

Apple IIGS System Software Release Notes

Version 6.0.1
May 21, 1993

These release notes detail changes made from version 6.0 to 6.0.1 of the Apple IIGS System Software.

Much of the material here is intended for programmers. For information affecting users, read the Whats.New and Shortcuts files on the SystemTools2 disk.

This document does *not* describe new 6.0.1 toolbox and GS/OS features in detail. See the 6.0.1 addendum to Programmer's Reference For System 6.0.

Sys.Resources Changes:

Changed the rVersion resource ID=1 string to read "System Software" instead of "System" (this is important for the Finder's reorganized About box).

Updated rErrorString resources to say "disk" instead of "volume."

Fixed the standard window color table resource to have a white info bar fill color (the byte at offset 8 is now \$F0 instead of \$00).

The Icon Button control now supports "sticky" icon controls. If bit 4 of the ctlFlag field is set and the mouse button is released when the cursor is inside the control, the control stays highlighted to show that it is "selected." The ctlValue field contains \$0001 when the icon is in the selected state, and \$0000 when it is not. An extra parameter, #12, has been added to the control template to allow for an initial value word for this type of control.

Setting a thermometer control's value no longer draws anything if the control is invisible.

The Launcher now zeros the BUSY flag (this makes Option-Open work in the Control Panels NDA).

The Launcher sets the noResourceMgr bit in StartUpTools (to avoid a duplicate startup, which StartUpTools now checks for).

ToolBox Changes:

Window Manager v3.4 Tool014

Added a new call, UpdateWindow, to help redraw other people's windows safely.

AlertWindow now uses UpdateWindow.

DoModalWindow now uses UpdateWindow. If the mwUpdateAll bit is clear, it passes flags of \$0000 for the dialog window, but \$8000 (background update) for other windows--so the other windows update only if it's safe for them to background-update. If the mwUpdateAll bit is set, it always passes flags of \$0000. It returns an error (\$0E05, cantUpdateErr) for a background update of a System window, since System windows should always get updated from inside SystemEvent.

Changed DoModalWindow so it doesn't invalidate controls on an activate if the window's fCtlTie bit is set, saying that the control states are independent of the window state.

After DoModalWindow does an LECut, LEPaste, TECut, or TEPaste, it returns the ID of the control that was just edited.

HandleDiskInsert duplicate-disk dialog now uses the Note alert icon instead of the 3.5" disk icon, since the dialog also appears for disks that look nothing like 3.5" floppies.

HandleDiskInsert now treats error \$51 (directory damaged) like error \$5A (damaged bitmap); it puts up "This disk is damaged and can't be used. Eject/Initialize".

Bits 31-24 of the updateProc to HandleDiskInsert are now honored even if the rest of the pointer is NIL. This way you can flag your window as not needing GS/OS for its update.

HandleDiskInsert never replaces a window's existing wContDraw procedure, if it has one (we give it a wContDraw on the first call if it doesn't have one).

Menu Manager v3.4 Tool015

In AllocateCache, we now clip the menu height to 200 for computing the size handle needed to save the screen. We were overflowing the computation for tall pop-up menus. See also TS3.

When the pop-up control defproc handles a do_event message, now it only sends keyDown and autoKey events to MenuKey, not any old event. It also preserves the menu bar around the MenuKey call, so we don't accidentally leave the menu bar set to the pop-up menu control.

DeleteMenuItem now properly shrinks the menu handle by twelve bytes (was ten). Not worth patching for ROM 3 (you get the bytes back when the menu goes away anyway).

Pop-up proc uses DrawStringWidth to draw non-popped-up pop-up controls (it truncates long items in the middle).

Control Manager v3.4 Tool016

Added four new calls: SetCtlValueByID, GetCtlValueByID, InvalOneCtlByID, and HiliteCtlByID.

CtlStartUp removes the RefreshDesktop run queue routine, so the desktop doesn't refresh an extra time when starting an application in a different resolution from the one you used last (the scroll bars thought they had to redraw in 6.0. even though they really didn't).

The Static Text control now supports a new ctlFlag bit. If bit \$10, fSquishText, is set as well as fBlastText, the control will draw the text with DrawStringWidth to compress and truncate on the right.

Replaced one GetSysPrefs call and two SetSysPrefs calls with two DoSysPrefs calls, which should significantly speed up every CMLoadResource call. (DoSysPrefs has lower overhead; it does not call GS/OS internally.)

QuickDraw II Auxiliary v3.5 Tool018

Added three new calls: DrawStringWidth, UseColorTable, and RestoreColorTable.

DrawStringWidth draws a string to fit in a specified horizontal width, compressing and/or truncating the string as necessary. The string can be Pascal, C, or GS/OS, and you can reference it by pointer, handle, or resource. The truncation is an ellipsis on the left, on the right, in the center, or no truncation. Compression is optional.

UseColorTable preserves SCBs and sets them to use a color table you specify. It also preserves the old contents of that color table and sets the color table to the data you specify, or to a standard color table. UseColorTable returns a handle that you later pass to RestoreColorTable to put things back how they were. Generally, you will call UseColorTable on an activate event and RestoreColorTable on a deactivate event.

Added several new icons to GetSysIcon: text (\$04), source file (\$B0), Applesoft BASIC program (\$FC), archive (\$E0), and binary file (\$06).

GetSysIcon now calls SendRequest with a new request code, systemSaysGetSysIcon, to allow utilities and applications to provide a replacement icon.

Print Manager v3.2 Tool019

Fixed bad-parameter error code from PMLoadDriver & PMUnloadDriver to be \$1309, as documented (was returning \$1308 before).

Line Edit v3.4

Tool020

Added a new call, LEClassifyKey, which classifies a keystroke as some combination of digit, letter, hex digit, editing key, and non-control-character.

Added a new field to the Edit Line control template. Parameter number 9 is a word called keyMask. The control accepts keys only if the LEClassifyKey result has some bits set in common with the keyMask parameter.

Changed interpretation of the Edit Line template pwChar field: A value of \$FFFF now means the control is not for password entry. (Previously, the legal values were \$0000 [default password character] and \$0001 through \$00FF [specific password character]. The parameter's presence implied that the control was for password entry, which is no good with the new optional ninth parameter.)

Made LERSetText erase the gap between the destination and view rectangles at the left edge of a scrolled field.

Dialog Manager v3.5 Tool021

Fixed a bug that caused a crash if you made a SetDAFont to the FastFont handle (returned by GetSysFont), and then caused the Dialog Manager to beep. (This affected ROM 1 systems only.)

Scrap Manager v3.2 Tool022

Added one new call, ShowClipboard, which creates a System window that takes care of the clipboard display for you (it's used in Finder and Teach, for example). It displays Text, Picture, and Sound scraps. If you copy a Sound to the clipboard (using the Sound control panel, for example), you can click on the Speaker icon on the clipboard to hear the sound. It also calls SendRequest with a new request code, systemSaysDoClipboard, to allow utilities and applications to customize or extend the clipboard display.

PutScrap now changes the scrap count, as returned by GetScrapCount (for polling to see if the clipboard contents changed).

Standard File v3.4

Tool023

Uses DrawStringWidth to draw standard items (allows compression and truncates on the right with an ellipsis).

Added three new key equivalents. Command-Shift-UpArrow is the same as a Command-D or a clicking the Disks button. Command-RightArrow is the same as Command-Tab; it moves forward to the next drive in the list. Command-LeftArrow moves backward to the previous drive in the list.

SFReScan now makes a DInfo and Volume call on the volume in prefix 8 and updates all of the controls accordingly. SFReScan now works in the volumes list as well as the files list.

Fixed a hanging problem on SFGetFile (not SFGetFile2) when prefix zero was empty.

Fixed a bug when closing the dialog where if the prompt string was a resource, it was being treated like a handle.

Standard File windows now update behind modal windows and try to update back windows using UpdateWindow.

Changed the way Standard File handles multiple edit line items in PutFile dialogs, so that there can be more than just the single edit line item.

Fixed a bug in the check that was trying to restrict the promptRefDesc to values 0-2 but wasn't returning an error if it was out of range. It now returns badPromptDesc (\$1701) as documented.

Now passes a real auxiliary type to GetSysIcon (was always passing zero in 6.0).

Volume name comparison in NextDrive routine is now insensitive to case, so if your prefix doesn't match the actual case of your disk, Command-Tab now takes you to the next disk instead of staying at the same one (but fixing the prefix).

Simplified no-volumes message to read "No disks are available. Please insert a disk."

Changed "volume" to "disk" in several messages.

Now shows "zero K" instead of "OK" in space-on-disk messages (the guts of this change are in Long2Dec in Integer Math).

Font Manager v3.4 Tool027

Expanded the size list in ChooseFont to 96 members.

Added code to strip out duplicate sizes from the size list in ChooseFont. Duplicates of the same size would show up if two or more real fonts in the same family had the same size (but different styles).

List Manager v3.4 Tool028

CompareStrings flag bit 15 now makes it compare GS/OS strings instead of Pascal strings.

ListKey takes Clear and Delete to mean "zap the prefix" (just like waiting 2/3 of a second before typing).

Fixed a bug affecting ListKey, CompareStrings, and SortList/SortList2 with compareProc of 1. Characters \$20 to \$3F (including digits), and \$60 were being accidentally "uppercased." For example, in 6.0 "5" would map into right-arrow, which moved down one item.

The standard item-draw procedure uses DrawStringWidth, with flags allowing horizontal compression and truncation on the right with an ellipsis.

TEPaintText now properly fully-justifies text.

Fixed a problem with non-targetable Text Edit controls. They could start out active (with a usable scroll bar for example), and then become inactive when the window became inactive, but the control would not get reactivated when the window came back to the front.

FMStatus call now pushes pre-zeroed result space, in case Font Manager is not loaded (the result is that you get an error reliably now, instead of just sometimes, if the Font Manager isn't available).

Added two new calls: `OpenResourceFileByID` and `CompactResourceFile`.

`OpenResourceFileByID` starts the Resource Manager for you if it isn't already started under your ID (and it makes you the current resource application if it was already started for your ID). Then it uses `LGetPathname2` to find pathname for the specified ID and calls `OpenResourceFile` for you on that file. Returns errors from `LGetPathname2` and `OpenResourceFile`. `oldResApp` is valid even if you get an error.

`CompactResourceFile` removes all free blocks (except the last special EOF free block) from a given file by moving chunks around in the file.

Added some error checking code to `OpenResourceFile` that absolutely makes sure that the map was entirely read; if what is read doesn't equal what we asked for, it returns a GS/OS `eofEncountered` error.

Added a new bit to `mapFlag`. Bit 0 is now defined as `fileReadWrite`. When a file is opened, it gets set to 1 if the file is opened read/write. If it's opened with read-only access, then the bit is set to 0. This bit is for examination only.

`AddResource`, `RemoveResource`, `WriteResource`, and `MarkResourceChange` now verify that the target file can be written to before actually doing anything. They all return a GS/OS `invalidAccess` error if the file cannot be written to. The exception to this is `MarkResourceChange` when the resource in question is being marked unchanged; it is allowed as it won't eventually cause a write.

Fixed `WriteResource` to write the size of the resource as it appears on disk, rather than the size of the resource's handle in memory. This properly allows for converters to write resources that are smaller than their in-memory size without destroying the file.

`CloseResourceFile` returns error `resFileNotFound` (\$1E07) instead of no error, on a nonzero argument that doesn't match an open file ID.

Fixed string-comparison bug in `RMFindNamedResource` (and `RMLoadNamedResource`, which calls `RMFindNamedResource`). Sometimes in 6.0 you could wind up loading a resource whose name began with the name you asked for but was longer.

Tool.Setup

Tool.Setup contains patches that are needed for both ROM 1 and ROM 3.

Tool Locator

Optimized SendRequest (25% speed improvement in some cases). Optimized AcceptRequests and GetMsgHandle, too.

StartUpTools now returns any error from ResourceStartUp. When we would have returned garbage for the result, we return NIL.

StartUpTools and LoadTools now skip the message "This application needs ToolXXX" for tools that are present but too old (like ROM-based tools!). They still return an error code to the caller.

ShutDownTools checks for a NIL input and behaves sanely. ResourceShutDown is now in the error-exit path, so it shuts down the Resource Manager even if you get some other error.

ShutDownTools tolerates errors from SFShutDown (for compatibility with errant NDAs that shut down Standard File during Desk Shutdown even if they did not start it).

ShutDownTools no longer calls HideCursor if QuickDraw is not active (in 6.0, it can crash if an application calls ShutDownTools with QD inactive). If the new-for-6.0 "no resource manager" flag bit is set, there is no problem (it was already skipping the HideCursor then).

Misc Tools

Added two new calls: DoSysPrefs and AlertMessage.

DoSysPrefs clears and sets the specified bits in the GS/OS system preferences word, and then returns the previous value (error \$0301 if GS/OS is not the current OS).

AlertMessage provides access to code inside GS/OS that is smart enough to put an alert on either the text screen or the Super Hi-Res screen, depending on the environment.

SysBeep2 now sends a new SendRequest code, systemSaysForceUndim, for all codes except \$006x (screen blanking, screen unblanking).

ShowBootInfo now "wraps up" to a new row if you have more than one row of icons. If you wrap off the top of the screen, it starts over at the bottom left, without erasing the screen. (It used to keep recycling the bottom row, wiping it to periwinkle blue every time it filled up.)

Desk Manager

Setting bit 31 of an action procedure pointer (for NDA or system window) does stack-based dispatch.

function MyActionProc(code: integer; data: longint): integer;

CDA menu checks ROM version before using ReadKeyMicroMemory to check for the control-panel-disable jumper. Was misbehaving on upgraded Iie ROM 1 machines.

ROM 01 Patch TS2

If Battery RAM byte \$59 bit 7 is set, TS2 installs Memory Peeker and Visit Monitor (just like ROM 3 does).

Event Manager

EMStartUp uses (26,17) (or (13,17) in 320 mode) for the initial mouse position if there is no message #6 specifying a position. This prevents top-left screen dimmers from dimming right away.

Integer Math

Int2Dec and Long2Dec return "zero" if the number being converted is zero, bit 31 of stringPtr is set, and the buffer length is at least 5.

ROM 03 Patch TS3

Desk Manager

DeskShutDown zaps \$07FC if slot 4 is internal. This stops the mouse from freezing in desktop applications after visiting the CDA menu, when you have previously run an application that left a nonzero value in \$07FC. This was a problem on ROM 3 systems only.

OpenNDA sends reOpenAction to action routine of an already-open NDA, to give the NDA a chance to do something other than just have the window come to the front.

Integer Math

Int2Dec and Long2Dec return "zero" if bit 31 of stringPtr is set and the buffer length is at least 5.

Patched Multiply to fix Menu Manager bug: AllocateCache was multiplying the bytes-per-line by the menu height, without clipping the height to 200 (screen height), as it does when it really uses the memory it allocates.

Misc Tools

SysFailMgr now provides useful default messages for common errors:

\$27: "Could not read or write disk. The disk may be damaged."

\$201: "Out of memory (or required memory area was already in use)."

\$308, \$681, \$682: "Detected trashed memory. Software bug or (less likely) bad RAM."

SANE

Fixed a problem in the ElemNum call FXPWRI where the source integer exponent was never being classified (whether it was zero, normalized, etc.), but we were still checking Src_Class to see if we should proceed. This could make really strange things happen in some applications. (For example, the AppleWorks GS spreadsheet could get stuck in an infinite loop if certain values were left over in bank zero.)

Scrap Manager

So that we can use the disk-based scrap later, undo what the ROM-based ScrapBootInit did before GS/OS was available (dispose all \$4201 memory and set the Scrap Manager's WAP to NIL).

Window Manager

Fixed TaskMaster to handle the tmNoGetNextEvent bit correctly (it was treating the bit as reserved). This bit has existed since 5.0.3, but until now it worked only on ROM 1.

Fixed a HandToHand call in WindStartUp which was copying data from a small handle to a big handle, using the size of the big handle (this was not harmful, but it was unclean).

AppleTalk Port Driver v3.2

Reworked significant pieces of the driver to try much harder to establish a connection without error, and to write data to the PAP connection successfully. Also made PAPStatus try harder to get a response.

Changed PrDevOpen to wait forever for the PAPOpen to complete, or until the user hits Command-Period. It always retries on \$0808 error (server too busy to respond). Retries ten times on \$0805 error (name not found).

Changed PrDevWrite to wait forever for the PAPWrite to complete, or until the user hits Command-Period.

Removed some completely unused code (driver is now 6 blocks on disk instead of 8).

Changed the algorithm used to determine if AppleTalk is running. It used to check to see if a slot was set to "AppleTalk". It now checks for the presence of the AppleTalk main driver (which is the supported way anyway).

LaserWriter Printer Driver v3.3

The driver appears to work with the LaserWriter Pro 600 and 630, but this combination has not been fully tested.

Changed PostScript definition of /spf (the spotting function) to get it defined correctly on level 2 printers, using a hard-coded default spot function for when the dictionary isn't of HalfToneType 1.

When we bypass the code that sets up the name tables and sends queries to the printer (because we're creating a PS file), we now explicitly set the LWType variable to 1, indicating a LaserWriter Plus or II. Now the code can find all the standard 35N fonts in the built-in name tables, instead of just the four faces used in LaserWriters plus Sonata, Carta and Shaston. This means you can now dump PostScript files that work correctly with the 35N fonts. (The "35N" fonts are included in the LaserWriter II and later.)

Changed drawing of job dialog to include a call to PrGetNetworkName so we can draw the chosen printer's name in the job dialog. If there is no network printer (PrGetNetworkName returns NIL), it draws "No network - not supported".

Added 3 lines in CkPrinterStatus to avoid a possible infinite loop. If our session timed out we start getting error \$0806 from ReadLaser, but we didn't care.

When printing to a PostScript file, there may be no user name. If PMGetUserName returns NIL, we now cleanly don't put a user name into the file. It used to insert garbage from \$00/0000 in the file.

IWE m 1.6

The test for the Personal LaserWriter NT (in order to fix the smaller imaging area) now checks that the product name contains the string "Personal", so it will also match the Personal LaserWriter NTR.

GS/OS Changes:

ProDOS (The Boot Stub, not ProDOS 8)

If the user hits the "8" key before we run, then we load and run System/P8 instead of System/Start.GS.OS.

Added code to compensate for a ROM 3 bug where the RAM5 RAM disk does not get created if you have 8 Megs of RAM. The ProDOS file now creates it for you and restarts the system (even if you hit "8" to boot into ProDOS 8). This means you get an extra beep on boot IF you have an 8 Meg ROM 3 and a nonzero RAM5 size.

GS/OS (composed of the files GS.OS, GS.OS.Dev, and Start.GS.OS):

SCM

Added code to put a \$4C at \$BF00 at the beginning of every GS/OS call, so the Vulcan without a loaded driver will be happy.

Two changes in the interrupt dispatcher:

- Returns CLC if and only if at least one interrupt handler claimed the interrupt (was previously clearing carry when the *last* handler claimed it).
- Restores MSLOT after dispatching to an IRQ.OTHER handler.

OSShutdown was accidentally skipping the DeskShutdown call if flag bit 2 was clear.

Changed SysBeep2 call in s_mount_msg to clean up stack on error (in case Tool.Setup isn't loaded yet). Was causing incoherent "yes/no/cancel" text dialogs during boot, if it tries to ask for the boot disk before Tool.Setup time.

Restart no longer changes memory at \$0300.

GS/OS sends systemSaysForceUndim before deciding whether to put a message on the text screen or the Super Hi-Res screen.

GLoader/GQuit

After you press a key to continue (from "pause mode"), it erases the message so you know something happened.

On a switch to ProDOS 8, instead of zeroing UNIT (\$43), we store a \$50 there. Zero was doing horrible things when left in DEVNUM and later fed to ON_LINE (returns much more data than expected). This was only left at zero in the somewhat strange case that you network-booted into GS/OS, and then switched to P8 but have so many devices online that your boot volume wasn't in DEVLST.

Device Dispatcher

Implemented a new driver characteristics bit to let us add new subcalls to DStatus and DControl. Drivers have always been required to validate all call requests that are sent to them, and return an error if they do not support the call. This includes

the main driver commands as well as all subcalls. Unfortunately, some driver authors didn't feel that the guidelines needed to be followed, and they do random things if given a call they don't know about (some also crash). If this bit (\$0010) is set in a device's characteristics word, the driver indicates that it properly follows the driver guidelines in the GS/OS reference and Device Driver reference. If this bit is not set, drivers will never receive a call that is not documented in the GS/OS reference (except device-specific calls, which are always passed on to the drivers).

Initialization Manager

Fixed a problem where supplying a non-editable volume name for the Format or EraseDisk calls, and the name was not syntactically acceptable to the chosen FST, the system went to the GS/OS version of system death.

Limited the supplied volume name to 32 characters to ensure that the 36-byte buffers used internally are not overflowed.

Changed the default disk name from Untitled to untitled.

System Loader

Filetypes \$0030 to \$003F (all auxiliary types) are now allowed for load files. Filetype/auxiliary type combinations must be assigned by Developer Technical Support, as usual.

System Error Messages

Added error message #66: "The selected AppleTalk connection could not be found. Be sure it is installed and the slot settings in the Control Panel are correct."

Added error message #67: "A driver for the selected AppleTalk connection could not be found. Be sure it is installed and then restart."

Added error message #68 for the RAM5 driver (used on ROM 1 if the minimum and maximum sizes are different).

In Error.Msg, changed "volumes" to "disks" in the message "ProDOS 8 applications can only be launched from ProDOS and AppleShare disks."

SCC Manager

Added more error checking and two new error messages in case the AppleTalk hardware couldn't be found, or no drivers were loaded.

SCSI HD Driver

Added support for the new Apple-defined DStatus subcall. Call \$4000, GetSCSITargetPriority, returns a word indicating the SCSI ID of a given GS/OS SCSI device.

SCSI CD Driver:

Added support for the new Apple-defined DStatus subcall. Call \$4000, GetSCSITargetPriority, returns a word indicating the SCSI ID of a given GS/OS SCSI device.

Changed default command bitmap to match the Apple CD-300, which does not return a command bitmap in the Inquiry call.

Compatibility with Apple CD 300

- With the 6.0.1 SCSI CD Driver, you can read data, but you can't play audio tracks, and you can't take advantage of the drive's ability to read audio tracks as data.
- With the 6.0 SCSI CD Driver, you can read data if and only if a CD was online when you booted. You can't play audio tracks.
- Under P8, the Apple CD 300 works just as well as the previous models, for data. Playing audio tracks is not supported.

SCSI Tape Driver

Added support for the new Apple-defined DStatus subcall. Call \$4000, GetSCSITargetPriority, returns a word indicating the SCSI ID of a given GS/OS SCSI device.

SCSI Scanner Driver

Added support for the new Apple-defined DStatus subcall. Call \$4000, GetSCSITargetPriority, returns a word indicating the SCSI ID of a given GS/OS SCSI device.

Console Driver

Fixed reset_trap to store the correct addresses in the fast-I/O vectors; they don't work in 6.0.

AppleDisk 3.5 Driver

Changed the SuperDrive case of DStatus(get_format_options) to behave better. Now it returns as much data as it can and returns a real transfer count. It used to return a transfer count of zero (and return no data) if the user's buffer was not big enough.

Driver no longer puts 'A', 'B', ... over the last character of its device names during a warm start. So now you can rename your 3.5 devices and not get the names mutilated on return from P8.

AppleDisk 5.25 Driver

During startup we now clear the internal/external slot number in our DIBs as we scan each slot. On return from ProDOS 8, the driver was looking for the Disk II card's ID bytes at the wrong address, like \$CE00 instead of \$C600. There was no problem for the internal Disk Port, only for Disk II cards. The drives did not show

up in GS/OS after visiting ProDOS 8.

Removed driver startup time check for drive being present. For some people, it was deciding they had no drive when they actually did.

RAM5 Driver:

New for 6.0.1. Replaces the generated driver and is much faster. On ROM 1, Minimum and Maximum RAM disk sizes must be set to the same value.

If the RAM disk has never been formatted, we now create an empty ProDOS directory, like the ROM does.

On startup, if RAM5 is an empty Pascal volume, we reformat it as an empty ProDOS volume. This covers a semi-obscure case where you have no 3.5 drives on the SmartPort, but the startup process scans past slot 5, touching the RAM disk before a ProDOS-like environment is set up and fooling it into formatting itself as Pascal. If you have a nonempty Pascal RAM5, we leave it alone.

A2.RAMCard (aka Slinky) Driver

Added format options so if you have 1024K on your card you can choose between 800K and 1024K. Formatting your RAM card to 800K means you can do block copies between 800K floppies and the RAM card (block copies are generally faster than file-by-file copies).

AppleShare FST

The 10-second volume change timer completion routine calls the GS/OS system service vector `os_event`. However, this really isn't legal since we're in an interrupt environment and GS/OS may have been what we just interrupted. Since `os_event` modifies a few variables on the GS/OS direct page, we now save and restore them around the service call. This should fix random \$4x errors during intensive file manipulations.

If we get a network error back from our SPWrite handler, it now loads the protocol error word from its parameter block, rather than the SPCommand one.

Fixed an icky little bug in our panic completion routine. It was incrementing its panic counter without setting its data bank. This caused a semi-random byte in bank \$E1 to be incremented by one each time the panic timer completed. This also meant that our quarter-second interrupt handler never knew when one or more of our volumes were in danger of being lost due to missing tickle packets from the server.

Shortcut the QDStatus and GetWAP calls to save doing one or two toolbox calls on every call to the FST.

If the arrows get drawn on the Super Hi-Res screen, they get UNdrawn there, even if a screen dimmer kicked in & displayed the text screen. (If the SHR screen is off and non-linear, it doesn't touch the SHR screen.)

Fixed volume-changed notification to pass correct device number instead of garbage.

Now we send volume-changed notifications even if there are no folders open on a volume.

Changed JudgeName syntax string to read "Disk names..." instead of "Volume names...."

DOS 3.3 FST

Ignores zero-length filename entries (GetDirEntry does not count them and does not return them). Internally, the FST treats a filename field of all blanks (30 \$A0 bytes) just like a deleted catalog entry.

Added an entry to the end of fcr_to_pblk so that Open will stop when pcount is 15 instead of trashing a variable amount of memory past the end of the parameter block.

FST name (visible in Finder, etc.) is "Apple II DOS 3.3" instead of just "DOS 3.3". No change in file name.

Changed Open, GetFileInfo, and GetDirEntry to return zero for resourceEOF and resourceBlocks fields for non-extended files (when the fields are present). These values were previously undefined, and this was confusing the Finder.

HFS FST

Added a STZ newline_flag to fix bug where Write stops prematurely after transferring 512 bytes, if the most recent Read call stopped because it hit a newline character.

Changed JudgeName string to say "Disk names..." instead of "Volume names...."

Optimized much of the computation-intensive code in the FST. Namely, the B-tree register push/pop routines which use the pseudo stack are much faster, and the multiply and divide routines handle common cases more quickly. An unaccelerated GS with fast hard disks should do large folder copies between HFS volumes in the Finder about 18.4% faster. Other configurations will have less of an improvement.

Pascal FST

Now recognizes RAM disks (device_id \$000E).

HS.FST (High Sierra FST)

FSTSpecific map_enable and set_map_table now post volume-changed notifications for all of our online volumes (High Sierra and ISO 9660).

Changed volume control records to have a Memory Manager handle to the path table, so we can allow path tables larger than 8K (the GS/OS Memory Manager limit).

MS-DOS FST:

New for 6.0.1.

The MS-DOS FST is a read-only implementation, similar to the existing DOS 3.3, Pascal, and High Sierra FSTs.

Added support for resource forks as defined and implemented by the Macintosh PC Exchange Control Panel. This is done by placing the resource fork of a given file into a normal MS-DOS file of the same name as the original file in a subdirectory named "RESOURCE.FRK" at the same directory level as the original file. The "RESOURCE.FRK" subdirectory will never appear to an application during a directory search on the GS, though it will appear normally on an MS-DOS platform. The calls affected are Open, GetFileInfo, and GetDirEntry.

ProDOS FST

Fixed a bug in slow_read that made multiple-character newline mode unreliable. Each time it moved to a new block of data it was forgetting about one more character off the end of the newline list.

ProDOS 8- Version 2.0.3

Updated the Thunderclock year table for 1993-1998.

When we don't have a 65C02, instead of saying "relocation / configuration error" we put a message below the P8 version line so it looks like this:

```
Apple II
ProDOS 8 V2.0.3    06-May-93
REQUIRES ENHANCED APPLE IIE OR LATER
```

Apple II Plus users will not be able to read the lowercase, but they can read the REQUIRES part.

Removed the message "*** Unable to find a ".System" file ***". We always use the Quit code if there is no .System file. The old check was flawed anyway--it was invalid after running the ATINIT file.

Changed copyright message at \$BF40 to start with a \$00, to prevent applications from crashing when they remove the /RAM volume and the device table is full.

Changed NMI vector to be initialized to be BRK, \$40, \$00 rather than JMP \$FF59.

Removed slot 3 drive 2 from the SmartPort remapping table, so the /RAM disk can never be overwritten by a remapped device. (It was already the last resort.)

Clock.Patch (new on SystemTools2)

Supports patching ProDOS 8 2.0.3, just like 2.0, 2.0.1, 2.0.2.

Use 0 for 2000, not 100 (fixed off-by-one error in line 1560).

Applications, Utilities, Other:

Control Panels

AppleShare Control Panel v2.1

Added range check on tracking checkboxes next to items in the volume list. If there were fewer than 4 volumes listed, we were tracking checkboxes that were not there.

Now forces an arrow cursor on closing the Connect As dialog (you could be left with an I-Beam in 6.0).

DC Printer Control Panel

Note that starting with System Software 6.0, DC Printer lets you choose LaserWriter, which is potentially useful for "printing" to PostScript files.

Apple tests and supports the LaserWriter driver using AppleTalk only. If you can get useful results with other configurations, that's great, but you're on your own.

Monitor Control Panel v1.2

Rewrote the control panel in assembly.

Made window wider and realigned the pop-up menus.

Added "Smoother Mouse Cursor" checkbox that sets Battery RAM location \$5F bit 2. It kills the cursor flicker that is especially noticeable with a Video Overlay card or an accelerator. This Battery RAM bit has been present since 6.0, but there was no Apple-supplied human interface to set it.

Namer Control Panel v1.1

Made window 80 pixels wider.

Changed the Help text slightly.

Removed custom listDraw routine, since the standard one now uses DrawStringWidth.

Set bit 30 to ModalDialog for I-beam cursor.

Fixed Rename Printer... button to have a valid key equivalent (Command-R).

NetPrinter Control Panel v1.1

Made window 80 pixels wider (it's like AppleShare now).

The Printer pop-up now has a pop-down triangle.

No longer uses a custom listDraw routine (the standard list draw proc now uses DrawStringWidth).

Removed some dead data (left over from when Namer was not a separate control panel).

Typing a colon or slash in the printer name gives SysBeep2(sbBadKeypress) instead of just SysBeep.

Changed completion routine to properly jump to the completion routine return address (returned by the GetInfo call) rather than assume it can just RTL.

Network Control Panel v1.1

Added keyboard equivalents: "P" for ProDOS, "G" for GS/OS.

RAMDisk Control Panel v1.2

Rewrote the help text, mainly to mention that the "Resize after reset" checkbox is only on ROM 3.

SetStart Control Panel v1.1

New checkbox: "Show icons during startup" (provides a human interface to the Battery RAM bit that disables the ShowBootInfo icons...the bit has been there since 6.0).

New checkbox: "Enable programmer CDAs" (does not show up if ROM 3 Control Panel disable jumper is present). Provides a human interface to the Battery RAM bit that asks for Visit Monitor and Memory Peeker to show up in the CDA menu. ROM 3 checks this bit, and 6.0.1 checks it during boot for ROM 1 machines.

Added a pop-down triangle in the menu. Edited some error messages.

Edited the help to say "This control panel lets you choose what application will start up automatically when you start (boot) your Apple IIgs. The checkboxes let you control whether icons appear while the computer is starting up, and whether Classic Desk Accessories such as Visit Monitor and Memory Peeker appear."

Sound Control Panel v2.1

Added a new request code, srqConvertRelPitch, to convert a relPitch value into a freqOffset.

Sound control panel no longer gets confused on a redundant startup (like if somebody installs a second copy of the Control Panels NDA, giving us a second BootCDEV message).

The Shutdown notify proc now checks a flag properly, so sounds don't get cut off when the system shuts down (in 6.0 they get cut off sometimes, because it accidentally loads the flag value from bank zero instead of the bank the Sound CP permanent code is in).

NDA's and Inits:

Control Panels NDA v2.1

Command-Shift-Esc works while the Control Panels window or a control panel window is in front (with and without Option). — That is, you can use Command-Shift-Esc to bring the main window to the front even if a control panel is already in front, and you can use Option-Command-Shift-Esc to open (or bring to front) the CDevs Finder window when the main window or any control panel window is in front.

Uses new calls OpenResourceFileByID and InvalOneCtlByID to save a few bytes and exercise the new calls.

Now accepts reOpenAction. If Option is down when you open the CP & it's already open, now it opens the CDevs window instead of bringing the NDA window forward.

When overwriting an existing CDev.Data file, the Control Panels NDA now makes sure to unlock it first. The old Control Panel created the file with no Write access, so we were failing to overwrite it (added one SetFileInfoGS call).

When trying to open CDev.Data for writing, if we got an error we were doing a Close with a stale reference number. Fixed.

Added WhooshRect to the main window opening and closing.

Fixed a bug that would trash some memory if there were NO active control panels available in the CDevs folder.

Moved the code that handles color table switching into the new QuickDraw II Auxiliary calls UseColorTable and RestoreColorTable.

Made the color tables not be schizophrenic when you switch between control panel windows in an application using nonstandard colors.

Now checks daylight savings stuff whenever the user closes a control panel. This makes the right thing happen sooner if you turn on the Auto Daylight Savings checkbox in the Time control panel.

When calling HiliteMenu to highlight or unhighlight the Apple menu, passes flag values of \$8001 and \$8000 instead of 1 and 0. This makes the highlighting work right even for a text-based Apple character with a weird color table.

Fixed cpOpenCDev request to work reliably. In 6.0, it did not always work. (The handling of finderSaysBeforeOpen has always worked fine.)

EasyMount v1.1

Loads even if an AppleTalk network is not available, since it's useful locally now.

Added support for folders and applications, and for local disks. Yes, you can make an alias for lots of things now! (You can't make aliases to documents, or aliases to aliases, and you can't do anything helpful with an alias other than open it...like drag things onto it.)

Menu item now reads "Make Alias..." and has the keyboard equivalent Command-M.

On handling a redundant finderSaysGoodbye call, EasyMount does nothing (as required in the 6.0 documentation).

We now accept the srqGoAway request (in case somebody wants to remove or replace us on the fly, with IR or a similar utility).

Added the message "You can make aliases for disks, folders, and applications only."

Fixed srvNameBuff and zoneNameBuf to be large enough to hold 31-character GS/OS strings, so we can now deal with the longest server and zone names.

We now use UseColorTable/RestoreColorTable in QuickDraw II Auxiliary when displaying the password dialog. It swaps in the 640-mode colors so the icon always looks correct.

EasyMount now works correctly in 320 mode. The dialog was redesigned a bit so it works equally well in both modes.

Made srqMountServer work properly with the new 1.1 file format that includes the alias pathname at the end.

Crunched code a little by using the new calls OpenResourceFileByID, HiliteCtlByID, and SetCtlValueByID.

Changed many error messages for clarity and consistency.

Find File NDA v1.1

Command-F opens Find File, if the application doesn't use the Command-F for a menu item. (Find File installs a request procedure and watches for systemSaysMenuKey.)

Excellent improved responsiveness by giving up control when EventAvail shows there is a keypress or mouse click waiting to be processed.

Passes a real auxiliary type to GetSysIcon (was always passing zero, so request procedures were unable to give different icons for files with the same file type but different auxiliary types).

Fixed intermittent \$0201 error on going to P8 after clicking on a found file. It wasn't disposing the \$53xx memory for the bottom list, and sometimes it would

wind up in special memory.

Removed code that truncates disk names for the pop-up menu, since the menu manager takes care of it now.

Removed leading colons from volume names in the pop-up.

Uses the new Control Manager call GetCtlValueByID.

Calculator NDA v1.1

Disables the Undo item when the window is active.

Media Control NDA

Changed version from 1.3a0 to 1.3.

VideoMix NDA

Changed version from 1.3a0 to 1.3.

Finder 6.0.1

User-Level Finder Changes

Changed the About box title and menu item to "About This Apple IIGs". Removed Finder version and engineer names. (You can Get Info on Finder to see the version, which should always match the System version anyway.)

Keyboard Navigation implemented. See Finder Help under "Selecting & Coloring Icons."

When you click on an icon, it starts renaming it only if you click on the *name*, not if you click on the icon part. (Terribly handy for keyboard navigation.)

Removed "File Copy Alternatives" dialog. Option-dragging a file to another location on the same disk now forces a copy rather than a move.

You can now click in the Name, Size, Kind, or Last Modified text in the info bar of a list-view window to change the view.

Icon Info Calculate button has key equivalent of Command-K. You can use Command-period to stop the calculation.

Added the display of SCSI IDs on the "Where" card of Icon Info windows. If the device that the window comes from supports DStatus subcall \$4000 (GetSCSITargetPriority), then the ID will be displayed after the device name.

Dragging control panels, desk accessories, drivers, Finder extensions, fonts, file system translators, sound files, initialization files, and tools onto the System folder icon on the boot disk now does "magic routing" after telling the user. (Font files get special treatment only if bit 15 of their aux type is clear, so that DTS can make

font filetype assignments for files that do not belong in the Fonts folder.)

The rubber-band icon selection now actively highlights and unhighlights icons, instead of waiting until you release the mouse button.

Fixed rename with an Icon Info window open on the target file to work for regular files. In 6.0 it only redrew the Icon Info name for devices and folders.

We now dynamically resize the long product name static text control rectangle on Icon Info cards to account for the existence of the "Inactive" check box. We also set the new control flag bit of the same control to use DrawStringWidth to truncate and compact to fit the area.

When running the Finder from an AppleShare file server, the Finder now properly draws the server volume name in the writing a bundle status dialog if the @:Desktop file is being written to. It used to always be blank.

Fixed a bug in the info card recalculation code. If you had any Icon Info windows open when you emptied the trash, the sizes of files would be updated to match the disk that they came from. Now only disk Icon Info windows get updated when the trash is emptied.

Command-Option-O (tunnel open) no longer closes your current window if there was nothing to open.

Changed window-closing code so it always selects the icon the window closes into. In 6.0 it selected the icon only if there would be no other selection, which was confusing.

New untitled folders now have lowercase names (like untitled.a).

Changing the Hide Invisible Files preference updates all windows, not just the front one (it was trying to update them all, but it was getting confused after the first one).

Creates Trash icon and window even if device #1 fails to mount (you could crash Finder 6.0 by booting from floppy, moving the floppy to some other drive, re-entering the Finder with the boot device empty, and then trying to leave the Finder).

Re-fixed the "trash icon turns into the icon you dragged into it" bug that was allegedly fixed in 6.0. It was *almost* fixed in 6.0.

Changed the "need an FType file" message to say "...the Icons folder on your startup disk" instead of "the *:Icons folder."

About box numbers now read "zero" when they would be "0", as in "OK". For example, boot with Inits/DAs off & see "zero K" for DAs.

Kind strings in list views now get compressed if necessary (the Finder draws them with DrawStringWidth instead of DrawString).

Files with HFS file type 'APPL' now show up with kind string "Macintosh application".

Inactive files no longer "miss" their specific Kind string & get the generic-filetype

kind. (When mapping a filetype/auxtype to a kind string, if the match is an any-auxtype entry that can be Inactive, and bit 15 of the auxtype is in fact set, the Finder repeats the search with bit 15 cleared, getting a kind string that is at least as descriptive.)

Added oneDocs in the Finder's resource fork for various Driver auxiliary types, which now show up as Kinds: GS/OS Device Driver, AppleTalk Driver, Media Control Driver, Printer Driver, Printer Port Driver, MIDI Driver.

Added an icon for DiskCopy files.

Dragging a different-sized disk (or option-dragging a same-sized disk) onto a read-only-filesystem disk (like MS-DOS) now correctly complains that you can't do that.

Improved the Clipboard display (uses the new ShowClipboard call in the Scrap Manager).

The "Finder requires 6.0.1 and 300K" message now shows a hex error code (from StartUpTools or other calls during startup, like LoadResource).

Erase and Initialize on a disk now update the icon's file system. Fixes bugs like: Erase a DOS 3.3 disk to be ProDOS; when you drag something onto the disk, it still complains that the disk uses a read-only file system.

Control works with Command-W and Command-Option-W. (Control reverses the save-finder-data setting.)

Changed a lot of window parameter blocks to allow windows to update in the background (like behind an AlertWindow).

Changed preference to read "Check 5.25" drives on Finder start" (key equivalent is now 5). It used to say "When starting up, check 5.25 drives", but this was confusing (could be misunderstood as affecting what device the system boots from).

Finder no longer gets confused about the file system of an icon that you drag to the desktop. Fixes bugs like Locked boxes showing up for HFS folders on the desktop, and not being able to launch ProDOS 8 applications that were just dragged onto the desktop.

Fixed a bug when shortening a disk name (rename RAM5 to RAM, for example). The Finder would get really confused about the pathnames of existing icons on the disk that you renamed.

DrawDesk uses full-screen rectangle, not just below the menu bar. This way HideMenuBar works as expected in the Finder.

Fixed up I-Beam handling (new code is smaller). Fixed bug where IBeam could mistakenly appear when the icon being renamed was in a window but scrolled outside the visible area, and the mouse was over the desktop.

When you change the system date/time format (for example, using the Time Control Panel), the Finder now bothers redrawing all directory/trash windows that have a

Last Modified column. (It used to notice the change but not force a redraw, so windows could get half-updated.)

Saved 1800 bytes by making menu resources string-based instead of template-based (so far they're rCStrings). This saves disk space and load time and reduces the handle count.

Developer-Level Finder Changes

Finder uses Volume Changed events to notice when windows need to be updated because of GS/OS calls made by anything other than the Finder. The `tellFinderAboutChange` call now does nothing, since the Finder now notices changes without special help. Note that the Finder does not notice when it should make changes to icons on the desktop (not even in 6.0 with `tellFinderAboutChange`).

Changed `tellFinderSetSelectedIcons` to not flicker an icon if it is the sole selected icon and we've been asked to highlight it again.

When Finder sends out multiple `finderSaysBeforeOpen` requests (when a group of icons are opened), the modifiers are now correct for all of the requests, not just the first one. In 6.0, the modifiers were accidentally zero for all requests other than the first.

`tellFinderGetSelectedIcons` (extended) now returns the icon height and width. This was broken in 6.0 (the workaround in the 6.0 documentation sees that the Finder version is not 6.0 and automatically does nothing).

Fixed the `oneDoc` match types `matchCreateDateTime` and `matchModDateTime`. They did not work in 6.0.

Finder no longer forgets what menu title to unhighlight if an Extras menu handler calls `tellFinderMItemSelected` (even with no-highlight).

We now send `finderSaysBeforeOpen` any time we launch an application, not just when the user explicitly launches one. This means that you can now properly trap the Finder any time it was about to quit to another application. (Finder 6.0 neglected to send a `finderSaysBeforeOpen` when the user chose the application using Standard File.)

Finder accepts the `srqQuit` request, handling it just like a `tellFinderShutDown` with a selector of `kQuit`.

When matching an icon by filename and using a leading "*", the "*" can now match zero characters (it works like 5.0.4 again). In 6.0, a leading wildcard accidentally required at least one character to match.

When matching an icon by filename, the case of the string in the Icon file or in the `rBundle` resource no longer matters. Previously, a string with a leading wildcard (like "*PAINT") would only work if all letters were uppercase, regardless of the case of any actual files being matched against.

Added a tick count parameter (long) to `finderSaysKeyHit`. The parameter count is now three.

The parameter count for `finderSaysBeforeOpen` and `finderSaysOpenFailed` is now seven. (It was accidentally only 6 in 6.0, but all seven parameters were there.)

For convenience, `finderSaysIdle` now passes the idle ticks (as would be returned from `askFinderIdleHowLong`) as the `dataIn` value.

Changed `finderSaysHello` `pCount` to 4 and added a word of flags, none of which are presently useful.

Fixed `tellFinderSetSelectedIcons` to handle bit 31 correctly (the 6.0 Finder documentation says not to set the bit; now it works as originally intended, deselecting all icons before selecting the specified ones).

`tellFinderRemoveFromExtras` validates the item number for `$Exxx`, returns error `fErrFailed` if invalid. (If you pass `itemID 0` to Finder 6.0, it happily removes the first (remaining) item from the Apple menu!)

Implemented the `geekPrefs` option for Shut Down default. Set the low two bits of the `X2` word to 0, 1, or 2. (This word was semi-documented with 6.0, but it didn't do anything.)

Fixed `tellFinderAddBundle` to use `GetIndResource(1)` instead of 0 when you pass zero to grab the first available `rBundle` (this case always failed with an error in 6.0).

`tellFinderSpecialPreferences` now returns a valid `finderResult` (\$0000). In 6.0, it was returning the `dragFlag` value as the `finderResult`.

Internal Finder Changes

When we're network-booted, we now load the Desktop file from the user prefix ('@'). In 6.0, the Finder wrote `rBundles` to the Desktop file there, but it forgot to read them.

Removed the new-for-6.0 server-specific copy code, because the code that built the destination hierarchy tree was whacked in the head. This means that duplicating files or copying from disk-to-disk on a single server now reads and writes file data instead of telling the server to do it.

`LoadExtensions` now tries to create `@:FinderExtras`. This way the folder already exists in case you drag a Finder Extension into your System folder. (If you're booting from a network, magic-routing a Finder Extension is not helpful, because it will try to route into your `*:System:FinderExtras` folder instead of `@:FinderExtras`.)

Fixed `CompareVols` to be case insensitive in its comparison. Internally Finder will always create icon objects and window names and pathnames in the same case as they have on disk. However, if a window was opened via `tellFinderOpenWindow` that used a different case, moving icons from/to that window could previously result in files being copied instead of being moved.

Fixed a fatal bug in the Icon Info size calculation code. If it couldn't read the current size from the device for whatever reason, the stack became unbalanced.

Changed `keyDown` to ignore the event if `systemSaysMenuKey` ate it (if the "what"

of the event is zero). For example, when Command-Ctrl-Esc opens the Control Panels NDA, the Finder does not accidentally process the Esc keypress and deselect all icons.

Fixed the problem where a server disconnecting unexpectedly would make the Finder hang if a server window was the current selection plane.

During tellFinderSetSelectedIcons, we now bring each icon to the head of the linked list as it is drawn. This fixes cosmetic problems when icons overlap and get partially redrawn.

Finder can now deal with 128 devices being online as opposed to the old limit of 64.

Fixed four incorrectly-coded 32-bit additions. There were problems when certain data structures crossed bank boundaries.

Changed SizeToString to do "zero " instead of "0" (affects info bars, icon info cards, & much more).

When Finder gets error \$46, \$51, \$5A, or \$2D from a Volume call on a newly-inserted disk, it no longer sets the hdiUnreadable bit to HandleDiskInsert (that bit is only set for a real I/O error).

Knowledge of what filetypes can be Inactive is no longer in the Finder. It comes from the Filetype Descriptor files instead. Flag bit 0 in an index entry for an any-auxtype entry means files of that type can be inactive.

Finder now uses Filetype Descriptor files even if some of flag bits 0..7 are set.

No longer calls SelectWindow on a non-window when you click on a desktop icon or start rubber-banding on the desktop.

The tellFinderGetWindowInfo call has a special case for the Clipboard window, which is now created by the system. tellFinderGetWindowInfo pretends it's a Finder window and gives you information about it anyway.

Saved a few bytes by removing a redundant MessageCenter call for deleting message \$11, the long-pathnames-to-launch message (the toolbox automatically deletes this message when the Finder deletes message 1, the pstring-pathnames-to-launch message).

Optimized main segment (AND #\$7FFF becomes ASL LSR; use Zero constant on direct page)

Uses new Line Edit call LEClassifyKey to determine what control characters to let through for editing when renaming an icon (saved a few bytes; no user-visible change).

Fixed the error-exit path when tellFinderOpenWindow gets an error from ExpandPath.

Various miscellaneous optimizations, including reserving one word of direct-page space to always hold a zero. Replaced many pea 0 with pei <zero, lda #0 with lda

<zero (saves one byte each).

FType.Apple

Latest database from Developer Technical Support. Includes lots of new descriptors for shipping products.

Note that the 6.0 Finder does not appreciate finding the 6.0.1 FType.Apple file in your boot disk's Icons folder.

Header now has version \$0101 and flags \$8001.

Bit 0 is set in generic filetype entries for filetypes that can be Inactivated.

Teach v1.1

Requires System 6.0.1 to run.

Removed 128K limit on document size.

Changed default window name to "untitled".

Undo has Command-Z key equivalent.

Added Show Clipboard to Edit menu (always enabled, so Edit menu no longer dims). This simply uses the new ShowClipboard call in the Scrap Manager.

Edited several messages to say "disk" instead of "volume."

Updated StartUpTools MasterSCB to use shadowing and fastPort.

Accepts srqOpenOrPrint (re-checks the MessageCenter for files to open or print) and srqQuit (asks Teach to quit later, from its main loop; the user can cancel if there are unsaved files).

The window update routine now has flag bits set so it can update behind alerts.

Checks GS/OS version first on startup, so it doesn't get weird "requires Tool003" message during StartUpTools under 6.0.

Tries to re-allocate safety handle before complaining that you're critically low on memory.

The \$7001(1) resources on Teach documents were being created with 4 extra bytes on the end. Now they're 12 bytes, like they're supposed to be.

Now calls InitCursor when closing a window and when quitting, so you never see an I-beam cursor on the desktop (ick), even briefly.

Uses the Caution icon in the save-changes dialog. Reworded the alert to be more standard.

Installer v2.1

Easy Update and the “*System 6...” scripts always install EasyMount, since it is now useful even without a network.

Added RAM5 driver to Easy Update. It will install if you have a RAM disk at the time (Maximum size non-zero). It will also update it if a copy of the driver exists, even if the maximum size is currently zero.

When installing to a non-ProDOS volume (the available space is not determinable) the thermometer is hidden. This is preferable to one that confuses the user by never updating.

New folder name defaults to "untitled" instead of "Untitled"

Changed “Stand by...” to “Please wait...” in two messages.

Added MS-DOS FST to Easy Update (it just re-copies the FST if you already have it).

Windows use lined title bars instead of black.

Saved 16K of disk space by changing the link options.

Installer Scripts

Added “Read.Me”, “Shortcuts”, and “Whats.New” (new file) to the Teach script.

The scripts “System 6: Hard Disk or FDHD” and “System 6: No ‘Start’ Update” now install EasyMount.

New RAM5 driver script: The script “Drive: RAM5” installs the RAM5 driver, which improves performance when using the “RAM5” RAM disk.

The scripts “System 6: Hard Disk or FDHD” and “System 6: No ‘Start’ Update” no longer install EasyAccess (but they update it if it's already present).

Added script “File System: MS-DOS FST” for MSDOS.FST. It displays its help as an alert when you try to install it: The script “File System: MS-DOS FST” installs the files needed for reading MS-DOS disks. Note: To read MS-DOS disks, you need an Apple SuperDrive connected to an Apple II SuperDrive card, or other hardware that can read MS-DOS disks.

Changed Help message for DOS 3.3 FST to specify Apple II DOS 3.3 instead of just DOS 3.3, to avoid confusion with MS-DOS.

Quick.Logoff—changed “server volume” to “shared disk” in help message.

synthLAB—changed “startup volume” to “startup disk” in help message.

:SystemTools2 - Disk

Read.Me file: Rewritten for 6.0.1.

Whats.New file: New file listing the changes in 6.0.1 for the user.

Shortcuts file: lots of changes.

Added CLOCK.PATCH (Applesoft program for patching ProDOS 8 Thunderclock driver).

Apple II Setup disk v3.0

To boot Apple IIGS System Software 6.0.1 from an AppleShare file server, you must install Apple II Setup disk v3.0 onto the server. Earlier versions of Apple II Setup do not work with 6.0.1.

Includes Logon version 1.5 and ProDOS 8 2.0.3.

In Logon, added a "Looking for AppleTalk zones" message. Also, keyboard navigation is now insensitive to bit 7 of the first characters of strings in the lists (since they are already sorted bit-7 insensitively).

The Apple II Setup disk is no longer bootable. There was no longer room to put a current Macintosh System Folder and all of the installed parts on one disk. Installer version 3.2 is used, so you should just double-click the Installer application.

Note: For AppleShare 2.0 users, you'll need to get to the Finder to do this; just start up from a disk that includes the Finder (such as your System 6.x System Disk, or the AppleShare Admin disk). Hint: You can then select your server's boot disk and pick "Set Startup..." from the Special menu; set it to start up with Finder (or MultiFinder) and then restart. This way, you'll boot into the Finder from your hard disk and won't have to swap floppies. When you're done, you can set startup to the "AppleShare File Srvr" program in the Server Folder.

100

100