** through **Frank.Lloyd7**.

These pictures are stored in Apple Preferred Format (i.e. they are compressed), so you will need to use a paint program such as *DeluxePaint* or *PaintWorks Gold* to view them.

Compat.Guide

Our Compatibility Guide department is **STILL** on leave this issue, but it definitely **WILL** return next time. In the mean time, we have once again included, in the **Compat.Guide** folder, the complete text of our first two installments of this column. These two files, **V1.N1.COMPAT** and **V1.N2.COMPAT** are plain text files that you can read with **EGOed** or any other program that can read text files.

CompEyesRev

This folder contains several pictures that we captured during our review of *Computer Eyes*. There are six pictures in this folder. They are:

- Creative.Team** - A touching picture of the people that bring you **GS+**
- Diz.Bunny.Idea** - The picture that was the inspiration for this issue's cover.
- Diz.What.The** - Stump the publisher!

Motley.Crew - Michael Quinn, Joe Wankerl, Diz, and Nory consider mirror technology

Pliable.Nory - Nory at deadline...

Sticky.Fingers - Yikes!

None of these pictures is compressed, so you should be able to use any **IIGS** paint program to view them.

DeskColorCDEV

This folder contains 6 files:

DC.Icon.asm - This is the Assembly Language code for the Desk Color Icon as generated by **DicEd**.

DColor.cc - This is the **ORCA/C** source code for the Desk Color **CDEVCodeResource**.

DeskColor.Icon - This is Finder Icon file that **DC.Icon.asm** was generated from.

DeskColor.CDEV - This is the Desk Color **CDEV**. Using the Finder, copy it into the **System:CDEVs:** directory of your startup disk. It will then be available from the Control Panel **NDA**.

DColor.rez - This is the **REZ** language code for the Desk Color **CDEV**

Make - These files are used to compile and link the Desk Color **CDEV**

Make2

Make3

For more information on using the Desk Color **CDEV**, see *How to Write a Control Panel Device* on page 7.

EGOed.1.2

The **EGOed** update is in the folder named **EGOed.1.2** on your **GS+** disk. There are four files in this folder:

EGOed - This is the **EGOed NDA**.

EGOed.p - This is the **TML Pascal II** source code for **EGOed**.

EGOed.r - This is the **TML Pascal II** resource file for **EGOed**.

EGOed.rez - This is the **REZ** language file that is used to create the file **EGOed.r**.

EGOed works only with **System Software v5.0** and later. This means that your system must have at least **512K** of memory (but more is better). It **WILL NOT WORK** with **System Software v4.0!** This is because several of the tools required by **EGOed** (the **Text Edit Tool Set** and **Resource Manager** in particular) do not exist in **System 4.0**. To install **EGOed**, use the Finder to copy the file **EGOed** from the **EGOed.1.2** folder on your **GS+** disk to the **System:Desk.Accs:** folder on your startup disk. Do **NOT** rename the **EGOed** file! Once you have the file copied, you must restart your **IIGS** to make **EGOed** available from the Apple menu.

Icons

This folder contains the Icons discussed in the Icons column on page 21. To use them, copy them into the **Icons** folder of the disks that you run each of these programs from. The next time you are in the Finder, you should see the new icons for these programs.

JamSessionRev

There are four files in this directory. They are:

Jam.Song.1

Jam.Song.2

Jam.Song.3

Jam.Song.4

These are songs created by Jeff Walker during his review of *Jam Session*. You must have *Jam Session* in order to play these songs.

Writers.Guide

There is only one file in this folder: **Writers.Guide**. This is a plain text file that tells you what you need to do to write reviews, articles, programs, etc. for **GS+**. Use **EGOed** or any other program that can read text files to read or print this file.



ICONS

This issue, we have a bunch of GREAT Icons for you. Jami Lowery, a friend of ours here in beautiful Chattanooga, TN, recently got herself a IIGS, a hard drive and a copy of DICEd v1.2, and went just a bit berserk. Thankfully, she decided to share the results of her Icon insanity with us, and so, in the Icons folder on this issue's GS+ disk, you will find Icons for:

Battle Chess
Tunnels Of Armageddon
Computer Eyes
SoundSmith
Instant Music
NeuroMancer

Aside from Jami's fabulous Icons, you will also find a fairly unimpressive little Icon for Jam Session, done by the guy on the cover.

To use these Icons, just use the Finder to copy them into the Icons folder of the disk that you run the program from.

Jami has lots more Icons in the works, so look for them in the coming issues of GS+. If you have any neat Icons of your own, send them in and share them with the rest of the Finder fanatics out there!

DISKLESS?

If you did not receive the disk with this magazine and have decided you would like to have it, just send a check or money order for \$6.00 (\$5.00 plus \$1.00 shipping and handling) to:

GS+ V1N4 Disk Offer
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415-0366

Or, call us at (615) 870-4960 to bill it to your MasterCard or VISA.

Tennessee residents add 5.5% sales tax.
Chattanooga residents add 7.25% sales tax.

LRO Computer Sales

Memory

Applied Ingenuity

GS Juice Plus
 0k - \$70; 1 meg - \$149; 2 meg - \$228;
 4 meg - \$384

GS Ram Plus

1 meg - \$284; 2 meg - \$375; 4 meg - \$552

Checkmate Memory Saver - \$119

1 meg 80ns expansion sets for GS Ram Plus
 and GS Juice Plus w/ 5 year warranty - \$ 80

256k 120ns expansion sets for most
 Apple IIe/c/gs Exp. boards w/ 5 year warranty - \$ 26

256k expansion sets for GS Ram Ultra
 and Ram Pack4GS - \$ 22

Complete GS Configurations

GS Starter System

Apple IIgs with 1 meg CPU, Mouse & Keyboard
 Magnavox RGB Monitor
 Fortis Imagewriter Compatible Printer
 AMR 3.5 Drive
 Mouse Pad
 Box of 10 3.5" Maxell Diskettes

- \$1599

Apple IIgs 1 meg CPU, Mouse & Keyboard
 Apple Color RGB monitor
 AMR 3.5 Drive
 Apple Imagewriter II color printer with cable
 Applied Ingenuity 20 meg OverDrive
 Box of 10 Maxell 3.5" Diskettes
 Mouse Pad

- \$2499

Software

Utilities

Copy II Plus 9.0 - \$25
 Salvation GS HD Backup - \$37
 Print Shop GS - \$27
 Prosel 16 - \$66

Graphic Disk Labeler

Print Color Disk Labels on your IW II!

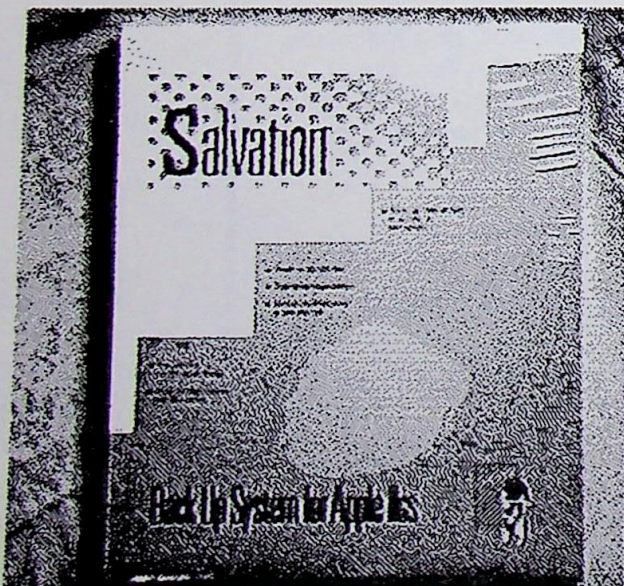
\$24

Business

AppleWorks GS - \$212
 Manzanita Businessworks - \$294

Education

Designasaurus GS - \$33
 Geometry GS - \$56
 Talking Once Upon a Time - \$34



Salvation GS - \$37

1-800-869-9152 M-F, 9-6. Sat. 12-4

More Software...

Games

FutureShock Entertainment system w/ FuturePad	- \$54
Test Drive II	- \$34
Mean 18	- \$26
Jigsaw	- \$26
Chessmaster 2100	- \$37
Tunnels of Armageddon	- \$32
Zany Golf	- \$25

Numerics GS

Amazing math program that does most math functions from Algebra to Trig!

\$104

Drives

Chinook 100 meg HD w/ Apple SCSI	- \$1299
CMS 60 meg HD w/ Apple SCSI	- \$859
CMS Removable 45meg HD w/ Apple SCSI	- \$979
Applied Ingenuity OverDrive 40 meg	- \$509
Applied Ingenuity OverDrive 20 meg	- \$449
Applied Ingenuity InnerDrive 40 meg	- \$599
Applied Ingenuity InnerDrive 20 meg	- \$479

AMR 100% Apple compatible 3.5 in. Drive	- \$189
AMR 100% Apple compatible 5.25 in. Drive	- \$154
Apple 3.5 in. Drive	- \$324

Accessories for the GS

Transwarp GS	- \$289
Sonic Blaster GS stereo board	- \$ 96
Sound System II speakers for GS	- \$ 99
Visionary GS realtime color digitizing board	- \$279
Quickie Hand Scanner (400 DPI)	- \$259
Conserver GS	- \$ 89
System Saver	- \$ 69
A+ Optical Mouse ADB	- \$ 87
Cordless Mouse	- \$109
Nite Owl clip-on GS battery	- \$ 14
Applied Ingenuity VCR backup	- \$174

Cables

//GS to Modem	- \$ 15
//GS to IW II/LQ	- \$ 16



FutureShock 2.0 - \$54

Modems

USR 14.4 Kps Courier HST w/ cable	- \$594
USR Dual Standard w/ cable	- \$999
Prometheus Promodem 2400G internal	- \$144
Cardinal 2400 baud with DSP and cable	- \$134
Supra 2400 w/ cable	- \$137

Individual GS Hardware

Apple Iigs 1 meg CPU with Keyboard and Mouse	- \$849
Apple Color RGB monitor	- \$479
Magnavox RGB monitor	- \$319
Apple ImageWriter II color printer w/ cable	- \$465
Fortis ImageWriter-compatible printer w/ cable	- \$229
Apple II SCSI	- \$ 99

LRO Computer Sales
665 West Jackson Street
Woodstock, IL 60098

(815) 338-8685

(800) 869-9152

Visa / Mastercard / Check / COD. Prices subject to change without notice. Illinois residents add 6.5% sales tax.



1-800-869-9152 M-F, 9-6, Sat. 12-4

GS+ CLASSIFIEDS

Disk Label Printer Pro v3.0

Makes labels for 3.5" disks. Reads all the ProDOS files including subdirectories, up to 700 files. Edit files to be shown. Also reads DOS 3.3, CP/M, ProDOS and Pascal files from 5.25" disks. Prints on 4" or 5" labels. Makes AppleWorks Database file of all files. Requires IIe, IIc, IIc+, IIGS with 128K, ImageWriter or Epson printer. Price: \$30 post paid.

Hardsoft Products
P.O. Box 90012
Honolulu, HI 96835
(808) 737-1763

JumpStart

The easy-to-use utility that launches programs that the FINDER can't! Maintains an unlimited run list of programs named the way YOU want. Latest Version 2.6's full-featured utilities include format/erase, disk and file copy, lock/unlock, delete, rename, change file type, enable/disable CDAs, NDAs etc. Handles GS/OS resource forks, provides up to 64K printer buffer, and much more! Shareware for \$29.95, available from the author, or CompuServe, GENie, or America Online. VISA/MasterCard.

MaineFrame Software
Box 315A, Cousins Island
Yarmouth, ME 04096
(207) 846-6745

Software for Sale:

TML Pascal II with Introduction to GS programming and Pascal learning books - \$100. Designasaurus, Jigsaw, Postcards, Thexder, Zany Golf, 4th & Inches, HardBall, Mean 18 - \$15 each. Money Orders Only.

Phill Borrero
123-3 Maine St.
Travis AFB, CA 94535
(707) 437-5024

1/2 Price Software:

Ancient Land of Ys.....\$22.50
Gnarly Golf.....\$15.00
Jam Session.....\$25.00
Laser Force.....\$15.00
Life & Death.....\$25.00
Rocket Ranger.....\$25.00
The Three Stooges.....\$25.00
Where in the World is
Carmen Sandiego?.....\$22.50

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

Readers can place an ad in the *GS+* Classifieds for only \$5. This cost buys 25 words in one issue of *GS+*. Additional words are just 10 cents each. The *GS+* Classifieds are the perfect way to contact all of the other IIGS owners out there. The deadline for inclusion of a classified ad in the next issue (Volume 1, Number 5) of *GS+* is May 1, 1990. Simply fill out a photocopy of the coupon below; or send your ad along with your name, address, phone number, number of issues to run, and payment (**made payable to EGO Systems**) to us here at *GS+*; or call us at (615) 870-4960 to place an ad with your MasterCard or VISA.

GS+ CLASSIFIED AD ORDER FORM

Ad copy: _____

Number of issues to run: ____ Total number of words: ____ Total enclosed: \$ ____

Name: _____ Phone: _____

Address: _____

City: _____ State: _____ Zip: _____

PROGRAMMER'S QUEUE & A

(continued from page 3)

ResultBuf255Hndl is simply the name of a data type defined by Apple for use with GS/OS. Since it was defined by Apple both TML Pascal II and ORCA/Pascal should both use the same definition for it. In other words, it should be portable from TML to ORCA and visa-versa.

StuffHex() is a different story. StuffHex() is a dandy little extension built into TML Pascal II that allows you to "Stuff" hex values into a variable. This makes it simple to define cursor shapes, icons, patterns, etc., without having to resort to Assembly Language or a million assignment statements. Unfortunately, ORCA/Pascal has no comparable statement (that I know of), so your best bet here is to

either use a bunch of assignment statements or define and initialize the needed variables in a separate Assembly Language program and link the two together.

That's all we have room for this time. If you have a question about programming the IIGS, send it to Programmer's Queue & A care of *GS+* Magazine.

REVIEWS

InnerDrive

20 Megabytes - retail \$499
40 Megabytes - retail \$699

Ingenuity, Inc.
14922M Ramona Blvd.
Baldwin Park, CA 91706
(800) 346-0811 or (818) 960-1485

Vs.

Vulcan

20 Megabytes - retail \$649
40 Megabytes - retail \$849

Applied Engineering
POB 5100
Carrollton, TX 75011
(214) 241-6060

Reviewed by Joe Wankerl

Now that two companies are making internal hard drives for the GS, it's about time that some comparison be drawn between them. Applied Engineering's Vulcan and Ingenuity, Inc.'s (formerly Applied Ingenuity) InnerDrive are seen by most as the same product. While the drives are incredibly similar, there are a few differences, other than price, that potential buyers should be aware of. In this article, we will look at differences in each drive's software, features, speed, and the technical support from their manufacturers.

Software

The InnerDrive comes with a 3.5 inch disk formatted with GS/OS and System Disk 5.0.2 installed. There are four major utilities on this disk: Backup II, two head parking programs (one for ProDOS 8 and one for GS/OS), Disk Doctor, and a drive formatting program.

Backup II is Apple's standard ProDOS 8, incremental file backup utility. Why Backup II is included with these drives is a mystery. Backup II refuses to work with GS/OS Extended Files (i.e. files with Resource Forks) which means that you can't even backup the software included with these drives. Both Applied Engineering and Ingenuity should begin shipping their drives with either Salvation or ProSEL 16, both of which can handle Extended Files.

A head park program moves the read/write heads of the hard drive to a safe area so that

they do not accidentally crash against the disk surface and damage it. Usually you don't have to park the heads unless you are planning to move the hard drive.

Disk Doctor is a hard drive repair utility. It scans the surface of the drive for defects, marks the bad areas, then maps some additional drive space to make up for the bad blocks. This program may make you wonder about the reliability of hard drives. Not to worry, though. Only under bizarre circumstances will a hard drive develop a bad block.

The formatting program changes the drive's partition sizes and does both a low-level and high-level format of the drive. A low-level format prepares a disk for use by writing basic structural information to the surface of the disk. A high-level format writes directory information to the disk so an operating system will recognize it.

The Vulcan comes with a 3.5 inch as well as a 5.25 inch disk. The 3.5 disk is formatted with GS/OS and has System Disk 5.0.2 installed. The 5.25 disk is double sided, one side is formatted with ProDOS 8 and the other side is formatted with DOS 3.3. The 3.5 disk and the ProDOS side of the 5.25 come with Backup II, a ProDOS 8 head parking utility, and a partition manager.

The partition manager program is used to change the names of the partitions (under GS/OS, this is the same as changing the device name), change the size of each partition, select the file system used by each partition, select write protection for

individual partitions, select whether or not a partition is online, and to indicate which partition is the boot partition.

Both manufactures supply custom GS/OS device drivers that speed up access to the hard drive. The InnerDrive's driver indicates drive access by placing a flashing block in the upper right hand corner of the screen. Also, with the InnerDrive's driver installed, selecting Shutdown from the Finder will automatically park the heads. This is a super nice feature that, unfortunately, is missing from the Vulcan. The Vulcan has a drive access indicator L.E.D. on its controller card. Unless you operate your computer with the top off (an FCC no-no), this internal lighting arrangement is only useful in determining if the drive is functional when you first install it. If you wish, you can purchase an external drive access light from Applied Engineering, or you could make your own (they supply the technical specifications for creating such an attachment), so that you won't be left "in the dark" wondering what your drive is doing.

Also common to each drive is a Classic Desk Accessory (CDA). The InnerDrive's CDA shows information about the drive such as when it was formatted and the number of partitions, as well as a bunch of other fairly useless information. The Vulcan CDA, on the other hand, is truly useful. It allows you to select which partition the hard drive will boot from, select write protection on individual partitions, and park the heads, thus making a separate GS/OS park program unnecessary.

Abilities

The InnerDrive works only with ProDOS partitions. If you have any CP/M, Pascal, or DOS 3.3 programs you want to put on the drive, forget it. If not, then this is no big deal.

The Vulcan allows partitioning for CP/M, Pascal, DOS 3.3, and ProDOS. Also, the Vulcan can "turn off" partitions so they are not accessible - a great help for ProDOS 8, which can only recognize 8 devices at a time. The Vulcan can also write protect a partition and allows booting from any active partition. This lets you put GS/OS on one partition and ProDOS 8 on another and start your computer from whichever one you desire. The InnerDrive only boots from the first partition and partitions can not be "turned off" or write-protected.

Speed

And now, here's what you've all been waiting for: The speed comparison. I compared times from a PR#7 (from the AppleSoft "J" prompt, this command will activate the device in slot seven, i.e. boot the hard drive) until the Finder menu bar appeared. I also compared times from the Finder to AppleWorks GS version 1.1, with no modules pre-loaded. Then I compared loading (still, with no pre-loaded modules) and saving an AppleWorks GS 13K word processor document. And I did all this with and without the custom GS/OS drivers. The results of these tests are shown in Figure 1.

Service

I made up some problems with both hard drives and wanted to see how well each company could solve my dilemma. First I tried Ingenuity. (The support guy answered the phone "Applied Ingenuity." Apparently someone forgot to tell him of the company's name change.) I described my problem to him as being that I put the hard drive controller in slot six and now it wouldn't boot. He asked me if I had my control panel settings correct. I said I thought I did. He asked me if I had the Startup Slot set to six. I said yes. He was baffled. He said it should work. Well, after a couple of minutes, he

then asked me if I had slot six set to "Your Card." I said no and then that fixed the problem. Not to let him off that easily, I then asked if I could use my 5.25 drives with the control panel settings that way. He said of course because the Smart Port is slot five. Well I think he needs to go back to basic training course. Even though the Smart Port is slot five, since the hard drive is using slot six, the 5.25 drives cannot be there as well.

Next I tried Applied Engineering. They were very difficult to get in touch with. For three days all I got was a busy signal. But finally I got a line through. I described my problem to the support guy as being that I put my hard drive in slot six and when I booted the computer, I got a beep and everything stopped. He asked me if I had my dip switches for pseudo slotting set correctly. I said I guess so. He asked me what they were and I told him. He took a minute or two to find the correct settings. He told me to try them and then that fixed the problem. Before I let him go, I asked him why they were still shipping Backup II with their drives. He said that it was because it was there when they were shipping the drives with System Disk 4.0 and nobody had bothered to change it.

My overall impressions from both companies after these interviews were that Ingenuity needed more knowledgeable people to staff their support lines and that Applied Engineering needed more people to staff their support lines. But both companies did

manage to find and fix my problem within a reasonable amount of time. I was generally pleased with the way I was treated by each company.

Conclusions

As you can plainly see, the difference in speeds are almost negligible. The ability to do a low-level format and the automatic head parking on the InnerDrive make it a formidable competitor. But, the Vulcan, with its capability to have different file system partitions, write protection, change the boot partition, and park heads from a CDA, puts it ahead, in my opinion. Also, Applied Engineering seems to be the more stable company. They have been in the business of producing Apple II specific peripherals for many years and will probably be around for quite a while. Ingenuity is a relatively young company that seems to be going through a bit of an identity crisis. However, they were the first company to make an internal hard drive for the IIGS. If it weren't for the success of the InnerDrive, there probably would not be a Vulcan. So, for most folks, the choice will come down to cost. If you need a great hard drive and want to spend the least amount possible, get the InnerDrive. If you need a great hard drive with some great extras, and you don't mind spending a little more for those extras, get the Vulcan. It would be tough to go wrong with either drive.

Figure 1. Speed Comparison

	Vulcan	InnerDrive
With Driver:		
PR#7 to Finder -	19.9	19.9
Finder to AppleWorks GS 1.1 -	18.1	20.1
Loading 13K document -	4.2	4.2
Saving 13K document -	2.7	2.7
Without Driver:		
PR#7 to Finder -	28.1	27.1
Finder to AppleWorks GS 1.1 -	29.9	30.2
Loading 13K document -	6.7	6.7
Saving 13K document -	3.1	3.1

Salvation - retail \$49

Not Copy-Protected - Requires 512K RAM.
Program by Christopher Warner.

Vitesse, Inc.
13909 Amar Road, Suite 2
La Puente, CA 91746
Orders: (800) 777-7344
Information: (818) 813-1270
Technical Support: (818) 813-1274
FAX: (818) 813-1273

Reviewed by Steven W. Disbrow

Over the last few years, the number of IIGS users that own hard drives has steadily increased. With the growing size of GS/OS and IIGS application programs, the storage capabilities and speed of a hard drive have become a necessity for IIGS owners that value their time and sanity.

The only major problem with a hard drive comes from one of its major strengths: Storage capacity. If you have 30 Megabytes of original work on your hard drive, what do you do if it becomes damaged somehow? It could take months to try and recreate all of the lost information and, even then, some of the information will still almost certainly be lost.

The solution is to keep backup copies of the information on your hard drive. If the files on the hard drive become damaged, you simply restore the information from the backups to the hard drive. As you might imagine, this can be a difficult task to do by hand. What's needed is a program specifically designed to perform back up and restoration of data. At least two programs arose to help with this task (Easy Drive and ProSel 16) but neither of these programs took full advantage of the capabilities of the IIGS. Salvation, on the other hand, is a full-featured back up and restore program that takes FULL advantage of everything the IIGS has to offer.

The Good Stuff

Salvation is such a wonderful product that it's been tough to decide where to start this review. So, I'll start with the first thing you

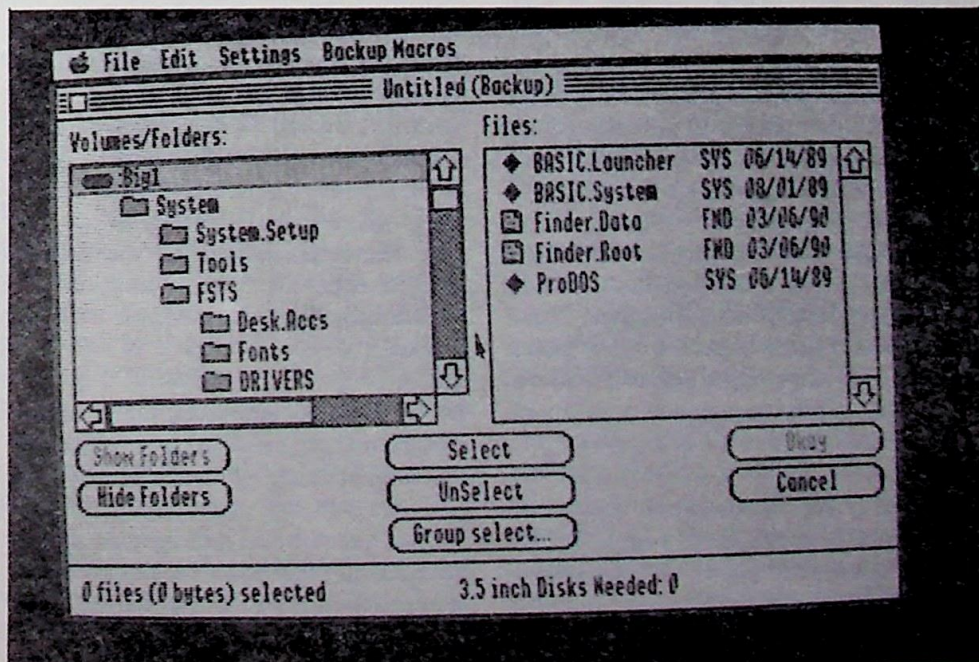
see when you open the box, the manual. Ordinarily, I don't read the manual until I'm knee-deep into a new program. For reasons still not exactly clear to me, I made an exception with Salvation, and I did not regret it. The first part of the manual is an "Introduction And Overview" that explains exactly what Salvation can do and how the rest of the manual is laid out. The next section of the manual is for experienced users that want to get right to the business of backing up. The third portion is for beginners or novices that might not be entirely comfortable with their IIGS. The final section contains detailed descriptions of Salvation's more advanced features. Last, but not least, there is a complete index.

After I finished reading the Salvation documentation, I felt as if I had already used the program. Every aspect of the program, from the contents of the menu bar to the effect of drive speed on the time required for a backup, is clearly explained in the manual. The Salvation User's Guide is an exceptionally well-written piece of documentation, one that every software manufacturer should take a close look at.

Thinking that the things I had read in the manual were too good to be true, I proceeded to install Salvation on my hard drive and make a long overdue backup. As you might

expect, Salvation is not copy-protected and it installs easily on a hard drive. Once the Salvation menu bar appears on your screen, you have several courses of action. You can begin a new backup, continue a backup that you may have previously stopped, restore files that you have previously backed up, or you can quit. The first thing I needed to do was to start a new backup.

The first step in backing up files is selecting the files that you want to back up. To accomplish this, Salvation first shows you a list of all disks that you have online. You then double-click on the disk (or disks) that you want to back up, and Salvation automatically calculates the number of 3.5" disks that will be required. If you don't want to back up the entire contents of a disk, Salvation can show you a hierarchical listing of the directory structure on the disk and you can then double-click on the individual files and/or directories that you want to back up. This ultra-simple method of file selection comes from a strict adherence to Apple's Human Interface guidelines and makes it unnecessary to remember or type long, hard-to-remember pathnames. However, this is not the only way to select files with Salvation. The Group Select feature allows you to select additional files for backup by several different criteria. You can select only the files that have been modified since you



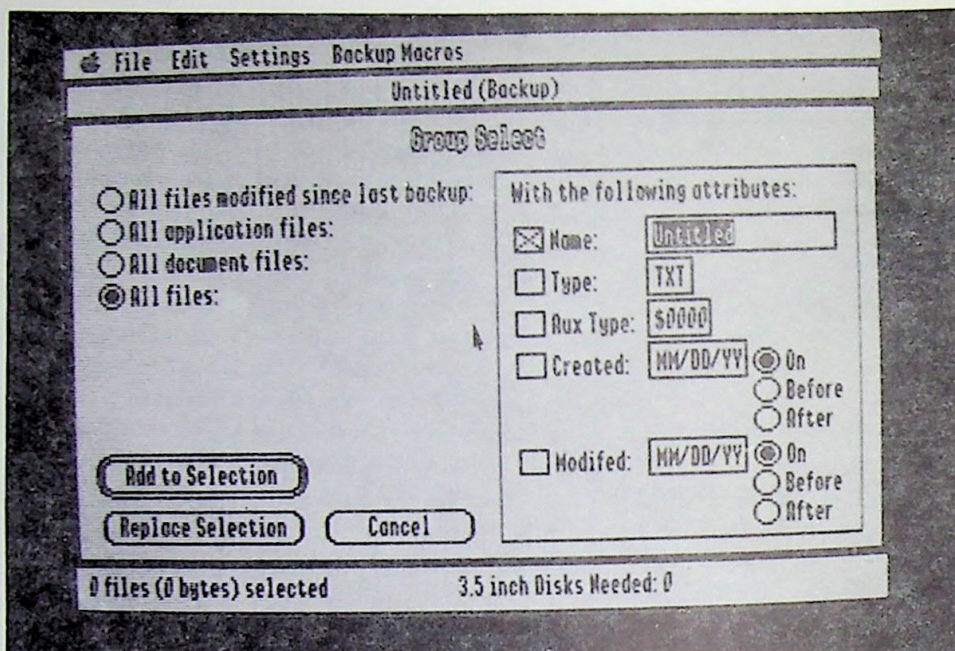
last backed up. You can select only application programs, only document files (i.e. not applications); Salvation will even let you group select files by Name, File Type, Auxiliary Type, and by the dates that files were created and last modified!

If you end up with a particularly complex set of selection criteria, or if you will be using the same criteria over and over again, Salvation will let you save those criteria in a Backup Macro. These Backup Macros can be activated with a single keystroke (Open-Apple-0 through Open-Apple-9), so the next time you need to back up that particular set of files, you only to need press two keys to start the backup!

Well, all I needed at this point was a total backup of my hard drive. So, I double clicked on **:Big2** and waited for Salvation to tell me how many 3.5" disks it would need to do the backup... 22! Plus 1 catalog disk! Wow! This was the first disappointment I had with Salvation: it does not do any data compression when backing up files. For large hard drives this means that you will need LOTS of 3.5" disks for your backups.

So now we get to the bit everyone wants to know about, how fast is Salvation? The answer is: very fast. There were 1815 files for a total of 17163K (17,574,912) bytes selected for that backup. I started at 3:05 P.M. Thirty-two minutes later, at 3:37 P.M. Salvation announced that the backup was complete. The last time I had backed up that hard drive, with Easy Drive, there were about 800 fewer files (about 7 Megabytes less) and it took 45 minutes to complete. (So, how does Salvation compare to ProSel 16? To be honest, I have never used the Backup feature of ProSel 16, so I can't give a credible answer to that question. However, I did demonstrate Salvation for my users' group and there was not a single user of ProSel 16 that did not express amazement at the speed of Salvation.)

Of course, I have backed up both volumes of my hard drive several times since that first backup. My hard drive is now almost completely full (60 Megabytes total) and each time, Salvation has finished the chore in less than 90 minutes. Based on all of the backups



that I have done with Salvation, I have figured that, on my system, Salvation backs up data at the rate of about 0.6 Megabytes a minute. Now, some of you may remember that I have a TransWarp GS on my system. I also have two systems without TransWarps and Salvation has performed at almost the same rate on those as well. The biggest effect on backup speed is the speed of your hard drive and not necessarily the speed of the computer itself.

But, it's not just raw speed that makes Salvation so fast. Salvation does lots of nice things that make the user (usually a big bottleneck in the backup process) more efficient as well. For example, when Salvation is done with a disk (either reading or writing), it ejects it for you. If you have more than one drive connected, you can put a disk in each drive and Salvation will automatically read from or write to the drives in order.

Another backup bottleneck that Salvation effectively eliminates is the unformatted disk. If you insert an unformatted disk in a drive, Salvation will format the disk in about 5 seconds (honest!) and then continue on with the backup. For other backup programs, dealing with unformatted disks can add tens of minutes to the total time required for the backup.

What other important things can I tell you about this program before I run out of space? Well, if you are in the middle of a backup and an emergency arises, you can pause the backup. When you get the time to finish, the Continue Backup option will let you resume the backup at the point it was paused.

The Bad Stuff

Well, after all of that gushing, I bet you thought I was going to just forget about the problems I found with Salvation didn't you? Not a chance.

The first problem that I have with Salvation is that it does not allow for Disk Image backups. While not as flexible as the File-By-File backups that Salvation does provide, Disk Image backups are usually much faster.

The second problem I have with Salvation is that it does not seem to support the LaserWriter for printing either backup reports or diskette labels. While this is not a problem most IIGS owners will care about, it is a rather glaring omission in a program that is otherwise a shining example of how to follow Apple's Human Interface guidelines.

Another slip in Salvation's user friendly interface is the lack of descriptive error

messages. Actually, they aren't even messages, they are just error numbers. For a program that goes so far out of its way to be user friendly, it is surprising to see it use cryptic error numbers that even a programmer like myself will probably have to look up in a book.

The last problem I found with Salvation is, well, it seems to crash a wee bit. At this point, I can't tell exactly why or how, but sometimes when I am trying to restore files, a dialog filled with "I"s will appear and the program will lock up.

More Good Stuff

Which brings me to the subject of Technical Support. When I called Vitesse to inquire about this problem, I was told that, yes there was a problem with the restoration feature of version 1.0 and that they would place a copy of the new version 1.01 in the mail tomorrow! (Actually, it turned out that the

problem I was having was completely unknown to them. It was hoped that version 1.01 would fix it, but I have seen the problem in the new version too. As soon as I can figure out how to duplicate the problem consistently, I will be contacting Vitesse's Tech Support again. Still, if you have version 1.0 be sure to call Vitesse to receive the version 1.01 update!) All of this took place BEFORE I had a chance to say anything like, "I'm doing a review of this product for a magazine." Normally, I don't EVER tell the Tech Support people that I speak with that I am doing a review. But after about 15 minutes of my standard "unreasonable customer" routine, I could tell that this was a genuinely knowledgeable and NICE person that deserved to know the truth. After I spilled the beans, he became very concerned, "You should do a review of the most recent version, don't you think? After all, we don't want your readers to think that this fix is just a bunch of VaporWare! Do you have a modem by any chance?" I said

that I did, and, just a bit later, I had a working copy of version 1.01 of Salvation... and they still sent me the update disk! While not everyone publishes a IIGS magazine, the courtesy, respect and honesty that I was treated with BEFORE I told them exactly what I was up to, has me convinced that Vitesse is a company that is committed to customer satisfaction. The only way that Vitesse could make it better would be to add a toll-free number for support.

The Wrap Up

Salvation is one of the best backup programs I have ever used, on any computer. Christopher Warner and Vitesse should both be very, very proud of this piece of software. At only \$49.95, it's probably the best money I've ever spent on IIGS software. The service isn't bad either. If you have a IIGS and a hard drive, you need to buy a copy of Salvation. That's all there is to it.

ORCA/Disassembler - retail \$49

Not Copy-Protected

Requires: 512K RAM (Text version)
800K RAM (Desktop version)

Program by Paul Elseth

The Byte Works, Inc.
4700 Irving Blvd. NW, Suite 207
Albuquerque, NM 87114
(505) 898-8183

Reviewed by Joe Wankerl

Have you ever wondered what makes the Finder tick? Ever ponder the inner workings of AppleWorks GS? If you have, the ORCA/Disassembler is just the program you need to satisfy your curiosity! This wonderful program should bring a smile to anyone who is interested in how other people write programs or anyone that loses the source code to a program of their own.

But just what is a disassembler? Ordinarily, a programmer types in source code and then a compiler or assembler takes the source code

and translates it into a form which the computer can understand. A disassembler is used to reverse this process by taking the final program and translating it (with a LOT of help from the user) back into a more human-readable assembly language source code file. This means that if a program was written initially in Pascal, then compiled, and then disassembled, the result would be an assembly language rendition of the original Pascal program. Disassembly is not easy, though. Much time will be spent wading through hexadecimal numbers, trying to figure out which sections are data and which are actually executable code, and just generally making lots of incorrect guesses. A good disassembler helps lessen the guesswork by providing tools to help you more easily recognize and disassemble both data and executable code. It's a lot of work, but it's definitely worth it; studying the code of others is an excellent way to learn more about programming.

The ORCA/Disassembler includes two versions of itself in one package. The first is an APW/ORCA shell (EXE) file and the second is a stand-alone desktop program. The

EXE version has all the options that the desktop version has, but it's slightly faster and a bit harder to use, as well. Since the ORCA/Disassembler manual spends most of its time discussing the desktop version, that is what we will look at in this review.

Bad News

The bulk of the badness of this program is in the manual, or should I say, lack of manual. It contains only one example of a disassembly, and that is a fairly trivial 'Hello World' program that takes almost no time to disassemble. Disassembling a regular IIGS desktop program is not a trivial exercise! It would be nice if the documentation had more examples and gave more space to some of the "tricks of the trade" of disassembly.

The manual also fails to mention a very confusing situation where disassembled code can sometimes disappear. When you're in the middle of a disassembly, code further on in the program will not be correctly aligned unless you've properly disassembled all of the code that comes before it. If you do not disassemble your code from "top to bottom",

code that you disassembled further on in the program can disappear right before your eyes. It's still there, you just can't see it. This can be an incredibly confusing situation! One moment you are staring at a neatly disassembled block of code, the next, all of that work appears to vanish. However, once you realize what is going on, it's a bit easier to deal with.

But Now For the Good stuff!

Again, let's start with the manual. It might seem contradictory that the manual can be both good and bad, but for the most part, the manual is very clearly written and explains most of the program's features very well. The program itself is written so that you don't usually have to have the manual handy; everything is very intuitive. Again, the only thing missing from the manual is an adequate number of examples.

Another strong point of the ORCA/Disassembler is that it is VERY bug free. That's saying a bunch for a first release of a program this complex. Most other version 1.0 programs come with lots of free bugs that you usually have to pay to get rid of in later versions. The ORCA/Disassembler has only a few minor bugs, none of which are serious enough to hamper the operation or usefulness of the program.

The ORCA/Disassembler is amazingly fast! The two slowest things you will encounter in the ORCA/Disassembler are loading the program and using the Generate Labels option (which assigns labels to memory references). Generally, you use Generate Labels only once per disassembly so this is not a very serious problem.

Another plus is that you can disassemble ANY file that you can coerce into a BIN file. This can be very handy for looking at and saving, the hex representations of data files. However, at this point, the ORCA/Disassembler can not disassemble Resource Forks at all.

Another strength of this program is the way it stores its "dictionary" of IIGS Toolbox calls. All of this information is stored in a plain text file that follows the standard established by Dave Lyon's popular ShareWare program, Nifty List. This allows you to add new tool calls to the file simply by modifying it with any text editor. No need to wait for an update from Byte Works! This setup also allows you to correct any mistakes that you might come across. For example, in the process of disassembling a game that makes extensive use of the Audio Compression & Expansion Tool set, I noticed that the _ACEBootInit tool call kept showing up in the disassembly. This is a

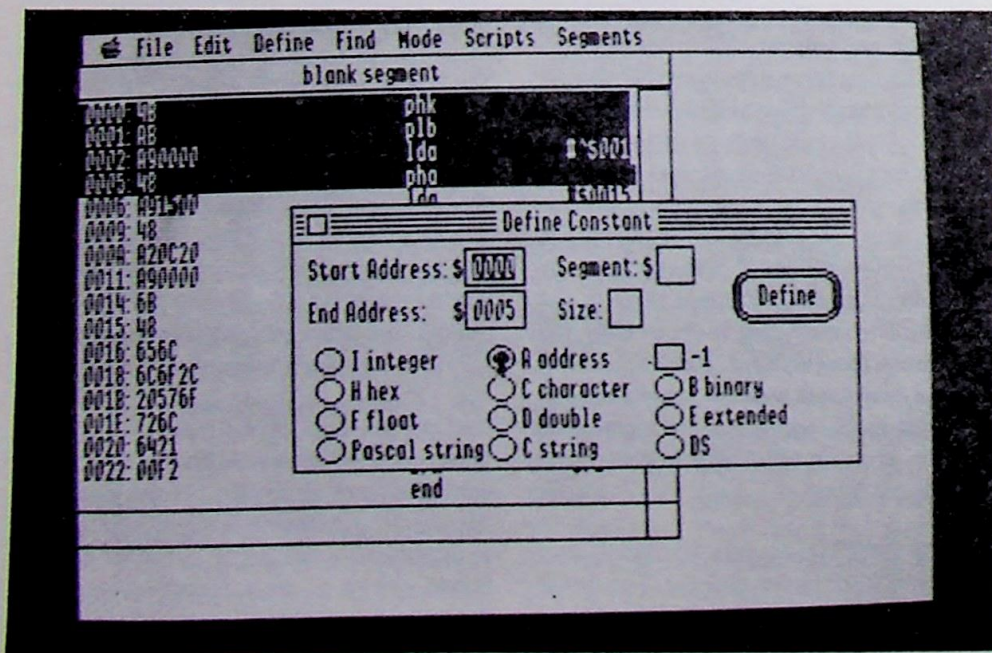
call that an application should NEVER make! By looking at the dictionary file, I could see that ACEBootInit had been defined twice and ACEShutDown was nowhere to be found! So, I simply changed the second ACEBootInit to ACEShutDown and everything worked fine from then on.

The ORCA/Disassembler also allows you to automate the disassembly process through the use of scripts. The ORCA/Disassembler comes complete with a large number of scripts that will automatically disassemble many common IIGS programming structures, such as CDA/NDA headers, window parameter blocks, GS/OS parameter blocks, etc. If none of the included scripts does the job, you can build your own script using the built-in script editor. There is even a script trace facility that allows you to debug your scripts by single-stepping through them.

Each disassembly is saved in a template. A template is a history of all the changes you've made to the program you're disassembling. This allows you to save your work at different stages in the disassembly so that if you make a mistake you can always revert to a previous template you've saved and start again from there. When you're done with the disassembly, you can output the disassembled program to an assembly language source file. You can then modify or assemble the code to your heart's content.

And then there are the actual tools for disassembling code. You can define and remove labels, define 11 types of constant storage (integers, Pascal strings, real numbers, etc.), you can create new direct page areas for a disassembly, add relocation records for non-OMF files, the list goes on and on. When you add to this the fact that almost all of these features can be automated in a script, you have a very powerful utility at your disposal.

All in all, the ORCA/Disassembler is a wonderful program. However, to get the most out of it, you need to have a firm grasp on IIGS programming in general and assembly language in particular. If you qualify, or if you just want learn assembly language from the "inside out", this is an excellent program that definitely belongs in your library.



Computer Eyes - retail \$250

Program by Daniel Lovy

Digital Vision, Inc.
66 Eastern Avenue
Dedham, MA 02026
(617) 329-5400

Reviewed by Steven W. Disbrow

No matter how good a computer artist you are, reality is always just a little better. Wouldn't it be nice to be able to capture just a bit of nature's handiwork for your own computer creations? That's just what Computer Eyes allows you to do. Computer Eyes is a video digitizer that plugs into one of your IIGS expansion slots (slot 4 is recommended) and allows you to capture images from any piece of standard video equipment (TV, VCR, Camcorder, etc.).

It's FUN!

As you can tell from the photo accompanying this review, the files on the GS+ disk and the, er, cover of the magazine, Computer Eyes is capable of producing, or helping to produce, some very striking images. Computer Eyes will also add quite a bit of life to any party that you might throw. All it takes is a little practice and a LOT of patience.

The Computer Eyes hardware consists of a single expansion card and a set of three shielded cables that hook the card into your video equipment. I found the cables to be just about the right length for my desk setup, but just in case they are not right for you, Digital Vision also includes an extension cable that you can use to make one of the cables go an extra couple of feet. The installation process is straight forward and the manual does a good job of explaining exactly what needs to be done. If you read the manual first, the installation process should take about 10 minutes.

Once you have the hardware installed, the next step is to start up the Computer Eyes software and try to find something worth digitizing. The Computer Eyes software is not copy-protected and installs easily on a

hard disk. When the software finishes loading for the first time, you tell it which slot that you put the digitizer card in, and you are ready to go.

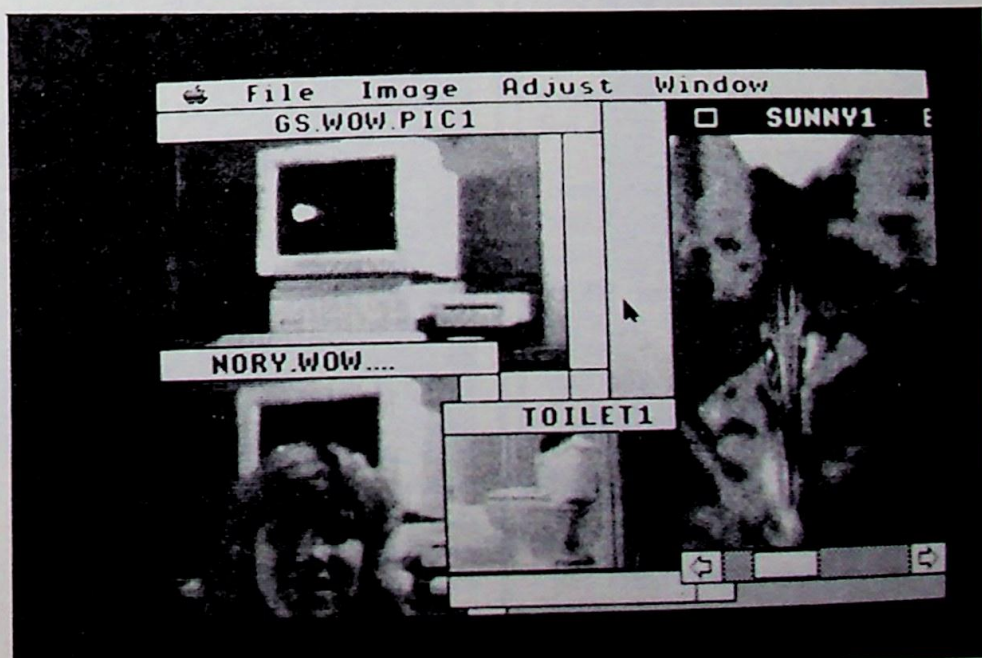
The Computer Eyes software is very easy to use. To capture an image, you simply select Capture Color or Capture B/W from the Image menu and, many seconds later, the image appears in a window on the screen. Once you have an image captured, you can use one of the several tools in Computer Eyes itself to modify the appearance of the picture. The Adjust menu allows you to change the color palette and the contrast of the picture. You can also create a mirror image of the picture, invert all the colors in the picture and shrink or expand it. All of these capabilities allow a great deal of control over the final appearance of the image. However, if you want to do something a bit more creative, like turning your publisher into a rabbit, you will need to use a dedicated paint program like DeluxePaint II or PaintWorks Gold.

But Do You Have the Time?

While it's true that Computer Eyes is a very capable digitizer, it is also incredibly slow. On the fastest capture speed, it takes a full 6 seconds to capture a single picture. On the slow capture speed, it takes 12 seconds to

capture an image. If you are capturing a color image, it takes another 4 seconds or so to calculate the palette, and the results are not exactly what I would call spectacular. Fortunately, it is possible to get some fairly decent color images if you load the picture into a paint program and use its tools to manipulate the palette.

If you have a TransWarp GS, you might be thinking that you won't have to worry so much about the slow speed of Computer Eyes. Unfortunately, this is not the case. The Computer Eyes software is incompatible with the TransWarp GS. If you have a TransWarp GS installed and you run Computer Eyes, The Computer Eyes software will reset the system speed to "Normal", 1 MHz! This makes for unbearably slow operation and makes a Computer Eyes/TransWarp GS combination worthless. I called Digital Visions' Technical Support about this problem and was told that I should set the TransWarp GS to run at 2.6 MHz before running the Computer Eyes software. Alas, this did not solve the problem. Computer Eyes still insisted on setting my system speed to 1 MHz. The only solutions I could think of were to pull the TransWarp GS (Ha!) or install Computer Eyes in another, non-TransWarped, IIGS. Since I had just purchased another IIGS, this is what I did. On the non-TransWarped IIGS,

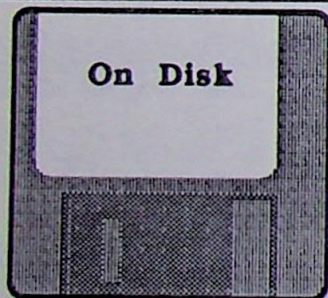


Computer Eyes behaves fine. However, this is definitely not a solution that will be open to most IIGS owners. In my opinion, this is a VERY serious problem that should NEVER have made it past the beta test stage.

The Computer Eyes software also suffers from several other, less severe, problems. The most annoying of these problems is that since the software only runs in 320x200 mode, that is the only resolution at which you can digitize a picture. It is possible to SAVE pictures in other formats (640x200, Print Shop GS, etc.) but the resulting pictures are usually quite horrible.

In the Final View...

Computer Eyes for the IIGS is a very good, but exceedingly SLOW, video digitizer that you can use to capture some really dynamite black and white pictures. If you do not have a TransWarp GS, Computer Eyes is a solid performer that you should take a close look at. Speaking of the TransWarp GS compatibility problem, I called Digital Visions' Tech Support just before we went to press, and I was told that there are no plans to provide a fix. Heck, they don't even have a TransWarp GS to test it with! If you do have a TransWarp GS and you are considering purchasing Computer Eyes, I would suggest that you consider something else.



On the GS+ disk, in the folder CompEyesRev you will find the following pictures:

Creative.Team	Diz.Bunny.Idea
Diz.What.The	Motley.Crew
Pliable.Nory	Sticky.Fingers

These pictures can be viewed with any IIGS paint program. These pictures were captured during this review of Computer Eyes.

Jam Session - retail \$50

Not Copy-Protected - Requires 768K RAM
IIGS Version by Neil Cormia

Broderbund Software, Inc.
17 Paul Drive
San Rafael, CA 94903-2101
(800) 527-6263 or (415) 492-3500

Reviewed by Jeffrey H. Walker

Jam Session is a new music program from Broderbund that requires only a little rhythm to play music like a professional. There are a number of songs on the disk in various styles (21 in all), with various bands, from classical to country to heavy metal. Once you pick a song, the band and instruments are loaded, and you are ready to go! Select Play from the menu, or press Open-Apple-P, and the band will start playing a background rhythm and bass line that you can play lead (or rhythm) to by pressing keys. Each of the four rows of keys on your keyboard controls your instrument, and each key evokes a different riff (sequence of notes in a specific key). The riffs have variations in speed, complexity, and duration depending upon which row of keys you select your riff from.

Making Your Own Music

I myself am an amateur musician, and I went NUTS with this program. But, after a few days, I was starting to get a bad case of *stupid fingers* (Oh, you know, that malady that makes your fingers get locked up and stiff, so they don't go where you want them). During the course of a performance, when the song changes key, the riffs associated with your instrument keys change to reflect the change in key of the song. On some of the selections you can get different instruments by pressing the shift key along with the riff key (this feature can be customized using the Edit Riff function). If you have shaky rhythm, there is an option to Sync To Measure that will play your riff in perfect rhythm (the keys are held in a buffer and only used at the top of a measure, so there are no overlapping riffs). I prefer to have this feature deactivated so I can press the keys in quarter time to produce wild effects.

Song length is adjustable in multiples of the original song length and song tempo is adjustable in value from 10 (the slowest) to 450 (the fastest). The default speed for each song sounds just about right. There is a menu item under the Options menu called Atmosphere that allows you to turn on/off a round of applause at the beginning and end of the song. There are provisions for recording and playing back songs to delight and amaze your friends.

There are simple editing capabilities included (under the Edit menu) whereby you can modify any of the existing riffs. This mode is also useful for musicians to learn any of the cool licks that are in the program. (There are some good ones!) In the edit mode there are four tools that you have at your disposal: Eraser, Quarter Note, Note Bender, and Rest. To insert a note, you select the quarter note and click where you want the note on the staff. If you want a note of longer or shorter duration than a quarter note, hold down the mouse button and drag left for shorter, or right for longer. The Rest tool works in basically the same way. The note bender tool allows you to adjust an existing note in pitch or duration, and of course the eraser allows you to erase any notes you might want removed. If you are satisfied with your edited riff, you can edit another by pressing the key of the riff that you next want to modify, and up it comes in the window ready for editing. In the country music setting, there are keys on the bottom of the keyboard that make Chicken sounds, and just for a joke, I edited one of them to squawk "The Ride of the Valkyries."

Is This A Rubber Band, Or What?

The graphics in Jam Session are outstanding! The musicians move their hands to the beat and riffs, and in the Heavy Metal background, whenever the guitarist runs his pick down the strings, you are dazzled by a lightning bolt from the sky! In the Jazz settings, while you are playing the piano and horn, your beer mug fills and empties, and then refills itself. Heads move in the audience, cigarette smoke fills the room, candles flicker and cast the audiences faces in the gloom, and the feeling creeps up on you that you are in a "you don't get in, unless you're one of the ten" Jazz bar

on the east side. In the country music setting, the chickens stretch their necks when they squawk and peck, and the musicians tap their feet. But, my personal favorite is the blonde female keyboardist in the Jamtana setting; she's dressed up in a leather miniskirt and fishnet stockings.

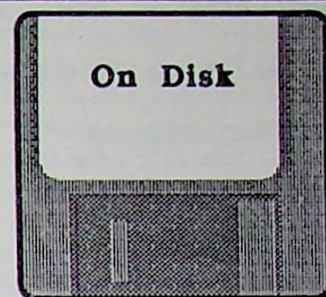
The Accidental Musician

The sound quality is very pleasing, the music itself doesn't have that unsavory "ELECTRONIC" sound, and this is accomplished by using accidentals (those little mistakes of timing that all musicians make, but the best are able to cover it up so it sounds like they meant to do it.) The effect was so good sometimes, I could almost smell the stale cigarette smoke and cheap beer. I recorded the rhythm track from Jamtana on my 4 track for a background to play my guitar to, and it sounded great. I ran the program on a IIGS equipped with Applied

Visions' Future Sound stereo card and a Sony "Boom Box." This helped to give the music that extra realism kick that really made the program for me.

Jam Session is not copy-protected and ran on both the old (ROM 01) and new (ROM 03) IIGS systems with no problems. The program comes on two floppies and easily installs on a hard drive. The manual is sufficient, and will answer any questions you may have.

I recommend this program to anyone who dreams of being a musician, or just wants a very good show-off program to demonstrate the sound and graphics capabilities of the IIGS. It's a very entertaining program that anyone can enjoy. Besides the fact that the program is flashy and well done, it's FUN! I personally think that, for pure kicks, it beats all of the other music programs on the market. Good work Broderbund!



On your GS+ Disk, in the folder **JamSessionRev** you will find the following Jam Session song files:

Jam.Song.1	Jam.Song.2
Jam.Song.3	Jam.Song.4

These songs were created by Jeff Walker as part of his review of Jam Session. You must have Jam Session in order to play these songs.

Product Updates

TransWarp GS Update

Applied Engineering's TransWarp GS is now shipping with a cable that allows you to plug it into either slot 3 or slot 4. If you have an earlier TWGS that only goes in slot 3, call AE Tech Support at (214) 241-6069, tell them about it, and in about a week you should have a replacement cable. You don't get any instructions, but hey, what do you want? This thing is absolutely free!

Graphic Disk Labeler (GDL)

Triad Venture, Inc. is now shipping an update to GDL that adds lots of new features and addresses quite a few of the problems mentioned in our review (volume 1, number 3, page 25). The retail price of the new version is being increased to \$39.95, but, owners of previous versions can upgrade for only \$9.95. For more information, contact Triad Venture, Inc. at (516) 360-0797 or (513) 732-3771.

Ancient Land of Ys - retail \$44.95

Copy-Protected - Requires 512K of RAM

IIGS Version by Dave Galloway

Kyodai Software Marketing, Inc.
58 Mitchell Blvd. Suite C-14
San Rafael, CA 94903-2101
(415) 492-3592

Reviewed by Jeffrey H. Walker

A radiation-burned, bleary-eyed subhuman sits in front of a flickering computer screen. A little man moves swiftly across the screen deftly avoiding obstacles, but his movement catches the attention of a local meany that looks a lot like a horse-apple with TEETH! (Oooo . . . He's a nasty one, he is!) After a little good-natured gnawing by the monster, the little man on the screen is acting a bit shagged out, actually he looks kinda... DEAD! AGAIN! AHhhhhhhh! Dam.

The Ancient Land of Ys is a top-view adventure game along the lines of that Nintendo favorite, The Legend of Zelda. The kingdom is being harassed by evil thieves, stealing all the kureria (silver metal) in the land and hiding it. Your job, should you take it, is to find all six of the ancient books of Ys, hidden by the evil sorcerer, Malificus,

and wake the goddesses from their long sleep, to rid this fair land of the evildoer. (You also get filthy rich and collect all sorts of goodies along the way). Five of the books are guarded by nasties and the sixth book is held by Malificus himself. During your travels you will talk to various townspeople, priests and (ahem) trees, to get clues and objects you will need for the completion of the game. There is also a time limit! I played through the game once to the end and BEHOLD! The priestesses were GONE! WHAT A BUMMER! Sooo . . . I played through again in about two days and finished it.

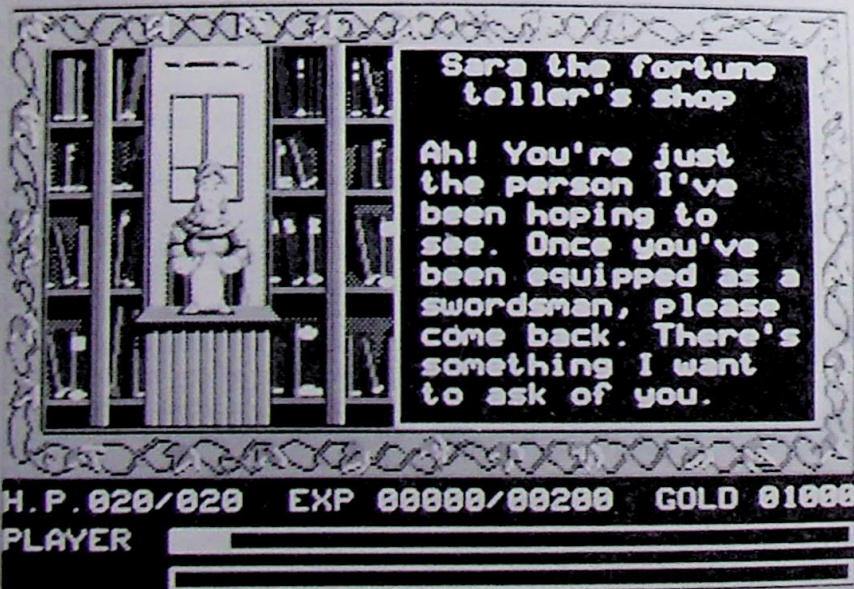
The monsters are laid out in a logical fashion, such that you usually don't encounter higher-level monsters until you gain a suitable experience level (just stay out of the lower levels of the tunnels when you are still wimpy). The preferred fighting technique is to get behind (or at least not in front of) the monster, and hack away (by moving into the beastie's space)! Some monsters will eat you up if you get in front of them (the twisted pacman on steroids is a prime example). The first step towards finishing this game is getting money to buy weapons and armor (including a shield). To do this, you leave the city and trounce a few wandering knights and a great many very rich wild animals to get the required amount of gold. There is an armor and weapons shop in

the city, as well as a doctor's office where you can get medicated. Be sure to talk to all the townspeople you can find (lots of subplots and clues here). And, finally, if you seem not to be able to do something, wait until you rise another experience level, and then try again.

There are many magical items to find and be given. Rings to heal, protect, and harm you, masks to peer through to see hidden doors, swords to hack with, plus armor and shields to protect you. There are healing potions, wings (to instantly return you to the city), and mirrors (to freeze the monsters on the screen, not infinitely reusable). Most of the things that you find in the game will be in chests, some of which require keys to be opened (keys to open locked chests are found in unlocked chests).

The graphics in this game are quite good, although indistinguishable from their IBM EGA counterparts. The environment including trees, huts, and the river are well done. The monsters look very businesslike (if eating visitors is your business) and are aptly animated. There is one monster that is all gnashing teeth and mouth. The mazes are twisty-turny, and you can only see a set distance (a nice effect), so don't go running around haphazardly (this is a good way to meet a bad guy). In the temple and the tunnels, make sure that you look at all the walls, especially the ones toward the bottom of the screen, before giving up on finding doors and passages (use the mask if you seem stuck). The scrolling is smooth and movement is acceptable although much better with a TransWarp GS.

The game has provisions for keyboard as well as joystick control, and if joystick is selected (control-J) the game will allow you to center the stick before play is resumed. Button 1 on the joystick is a quick substitute for the 'I' key, and takes you to the Inventory screen. From this screen you can use the joystick to select the armor, weapons and other items that you wish to equip your character with. The keyboard controls seem to be heavily buffered and are very frustrating to use. If at all possible, I suggest that you use a joystick. You will be a lot happier. The other game controls (load game, save game,



sound controls, etc.) are control-key combinations, but be careful because the Load Saved Game command is very close to the Quit command (a terrible way to waste some time)!

The music in this game ranges from monotonous to catchy. The instructions specifically state that there are 3 separate settings for the sound:

- (1) Music on, sound effects on.
- (2) Music off, effects on.
- (3) Both off.

But, in fact, as stated in the addendum card, there were only two settings: BOTH on, or BOTH off! The droning city music lost its fun sparkle for me after about 5 minutes, so I turned it off. But without ANY sound you can't tell when you've actually been damaged by a hit (without staring at your Life indicator), but this is a chance that I'll take. The dungeon and tunnel music was better, which is a good thing, since the monsters are much meaner in these areas and you need to be able to HEAR the combat.

The Ancient Land of Ys is compatible with both the old Apple IIGS (ROM 01) and the new Apple IIGS (ROM 03) running System Software 5.0.2. Copy-protected? You bet, and this is the only serious gripe I have with this game. After 5 hours of disk swapping you are tempted to let the bad guys win. I commandeered two floppy drives to finish the game on the new machine, and it was much less bothersome.

Even though Ancient Land of Ys isn't as "deep" as something like The Bard's Tale, it is fast-paced and just as satisfying. Besides, the monsters are kinda cute, in an ugly sort of way. If you like adventure games, I highly recommend this one.

Tunnels of Armageddon

Retail \$39.95

No On-Disk Copy Protection
Requires 512K of RAM

Program by Jarek Achinger, Adam Skorupinski, Mirek Zablocki & Krzysiek Koziarski

California Dreams
780 Montague Expwy., #403
San Jose, CA 95131
(408) 435-1445
FAX: (408) 435-7355

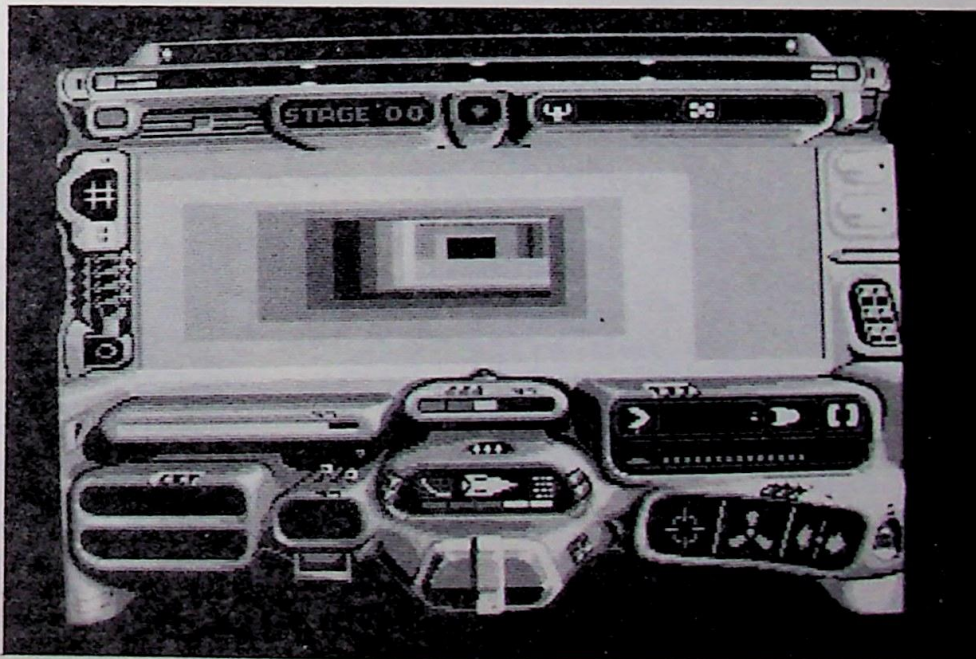
Reviewed By Steven W. Disbrow

Tunnels of Armageddon is a new arcade game from California Dreams that gives new meaning to "on the edge of your seat." In this game, you are the pilot of the Hermes hovercraft, and your mission is (among other things) to save the earth from total destruction. I won't bore you with the rest of the rather contrived plot other than to say that if you've seen *2001: A Space Odyssey*, it might bring a smile to your lips. The game itself will definitely bring a smile to your lips, along with a few unprintable words.

In Tunnels of Armageddon, you must pilot the Hermes to the core of the Earth through

twenty levels of twisting, turning, nausea-inducing tunnels. Once you reach the Earth's center, you must find and eliminate the doomsday device that threatens to destroy the planet. This is not your everyday, leisurely jaunt to the center of the earth you know. To successfully complete the mission you have to race against a detonation signal that is activated when you first enter the complex of tunnels. You can slow down the signal by shooting various signal boosters (called Energizers) situated along the walls of the tunnels. (Of course, all of this implies that the Hermes flies faster than the speed of light and that the speed of light is not constant. . . but if you think too much about petty details like that, you end up flying into a wall and you lose LOTS of time!) But, the best way to beat the signal is to fly at top speed and STAY OFF THE WALLS!

The tunnels that give Tunnels of Armageddon its name are exactly that: big, twisting tunnels that you fly down head first and see from an excellent first person view. There are dips and turns, splits and bumps and a large number of sliding walls that always seem to pop up just where you don't want them. There are also various power-up items that help you to navigate the tunnels more easily and a large supply of robot guardians that try their best to blow you to bits. There are three types of tunnels: grey, green



and red. The grey tunnels are the easiest to navigate, but they are also the longest. The green tunnels are shorter than the grey, but they have more surprises and are a bit harder to fly through without running into something. The red tunnels are the shortest of all, but they hold the most power-up items along with a deadly number of robot guardians.

Tunnels of Armageddon is great fun to play. The animation is INCREDIBLY fast and smooth and truly gives the feeling that you are flying head first down a very, very long tunnel. There were quite a few times I found myself quite literally falling out of my chair as I used "body english" to try and coax the Hermes off of a wall! To accomplish this magnificent animation, Tunnels of Armageddon employs a seldom used IIGS graphics mode known as Fill Mode Graphics. Fill Mode Graphics don't allow for a tremendous amount of detail, but they do allow you to produce animations that are so fast, the video hardware is hard pressed to keep up. In fact, Fill Mode is so fast, there is NO visible difference when running the game either with or without a TransWarp GS. The only way to tell is that the game takes longer to load without a TransWarp GS.

Tunnels of Armageddon also features a very good sound track. Even though they are only mono, the sounds in Tunnels of Armageddon do a great job of getting your blood pressure up, and telling you when you get hit by a robot or become a bit too friendly with a wall. If you don't like the music, you can toggle it off leaving only the sounds that clue you in on the important stuff (getting shot, hitting a wall, etc.) If you don't want to hear any of it, or you want it louder, you can adjust the volume by using the '+' and '-' keys. The manual recommends that you play the game with a set of headphones on. I recommend a boom box and lots of Krystal (or White Castle) hamburgers for the full gut-wrenching effect.

Tunnels of Armageddon runs easily from a hard drive and is completely compatible with System 5.0.2 and the new ROM 03 IIGS. The top ten scores are saved for all tunnel runners to see and you can save your game at

the end of every four levels. This is a handy feature due to the fact that, more often than not, the Hermes often ends up as a wall hanging.

Another nice feature of Tunnels of Armageddon is that after you complete all twenty levels, the game begins to generate random tunnel configurations. So, you never have to worry about the game becoming stale. This is a very addictive game, so this is a very nice feature to have. (Actually, I have not quite made it all the way through... so I have to take the manual's word on this one.)

For all its incredible graphics and wonderful game play, there are still a couple of problems with Tunnels of Armageddon. The first is fact that the only way to control the Hermes is with the mouse. While this is generally a dandy way to control a hovercraft, zipping around inside tunnels like this requires that you pick the mouse up, move it over to some empty area on your desk and slam it down again quickly in order to have enough room to maneuver. Joystick control, even as an option, would probably be a better way to do things.

The last problem with Tunnels of Armageddon is that it employs a copy protection wheel. While the disk itself is not copy protected and you can easily make a backup or install the game on your hard disk, it can be a bit annoying having to look up the appropriate color on the protection wheel. Don't get me wrong. Off disk copy protection like this is infinitely preferable to the copy protection employed on other games (Dungeon Master for example) but it would still be nice if companies trusted computer users enough to not copy protect software at all. Of course, trust must be earned, and software piracy is still a problem in the IIGS community.

So, if you like a game that gets your blood pumping, makes you alienate your loved ones and keeps your eyes fogged over from staring at the screen for hours at a time, Tunnels of Armageddon is a must have. It is an impressive "gee-whiz" type demonstration program and an incredibly fun and playable game to boot. Very highly recommended.

Contest 3...

Remember, the deadline for Contest 3, is **April 15th, 1990.**

In case you missed it, all you have to do to win Contest 3 is to send us the best HyperStudio Stack you can come up with. If you win, you can pick from one of the following fabulous prizes:

IIGS Toolbox Reference: Vol. I, II or III
Exploring GS/OS and ProDOS 8
Xenocide
Graphic Disk Labeler
A One Year Subscription To GS+

The Rules:

- 1) Send your entry on diskette to:
GS+ Contest 3
P.O. Box 15366
Chattanooga, TN 37415-0366
- 2) Or, pack your entry with ShrinkIt and send it to 'Obnoxio' via America Online.
- 3) Entries must be received no later than April 15, 1990
- 4) Be sure to indicate your prize choice and a phone number we can reach you at.
- 5) Only two prizes will be awarded, but Honorable Mentions will get their names and entries published along with the winner and runner-up.
- 6) Stacks submitted should not be password protected.
- 8) Naughty stacks won't qualify for a prize, but we really would like to see some just the same.
- 7) We welcome your ideas for new contests.

So, get busy and send those entries in today!

CLASSICS

Where in the World is Carmen Sandiego?

Retail: \$44.95
\$54.95 (School Editions)
\$109.95 (Lab Packs)

Copy protected - Requires 512K of RAM
IIGS Version by Loring Vogel

Broderbund Software, Inc.
17 Paul Drive
San Rafael, CA 94903-2101
(800) 527-6263 or (415) 492-3500

Reviewed by Noreen Ribaric

Broderbund's "Carmen Sandiego" line has been popular on the 8-bit Apple II's for years. They combine learning with the best elements of adventure, mystery, and arcade-style games. Now, the first of these has been released specifically for the Apple IIGS. It boasts enhanced graphics and sound, and, of course, is faster.

The game will load on a hard drive, but will check for the original disk when loading the game. The manual states that running the game on a hard drive requires a 3.2 system disk (ProDOS 16 - which they include), but I ran it under system 5.0.2 with no problems, except occasionally the program had a problem finding the "detective file," but quitting and resuming the game usually fixed this (you might lose a few cases off your record if you continue playing after seeing this message).

After loading the game, an animated title sequence will repeat until a key is pressed, or the mouse clicked, at which point the game will begin. You can begin the game as soon as you hear the music, but I think you should watch the introduction at least once because the animation is very good (much better than the IIe/IIc version). There is also an animated sequence when you capture the villain, which can be shortened by pressing a key or clicking the mouse when the JAIL appears.

The game is almost entirely mouse- and menu-driven. Except for entering your name

at the start of each session, you don't have to use the keyboard at all to play this game. But, if you enjoy using the keyboard, you may press the Return key whenever you see "Please click here to continue," and you may type 'Y' for Yes or 'N' for No instead of clicking on the appropriate button with the mouse. However, you cannot play the entire game with the keyboard, you must use the mouse for everything else.

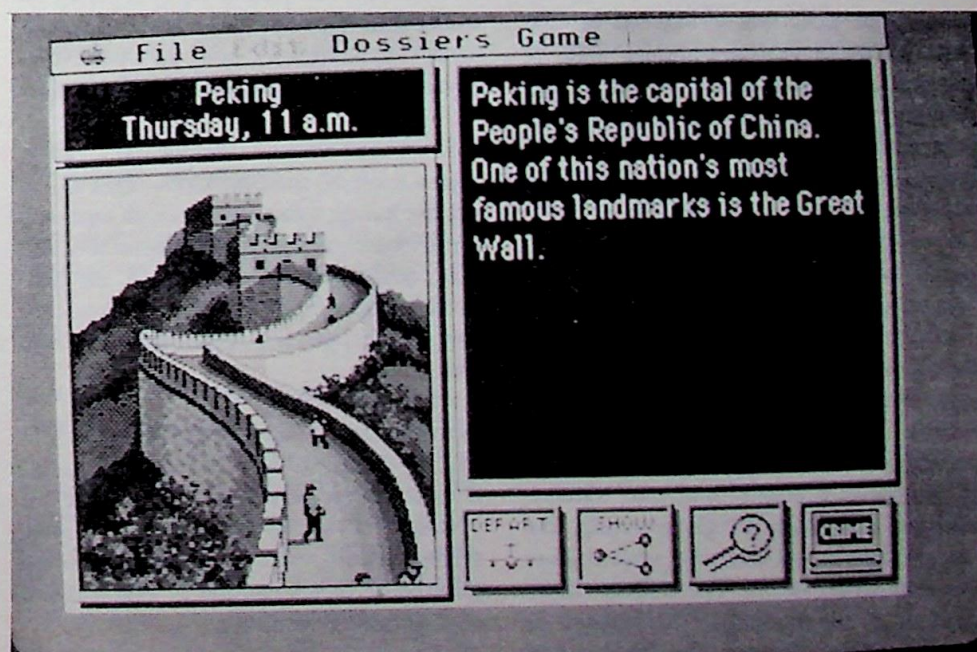
The game begins with you signing in as a Rookie detective at the ACME Detective Agency. Your first assignment is to track down Carmen Sandiego and her Villains' International League of Evil (V.I.L.E.). This won't be an easy task - there are 10 possible suspects and 30 cities around the world that are known V.I.L.E. operational bases that could be potential hideouts for the thief.

You begin each case at the scene of the crime and must track the villain from city to city, unearthing clues along the way. As your investigative skills improve you will earn promotions. After solving one case you are promoted to Sleuth, eventually becoming an Ace Detective. After solving fourteen cases you are entered in the Hall of Fame and cannot continue unless you use an "alias" (because you may be on a hit list

somewhere)! The cases get progressively harder with each one that you solve. You always get six full days to track down the villain, but you start out having to track the villain through five cities per case, after six cases you must travel through six cities, eventually having to track the villain through seven cities when you become an Ace Detective. You also get fewer clues as your rank progresses.

The game comes with the current edition of the *World Almanac And Book Of Facts*® to be used to help decipher clues you find along the way. There are over 1,000 clues in all, including the names of rivers, mountains, and industries. But the ones I found to be the most valuable in determining where the villain had fled were the flag descriptions. This is because the almanac has color pictures of the flags for almost every country in the world! But this can sometimes make catching the villain too easy! To make the game more challenging, you can try to figure out the clues without the aid of the almanac.

Upon arriving at each city you will be presented with a picture of a well-known landmark. The graphics are very impressive and a big improvement from the IIe/IIc version. You will also be given information



on the local geography and culture. The information is not always the same every time you travel to that city, so you have the opportunity to learn quite a bit about each location.

At each location you may choose between four options represented by icons: 'Depart' shows a map of the world with your present location and the three or four cities that can be reached by connecting flights. You can go to one of these cities by clicking on them. 'Show' also displays the three or four cities that can be reached by connecting flights from your present location and you can depart to one of these cities by double-clicking on it. The difference between this and the 'Depart' option is that you can have the connecting cities continuously displayed as you are investigating, so after each clue, you can look over at the list to see if any of the cities match the clues. After clicking on 'Show,' the icon changes to 'Hide' and, of course, clicking on 'Hide' will remove the list of cities from the screen and change the icon to 'Show.' Clicking on the magnifying glass icon will display a window with three specific locations within the city for you to investigate (hotel, bank, library, etc.). Clicking on the Crime computer icon will allow you to enter the clues you collected about the suspect into Interpol's Crime computer. You can then "push" the 'Compute' button to get a list of the suspects that fit the description entered in the computer. When you have enough clues to narrow the suspects down to one, it will issue you a warrant. You must have a warrant when you finally track down the villain. If you don't, he (or she) will get away!

Usually only two clues, in addition to the sex of the villain, are needed to get a warrant, but some villains may need three (for instance, Merely LaRoc and Carmen Sandiego are both female, have brown hair, and wear fancy jewelry). If you only found two "real" clues (that can be entered on the Interpol computer) and it doesn't single out one villain, sometimes other clues are given to help, like favorite foods or whether the suspect likes dangerous sports or not. These facts can be looked up in the dossiers of the suspects in question to single one out. Then pick one of

the "real" clues for that suspect from the dossier that you haven't entered on the computer yet and enter it. If the "other" clue is not in the dossier, remember it anyway, and when you find out who the suspect is, make a note of it for future reference!

Spend your time wisely. Remember, you only have 6 days (144 hours) to track down the villain. Investigate in as few locations as necessary. The first search in each city takes two hours; the second, three; and the third and on take four. Don't travel to another city unless you are almost positive that is where the villain has fled, because each flight takes three hours (although I wish it really were possible to fly from New York to Moscow in three hours)! The first activity you complete after 9 P.M. will cause you to sleep for nine hours (detectives need their sleep)! Also, don't press the "compute" button on the computer at Interpol for each individual clue you turn up about the suspect. Instead, just enter them (that takes no time), but don't press "compute" until you have enough to single out the suspect (usually three clues) because each "compute" takes three hours. You can return to your search by clicking on any icon below the computer.

Clicking on SHOW or HIDE takes no time and clicking on Cancel will abort any activity before any time is spent on it. Performing things from the menu also take no time. You can view the dossiers (open-apple-d), the hall of fame (Open-Apple-H), or the detective roster (Open-Apple-D) using the Open-Apple key as well as selecting them from the menu.

Other menu options include New game (Open-Apple-N) and Quit (Open-Apple-Q). If you are having a particularly tough time on a certain case, or you goof-up miserably and know you won't solve it in time, the New game option will immediately start you on a new case, but you will remain at your current rank.

When you see "Ready for your next case...?" (Y/N)," choosing "N" will restart the game, allowing you to play under a new "name"; if you said "N" by mistake (hit return without typing anything), you can continue under the same name; or you can quit by then selecting Quit from the File menu.

This is a game that can be fun for kids and adults... and you might even learn something! I spent many hours tracking the Carmen gang down (under several aliases, I might add, because even when you finally solve all fourteen cases, there are so many possible combinations, it's a different game every time you play)! I highly recommend this game. If you already have the IIe/IIc version, you can upgrade to the IIGS version for only \$7.50 by calling Broderbund at the number above.

Where is Carmen Sandiego? Not on the IIGS!

We spoke with Marshall Davis, a representative from Broderbund, about the rumor that Broderbund will be dropping all IIGS development and we have some good news and some bad news. The good news: there are several products in the works for the IIGS that will be released later this year. These are listed below. The dates given are tentative release dates.

Where in the U.S.A. is Carmen Sandiego?	3/21/90
Where in Europe is Carmen Sandiego?	4/21/90
PrintShop GS Companion	TBA
Shufflepuck Cafe	3/21/90

The bad news is that the rumor is true. These will probably be the last products they develop for the IIGS because their current IIGS sales do not even cover their developmental costs for those products. So we probably will not see a IIGS version of Where in Time is Carmen Sandiego. If this upsets you, give them a call! If enough people care, and buy the IIGS versions, maybe they will continue to develop products specifically for the IIGS. In any case, they will still develop for the 8-bit Apple II line, and I'm sure continue to produce quality software like the "Carmen Sandiego" line.

THE MOLEHILL

By Joe Wankerl

The review of The Programmer's Online Companion in the last issue of *GS+* had an interesting statement that, according to Steve, I twisted all out of proportion. He said that if I was going to be making "mountains out of molehills", I should do it in a regular column. So, I thought about it a bit and decided that this would be a great way to help people who have problems with their software or hardware and want to know how to get help. And what better way to start things off than to solve the problem mentioned in the Programmer's Online Companion review?

The Programmer's Online Companion review said, "This is a real time-saver that not only works with APW, ORCA/M, and the ORCA/Desktop, but also with any editor that correctly supports the Clipboard. (AppleWorks GS, I have just discovered, does not fall in this category.)" Well I decided to find out why. Sure, nobody is going to write a program with AppleWorks GS, but still, if AppleWorks GS isn't properly supporting the clipboard, then, to me, that's a problem. So I decided to draw upon my biggest resource of knowledge: *network mail*.

As a student here at the lovely University of Tennessee at Chattanooga, I have access to one of the world's biggest wells of knowledge: BITNET. BITNET is a computer network which forms connections between many colleges around the world. It is an acronym for "Because It's Time Network", or, as most people say it, "Because It's There". There is a gateway between BITNET and Internet, the worlds largest business network. On Internet, there is an Apple II Computer forum where people can ask questions and get answers on all aspects of their Apple II computer. This forum is also distributed to BITNET and is called Info-Apple. Since Info-Apple is actually a forum on Internet, companies such as Claris and Apple have access to it and some of their employees actually read the mail and respond to it in their free time.

Well, now that I've strayed far from what I was originally going to talk about, here's what happened when I asked the folks on Info-Apple about the Programmer's Online Companion and AppleWorks GS problem. The first response I got from my question was from Jeff Noxon. He told me that the problem was with the Programmer's Online Companion and not AppleWorks GS. This was because ORCA/M and APW use a file called SYSTEMP as a clipboard instead of using the Scrap Manager. Well that just didn't sit right with me, and sure enough after conversing with Steve I learned that the Programmer's Online Companion worked with EGOed, a NDA which does use the Text Edit Tool Set, and the Scrap Manager. So, I tossed that solution out the window. (This is not to say that I did not appreciate Jeff's input, incorrect information can often lead to the correct solution.)

The next response I got was from Syndey R. Polk, an employe at Claris who actually worked on AppleWorks GS. Here's what he said:

*"There is no way for AWGS to know that the system clipboard has changed when Programmer's Online Companion is invoked. The system clipboard is read when AWGS has to relinquish control to an *NDA* and when the program is launched. A CDA can happen at any point, and there was no reason to trap calls to CDAs, since there is nothing in the Apple Human Interface Guidelines that says CDAs support the clipboard.*

What does this beast do when the Scrap Manager isn't started, like in PaintWorks Gold? It should crash when it makes a Scrap Manager call.

I bet it also doesn't work very well when a line edit item is active."

Ahhhhhh... now that's more like it! So the problem wasn't really AppleWorks GS at all. That's a relief. But what actually does happen when the Scrap Manager isn't started or when a line edit item is active? Well, I asked Steve to try these things out and he

said that the Programmer's Online Companion checks first to see if the Scrap Manager is active. If it isn't, the Programmer's Online Companion writes to the SYSTEMP file. Nothing bad happens with the Line Edit situation, either. Line Edit items don't appear to support the clipboard or the Scrap Manager. While this did not solve the problem, at least I found out why it was happening.

I didn't have to use BITNET to find this information. I could have picked up the phone and called Addison-Wesley or Claris (they have phone numbers for a reason, right?). Or, I could have found a different way to get access to the Info-Apple discussion, which I recommend to anyone remotely interested in Apple II computers.

You can reach the Info-Apple discussion from BITNET, Internet, UUCP, ProLine, and a whole bunch of other networks. ProLine is probably the easiest to access. All you need is a modem. Just call up the nearest ProLine sight, register to it, and join the appropriate conference. From the commercial networks, all you need is to have an account which is able to send and receive mail.

If you can't get an account on a commercial network or you just don't want to pay long distance phone bills to a ProLine sight, pick up a pencil and paper and write your grievances to us here at *GS+* magazine. This is one of the reasons this magazine is published, to help other Apple IIGS users get the most out of their computers. Who knows... maybe your problem could appear solved in the next Molehill column!

WRITER'S GUIDE

GS+ needs submissions! In the next few issues' we hope to double the number of pages here in *GS+*. But, to do that, we need material! If you would like to help others get the most out of their IIGS, and make a tiny amount of money too, take a look at the following guidelines and crank up your word processor!

General Information

All submissions become the property of EGO Systems. If we return your submission to you, we give up our rights to it and you are free to do whatever you want with it. All submissions to *GS+* should include the following items:

- 1) A cover letter (on paper please) telling us who you are and what your submission is.
- 2) Return postage (if you want your submission back).
- 3) A diskette (3.5" preferred) containing your article/review/program. We can read just about any word processor format, but we would prefer that you send your submissions to us in AppleWorks, AppleWorks GS or plain ASCII text format.

Submissions to *GS+* may be made by one of the following methods:

- 1) US Mail.
Send your submission to:
GS+ Submissions
c/o EGO Systems
P.O. Box 15366
Chattanooga, TN 37415
- 2) America Online
Pack your submission with ACU or ShrinkIT and send it to, 'Obnoxio'.

Please don't submit stuff that you've already put in the Public Domain. One exception to this would be a program that you may have originally released to the Public Domain and have since enhanced.

Articles

Articles should cover something that will help readers get more use out of their IIGS with the lowest cost and time investment possible. To be a bit less vague, here are some examples of articles we would like to print in the near future:

- 1) Using low-cost Macintosh and/or IBM hard drives (and other peripherals) with the IIGS.
- 2) A survey of IIGS programming languages.

The main point to remember is that we want all of the material we print to be genuinely useful to the largest number of readers possible. If you have an idea but aren't sure it is what we might be looking for, drop us a line through any of the means listed above and we will let you know what we think.

Feature Reviews

Feature Reviews should be between one and two thousand words in length and should fully describe the product and your experiences, both good and bad, with it. All reviews should, at the least, cover all of the following points:

• Introduction

What the product is, what it is supposed to do, who makes it, who wrote it, the publishers address and phone, how much it costs, and the minimum system required to use it.

• Good Points

Everything has its good points. Tell us about them. What is it that makes this product worth buying?

• Bad Points

Nothing's perfect. Every product will have something wrong with it. We want to tell folks exactly what those things are. This includes copy protection.

• Summary

Would you buy this product again? Would you advise that anyone buy it? Why not? What could be done to improve it?

Graphics

Everybody loves fabulous graphics. Send your original graphics to us along with a short letter explaining the tools and any special tricks that you used to create them.

Programs

Programs should be written in C, Pascal or Assembly Language and **MUST** include source code. You should also write an article explaining how to use the program. Source code should be heavily commented and structured so that folks can figure out what you are doing. Programs should follow Apple's Human Interface Guidelines. We want all of the programs published in *GS+* to be useful and/or entertaining. Here are a few examples of programs we would love to publish for the IIGS:

- 1) A IIGS specific telecommunications program.
- 2) A desk accessory to find files hidden away on a hard drive.
- 3) Some great games!

Rates

So, what's in it for you? Well, at this point, all we can afford to offer is the following:

For	You Get
Article or Feature Review	\$10
Each Graphic	\$10
Program	\$50 - \$250

We admit it's not much... but it's more than you get for keeping your programs, graphics and opinions to yourself! And, who knows, you might just help someone. So, get involved and become a part of *GS+*!

