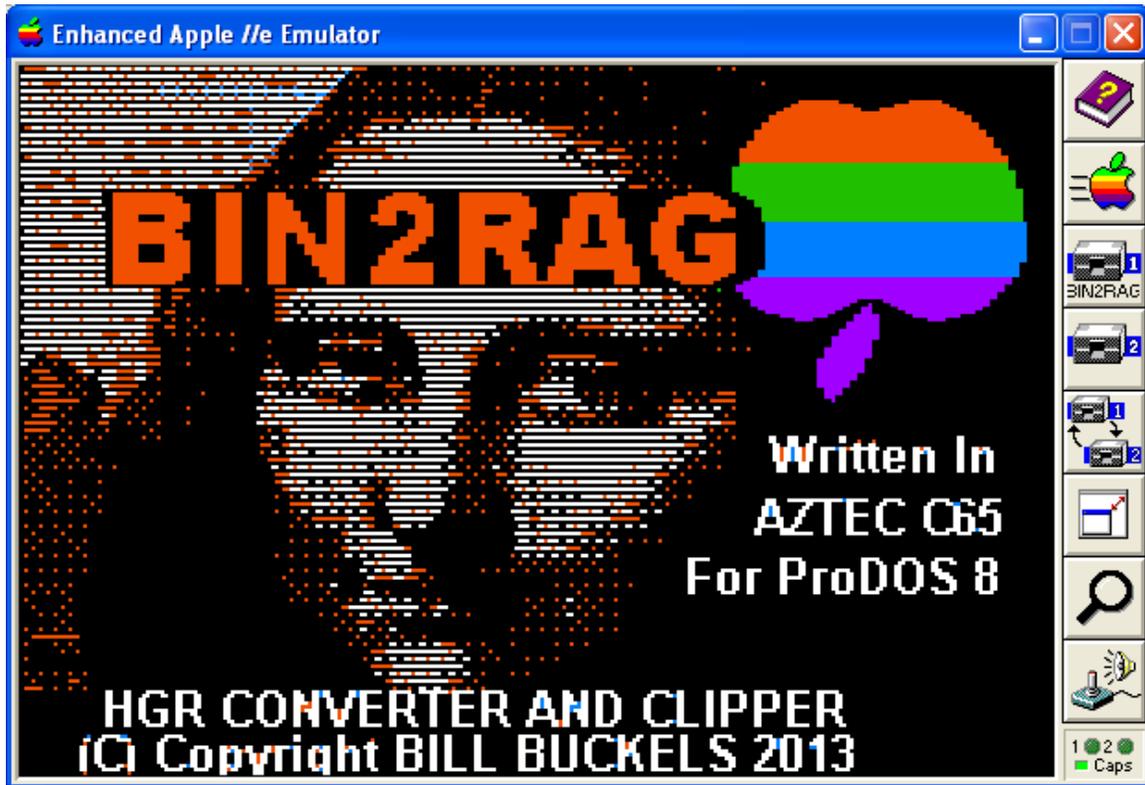


BIN2RAG – HGR Graphics Converter and Clipper for ProDOS 8
By Bill Buckels, March 2013



The BIN2RAG Splash Screen – Press Any Key to Start Using BIN2RAG

Table of Contents

BIN2RAG – HGR Graphics Converter and Clipper for ProDOS 8.....	1
The BIN2RAG Splash Screen – Press Any Key to Start Using BIN2RAG.....	1
Table of Contents.....	1
Dedication.....	2
Introduction.....	3
Getting Started – The File Selection Screen.....	3
Getting Started – The Clipping Screen.....	4
BIN2RAG Menu Commands.....	5
ClipBox Commands.....	5
Arrow Keys – Resize the ClipBox.....	5
Navigation Keys – Move the ClipBox.....	5
Reset Keys – Reset the ClipBox.....	5
Viewing and Editing Commands.....	5
File and Other Commands.....	6
Licence Agreement.....	7
Getting Started – Using The AppleWin Debugger to “Grab” HGR Screens.....	7
Synopsis: 2 - ways to capture HGR Screens from AppleWin	7

Scripting the AppleWin Debugger BSave Command.....	7
Entering the AppleWin Debugger to Capture HGR Screens.....	8
Saving HGR Screens Using AppleWin Debugger’s default naming:.....	8
The Down-Side of the Debugger’s Default Naming and Some Options.....	9
Getting Sore Fingers and Wasting Time typing File Names.....	9
The Production Option - hgrServer.....	9
Running and Using hgrServer.....	10
Closing Remarks.....	12
Appendix A – A Monochrome HGR Screen produced using the AppleWin Debugger.....	13

Dedication



Shown above in BIN2RAG’s HGR display, a typical [AppleWin](#) User first hearing the happy news that BIN2RAG comes with [hgrServer](#), so he can steal perfect native mode HGR screenshots in BSaved format in the AppleWin debugger without typing his fingers to the bone, and getting more donut juice on his keyboard than it already has. And then chunk-them into “sprites” using BIN2RAG to use in his own Aztec C programs.

Thank you Michael Pohoreski and the AppleWin team for all your good work and the many years of fun that you have given us.

Introduction

BIN2RAG is a Hi-Res (HGR) Graphics Utility. It runs-in ProDOS 8 on An Apple IIe.

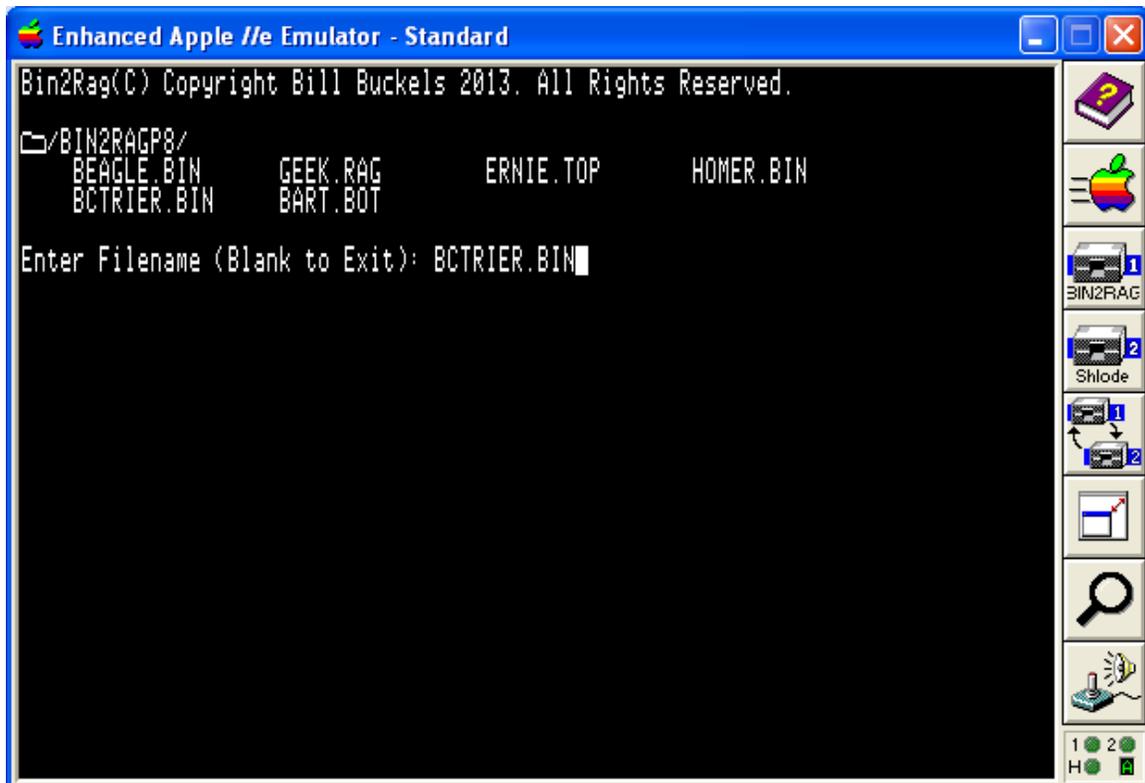
BIN2RAG serves 2 main purposes:

- “Round-Trip” Conversion between BSaved full-screen HGR Images (.BIN files), and full-screen and mixed-screen Raster Oriented HGR Images (.BOT and .TOP files), and Raster Oriented HGR Image Fragments (.RAG files).
- BIN2RAG produces Raster Oriented HGR Image Fragments (.RAG files) from the formats noted above.

If command-line utilities are not “your thing” when it comes to chunking-up an HGR image that has been created from an AppleWin Screen Capture or BSaved Memory “dump”, or (more likely) you just like to play with HGR bitmapped graphics on the Apple II, this little SYS program (BIN2RAG) may be fun for you to use.

But wait, there’s more!

Getting Started – The File Selection Screen



The file selection screen does not select a file for you. You type the filename in yourself.

The files that are listed are a “guess” of the files that could be graphics files in the current ProDOS directory. See the BIN2RAG source code for more information about how BIN2RAG guesses which files are HGR Graphics Files...

When BIN2RAG starts, and you press a key to “skip-past” the Splash-Screen (the Mona Lisa Screen with the upside-down and backwards “biting-apple” logo), the file selection screen is the next screen you see, and it is also the last BIN2RAG screen you see before BIN2RAG exits to ProDOS.

To exit from BIN2RAG to ProDOS, simply enter a blank line at the File Selection Screen instead of typing-in an HGR graphics file name.

Otherwise, if you want to load an HGR graphics file for conversion, or just to view it, type the filename (BIN2RAG will also load prefixed filenames from other disks and directories) and press RETURN (the ENTER key). You will be returned here, to the File Selection Screen, each time you leave the Clipping Screen, after either saving or abandoning your new graphics file that can be created from the file you selected.

Getting Started – The Clipping Screen



The BIN2RAG Clipping Screen shown above is displaying a “clipbox” around the geek.

BIN2RAG Menu Commands

The clipping screen menu (which is toggled on and off with the spacebar) is shown above below the geek. Menu commands are single characters. They are as follows:

ClipBox Commands

The BIN2RAG ClipBox is an inverse-video “xored” box around the area that will be saved. When an HGR image is loaded into BIN2RAG, it loads to the top-left corner, with the ClipBox surrounding the HGR image. If the image is a fragment, the ClipBox will initially be placed around the fragment.

The idea here is that you resize the ClipBox, and move it around on your HGR screen, then you save the area that is enclosed in the ClipBox to a new HGR file.

Arrow Keys – Resize the ClipBox

Left Arrow – Make ClipBox Narrower – 14 pixel (2 byte) increments

Right Arrow – Make Clipbox Wider – 14 pixel (2 byte) increments

Up Arrow – Make ClipBox Shallower – 1 raster increments

Down Arrow – Make ClipBox Deeper – 4 raster increments

Navigation Keys – Move the ClipBox

R – Move ClipBox Left - 14 pixel (2 byte) increments

L – Move ClipBox Right - 14 pixel (2 byte) increments

U – Move ClipBox Up – 1 raster increments

D – Move ClipBox Down – 4 raster increments

Reset Keys – Reset the ClipBox

F - Full-Screen Reset – 280 x 192 (Full Graphics Screen)

T - Top-Screen Reset – 280 x 160 (Mixed-Text Graphics Screen)

X - Start-Over with initial ClipBox (Image) Size “Panic Button”

Viewing and Editing Commands

RETURN (Enter Key) – Pause and View Image (hide the ClipBox and the Menu). Good for Windows (or AppleWin) Screenshots. After viewing press any key to resume. The Menu and the ClipBox will be displayed again.

Numeric Keys 0-7 – Select Background Color (relatively useless except for image fragment screenshots, or to expand image fragments with a matching background color). The colors are in standard “HCOLOR” order... Green and Violet are 1 and 2, Orange and Blue are 5 and 6, Black is 0, and 4, and White is 3 and 7.

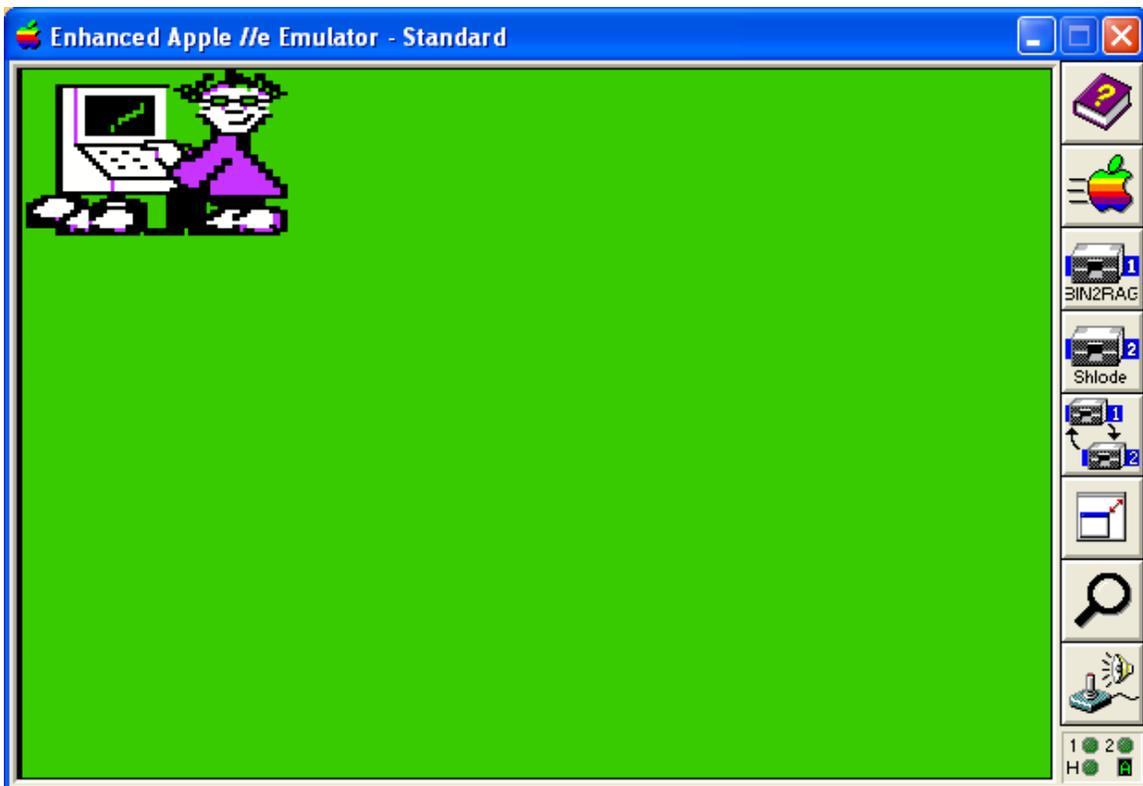
- I – Inverse Video (also relatively useless)
- G – Select Green-Violet Single Palette (somewhat useful)
- O – Select Orange-Blue Single Palette (somewhat useful)

File and Other Commands

- ESC – Return to the file selection screen without saving.
- DELETE – Delete the Currently displayed Image. You will be prompted first.
- S – Save ClipBox Area in rasterized BOT, TOP, or RAG format.
- B – BSave entire Screen in BIN format.

When Saving or BSaving, BIN2RAG will also accept a prefixed filename to other ProDOS disks or directories.

After abandoning, saving, or deleting you will return to the file selection screen where you can select another image (or exit to ProDOS by entering a blank line instead of an HGR image name).



The geek has loaded... As shown above, after saving the “geek” he can be loaded into the BIN2RAG Clipping Screen for validation, resaving, or deletion (or whatever).

Licence Agreement

You have a royalty-free right to use, modify, reproduce and distribute BIN2RAG, related source code, and related demos and the other stuff it comes with (“baggage”) including this document. You may use this program and its “baggage” for whatever you wish, provided that you agree that I have no warranty obligations or liability resulting from said use and distribution in any way whatsoever.

This seems fair enough since the rest of the planet can do exactly the same and I hope they will (at their own risk, of course, although I say this kindly).

Getting Started – Using The AppleWin Debugger to “Grab” HGR Screens

Synopsis: 2 - ways to capture HGR Screens from AppleWin

1. If you use the AppleWin emulator, you may be aware that you can capture a “rendered” screen from a running program within AppleWin as a Windows bmp file. These captured BMP’s can be optionally edited in Windows Paint, then converted back to Apple II HGR images using utilities like my BMP2RAG and DMONSTER programs. You can read a little more about doing exactly this in the following document:
http://www.aztecmuseum.ca/AppleWinScreenGrabs_2013.pdf
2. However, perhaps the simplest way to acquire (“Grab” or otherwise “steal”) an HGR screen is to use the debugger that is built-in to AppleWin to BSAVE HGR screen memory (BIN files) directly to your Windows drive, and then to place these Apple II HGR files on an Apple II Disk Image. After doing so, you can sort-of edit these in BIN2RAG.

In the pages that follow, I will:

- Provide a step by step summary of using the AppleWin debugger to capture HGR screens
- Introduce a Windows Program called hgrServer to help you organize your captured screens during the capture process.

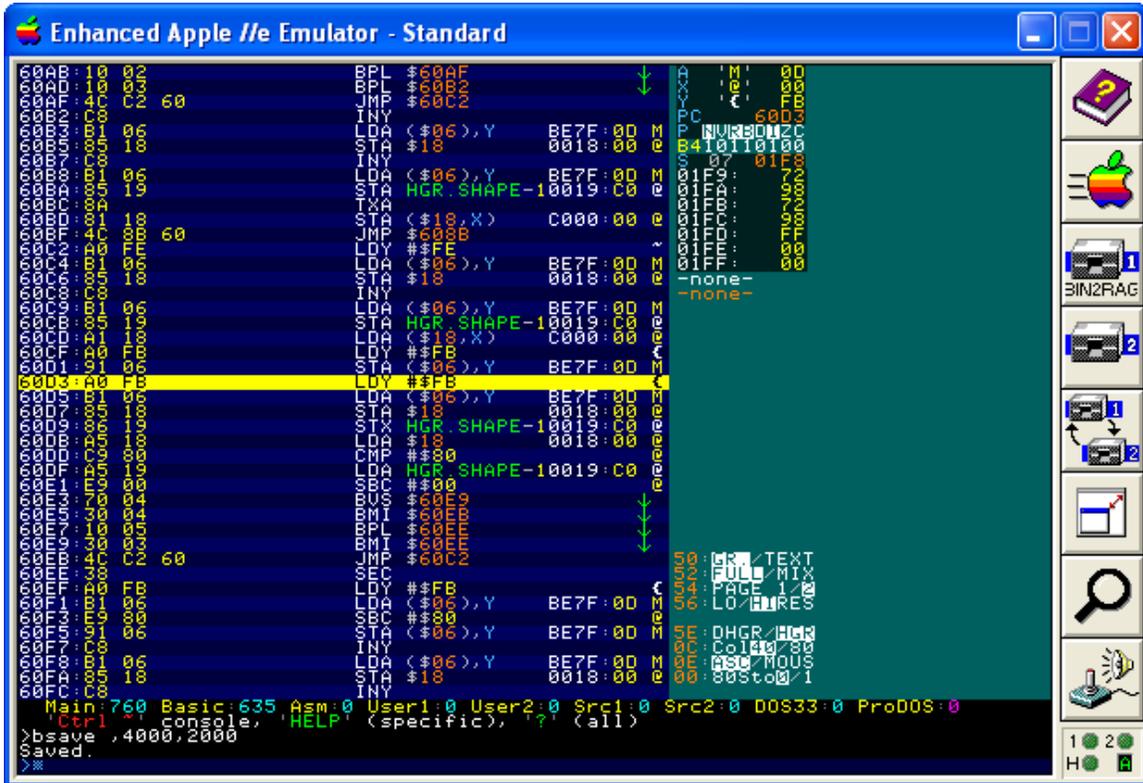
In addition to the details I provide, please see the AppleWin documentation for more information related to AppleWin. The AppleWin source code is informative as well.

Scripting the AppleWin Debugger BSave Command

The AppleWin debugger supports scripting. BIN2RAG comes with 2 simple scripts (hgr1 and hgr2) to “grab” HGR memory and BSave it to BIN files on your Windows disk.

So it is your choice. For example, to save from HGR page 1, you can either type-in the AppleWin debugger's BSave command "BSAVE ,2000,2000" and press ENTER or type "RUN HGR1", and press ENTER. But I am getting ahead of myself.

Entering the AppleWin Debugger to Capture HGR Screens



Before Entering the AppleWin Debugger to capture HGR Screens, you must have an Apple II program running (in the emulator) displaying an HGR screen. When you see an HGR screen you want:

1. Press the Pause Key to stop the Apple II Program
2. Press the F7 Key (or click-on the magnifying glass on Applewin's toolbar to enter the debugger)

Saving HGR Screens Using AppleWin Debugger's default naming:

1. Look at the right side of the debugger display to see if the HGR screen is using page 1 or 2.
2. If Page 1, type-in "b save ,2000,2000" or type "run hgr1" and press ENTER. AppleWin saves to 2000.2000.bin.
3. If Page 2, type-in "b save ,4000,2000" or type "run hgr2" and press ENTER. AppleWin saves to 4000.2000.bin.

The Down-Side of the Debugger's Default Naming and Some Options

The down-side of using the debugger's default naming is that the next time you BSave the same memory using default naming, the debugger will over-write the previous save after it warns you. Keep in mind though that this is a debugger, and as such it is not designed exclusively for capturing screens... in fact it does infinitely more than dump memory. The BSave command itself has some options, and one of these is to type-in the filename manually.

Getting Sore Fingers and Wasting Time typing File Names

You may already have noticed that running my humble scripts requires less typing than typing-in the BSave command itself. You have two options at this point that can take you even more time (we'll have a look at hgrServer later for a better option):

1. During screen "dumps" you can leave the debugger for a moment and click-on a Windows Explorer window or run a cmd window, and do some more typing and manually rename 2000.2000.bin or 4000.2000.bin to something else. While you are at-it you may as well right-click-on the file and load-it into [ClipShop](#) to see if you want it or not before bothering to rename it.
2. You can type the filename in manually to the BSave command and waste a lot of good energy especially if you don't want the file... and I guess you probably understand like I do that all this handling can be a big waste of time at times... but anyway here's an example of BSaving a Page 1 HGR bin file: Bsave MyGrab.bin,2000,2000

And yes, this is ok to do if you just want to grab an occasional HGR screen in bin file format here and there. But if you are an AppleWin Power User like I am you may want more.

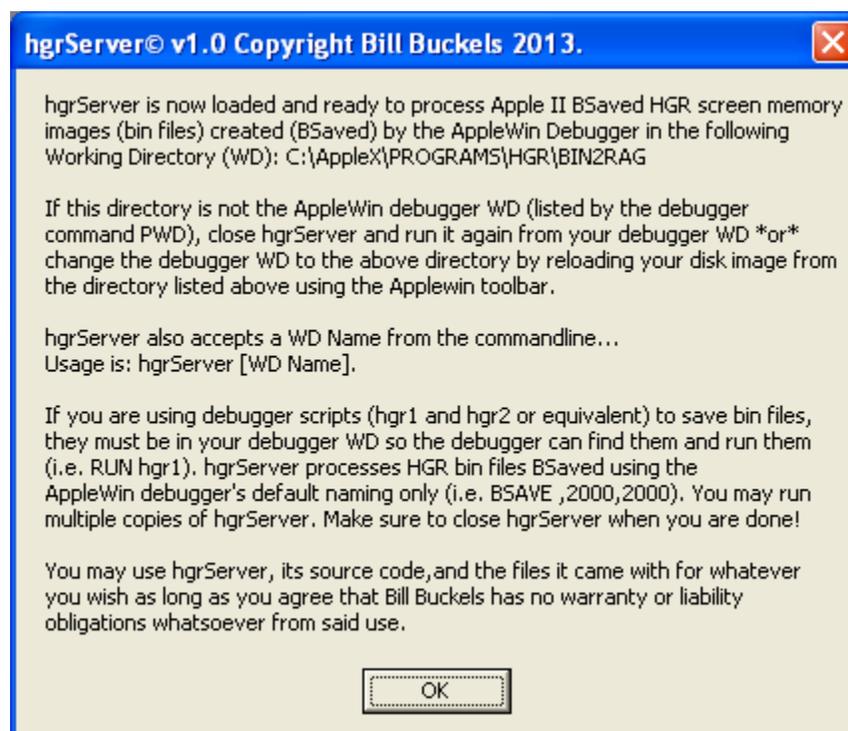
The Production Option - hgrServer



hgrServer is a very simple program. It is not an application or even a utility. All it does is runs in the background and renames default filename and numbers them sequentially while you merrily BSave HGR Screens in the AppleWin debugger using the debugger's default naming.

Later-on you can look at these and decide which ones to keep.

I apologize profusely for wearing you out with all my talk about other options. If I thought there were other options that were any good at all, I would not have taken time to write my little PowerToy:



Running and Using hgrServer

The hgrServer's welcome screen (above) says it all but here's some more anyway:

hgrServer is a native Win32 program. It is not huge like the bloatware that requires all sorts of installations and stuff... just drop-in hgrServer anywhere that Win32 applications run... no installation needed. And it comes with source code for those of you who like to see what you are using before you use it.

It is set-up to use a 5 second delay between file renaming, so unless you can type and dump like a banshee it should keep-up with you. And it doesn't interfere with AppleWin and the debugger (based on my testing).

When hgrServer starts, (see screen above) it notifies you that it is now loaded and ready to process Apple II BSaved HGR screen memory images (bin files) created (BSaved) by the AppleWin Debugger in the following Working Directory (WD):

It then prints the directory that it is currently running-in.

If this directory is not the AppleWin debugger WD (listed by the debugger command PWD), you can either:

- Give-up and do something else.
- Close hgrServer and run it again from your debugger WD *or*
- Change the debugger WD to the hgrServer WD by reloading your disk image from the directory listed by hgrServer's Welcome Screen using the Applewin toolbar.

hgrServer also accepts a WD Name from the commandline...

Usage is: hgrServer [WD Name].

If you are using debugger scripts (hgr1 and hgr2 or equivalent) to save bin files, they must be in your debugger WD so the debugger can find them and run them (i.e. RUN hgr1).

hgrServer processes HGR bin files that are BSaved using the AppleWin debugger's default naming only (i.e. BSAVE ,2000,2000 produces 2000.2000.bin).

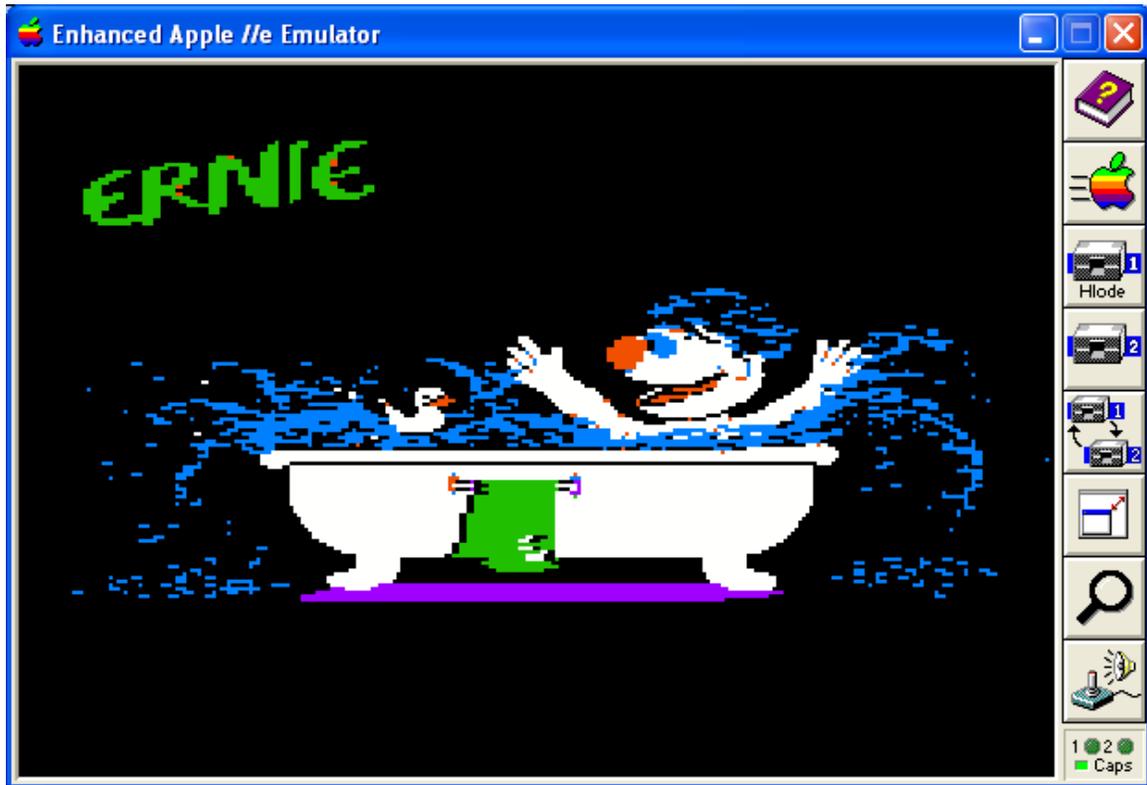
You may run multiple copies (instances) of hgrServer all over your computer if you wish.

Make sure to close hgrServer when you are done! (better option than the one above).

You may use hgrServer, its source code, and the files it came with for whatever you wish as long as you agree that Bill Buckels has no warranty or liability obligations whatsoever from said use. If you do not agree and use hgrServer anyway, may you turn into a guitar string on the International Space Station!

hgrServer is distributed with BIN2RAG. This is not really a feature and may not be a fact in the future. My plans for hgrServer probably include handling all the graphics modes on the Apple II once I work with AppleWin some more and get the time to extend this PowerToy.

Closing Remarks



I realize this document is neither overly technical, nor particularly informative to those of you who are intimate with HGR graphics on the Apple II and the AppleWin emulator. Despite that I am ending here.

I hope those of you who have a use for Apple II HGR graphics files will add BIN2RAG to your toolbox, and that you [AppleWin](#) users will add hgrServer to your toolbox. And I hope that after you see what is possible with Aztec C65 for the Apple II (BIN2RAG is written in Aztec C65), you will download the [AppleX](#) 2013 update in droves.

False Modesty prevents me from extolling the virtues and excellence of these little Apple II toys of mine further.

Have Fun!

Bill Buckels
bbuckels@mts.net

Appendix A – A Monochrome HGR Screen produced using the AppleWin Debugger



THE REST OF THIS APPENDIX UNINTENTIONALLY REMOVED

For more information call 411.