Technote 1176

Mac OS 9

Document revision 5

This Technote discusses changes and corrections in the next generation of Mac OS: Mac OS 9. This system follows Mac OS 8.6 and contains several enhancements and revised features.

All developers interested in creating products compatible with Mac OS 9 will want to review this document.

Contents

Revision Listing

4/24/2000 (revision 5):

- The PPC Toolbox section now shows how to send Apple events over TCP/IP.

3/13/2000 (revision 4):

- Updated Thread Manager section to include specific reference to Mac OS 8.1 for the `kPreemptiveThread` bug fix.

12/23/1999 (revision 3):

- New `ChangeDragBehaviors` API defined in the Drag Manager section.

12/6/1999 (revision 2):

- New information highlights:
  AppleScript 1.4 (aliases in Scripting Additions folder), Font Manager (suitcase limit increase), new HTMLRenderingLib section, new Power Manager 2.0 section, PPC Toolbox (port number information), new Text control panel section, and new Text Encoding Converter 1.5 section.

- Updated sections:
  AppleScript 1.4 (aliases in Scripting Additions folder, mount volume scripting addition note), AppleShare Client 3.8.5 (login security), ColorSync 3.0, File Manager (extents and catalog overwrite prevention), Finder 9.0 (icon positioning), Font Manager (suitcase limit increase), Hardware Requirements (boot from CD requirement), LaserWriter 8 version 8.7 (PostScript filters), Mac OS USB (USBPending fix), Memory 8.1.1 control panel,

Multilingual Text Editor 1.1, Multitasking/Multiprocessing 2.1, Open Transport 2.5 (kOTPortNetworkChange), PlainTalk 2.0.2 (version numbers), PPC Toolbox (port number information), and StdCLib 3.5 (alignment of vector data in printf).

- Other changes include updated documentation links and some cosmetic revisions.

Back to top

Hardware Requirements

Mac OS 9 can be installed on any Macintosh that originally shipped with a PowerPC processor that has at least 32 megabytes of physical RAM and a minimum of 40 megabytes of logical RAM (virtual memory is set to 64 by default if there is less than 64 megabytes of physical RAM installed).

Mac OS 9 has been tested and qualified by Apple to run on Apple PowerPC-based Mac OS computers. (Mac OS 9 is not tested or qualified by Apple for use on Mac OS-compatible computers.)

For installation on 6100, 7100, and 8100 models, the computer must be booted from the Mac OS 9 installation CD.

Back to top

Installation

Install Mac OS 9

The "Install Mac OS" application program requires System 7.1.2 or later to run, allowing users to install the full CD-ROM system installation of Mac OS 9 over System 7.1.2 or later. Users running Mac OS 7.5.1 or earlier are required to boot from the CD to perform an installation.

- The installer script for Mac OS 9 supports the concept of "Feature Sets" that allow users to select different kinds of installations beyond "for this computer" and "universal." These new installation types are tailored to suit the needs of particular interest groups such as, for example, Education, Business, or Gaming.

Back to top

Finder 9.0

The Finder provides a graphical interface for the user along with other Apple event-based services for applications utilizing the file system.

- Adds support for copying files over 2GB in size (up to a theoretical limit of 2^{63} bytes in size) and displaying information about them in **Get Info** windows.

- Long unicode file names are now preserved in file copies.

- Volumes which don't support the Desktop Manager (typically floppy disk volumes) with large "Desktop" files now mount much faster.

- The Finder creates an invisible folder called "TheVolumeSettingsFolder" on each writable disk, which it uses to keep track of information about open window positions, desktop printers, and other items.

- Provides some support for Carbon application bundles (folders containing one or more support files). In this implementation of packages, there are a few points to keep in mind:

    - When the Finder attempts to open an item inside of a package's folder, it sends an `'odoc'` event containing a list of documents to open. The difference here is the document list is stored in an optional parameter with the key `'fpdl'` (Finder package document list).

    - In the Finder interface, packages behave identically to applications even though they are stored as folders. The internal contents of packages are hidden from the user.

    - If an alias is created to an item in a package (say, while using an earlier system version) and the user attempts to open the alias, then it will be passed to the package's application using an `'odoc'` event with an optional `'fpdl'` parameter (even if the application's `'BNDL'` does not list the file type).

- An **Encrypt** item has been added to the File menu. When selected this item will launch the "Apple File Security" utility, passing the selected documents to it for encryption. Files encrypted in this way can be decrypted by double-clicking on their icons. Note: Apple File Security cannot encrypt a folder (even if it is a package).

- The Mac OS 9 Finder does not allow icon positioning by way of AppleScript for icons appearing in icon views where either the "keep arranged" or the "snap to grid" option is turned on. If a script attempts to set the position of an icon in such a view, the Finder will return error code number -15274 (window wrong type).

Related Materials:

Technote TN1188, "Packages in Mac OS 9"
Technote TB09, "Finder Flags"
Technote TB42, "Finder Icon Positioning and File Initialization"
Technote TB535, "Finder Q&As"
Technote TN1126, "Checklist for Building Applications and Extensions"
Icon Services Documentation
The Finder Interface chapter of *Inside Macintosh: Macintosh Toolbox Essentials*
Inside Macintosh: AppleScript Finder Guide
Anderson, G. Scripting the Finder. Develop 20

Back to top

Apple Help 1.5

Apple Help provides system-wide instructional help services for virtually all aspects of Mac OS. The Apple Help technology has two main components: Help Viewer and Apple Guide.

Apple Guide 2.5

In Mac OS 9, Apple Guide is used to coach users through sequences of commands required to perform a task.

- For both versions 2.3.1 and 2.4.1 of Apple Guide, the `Gestalt` version selector for Apple Guide returns `2.3.1`. For Apple Guide 2.5, the `Gestalt` selector `'ag_v'` returns `2.5.`

Help Viewer 1.5

Help Viewer is a lightweight HTML rendering application that allows users to browse online help materials conveniently. Help Viewer 1.5 provides the following new features:

- The ability to search locally for Internet-based Help content, and then launch a web browser to view it.

Related Materials:

Apple Help SDK

Back to top

Apple Data Security 2.0

Apple Data Security introduces security features to the Mac OS. New security features include facilities for signing and verification of file signatures, password maintenance, and file cryptography.

Apple Code Signing

Apple Code Signing is a new technology in Mac OS 9 that allows applications, plug-ins, and content to be signed by developers. Apple Code Signing Certificates assure your customers of your identity and the integrity of your products. For more information, please see the Mac OS Security and Cryptography Web site.

Apple File Security

Apple File Security provides cryptographic facilities. Users can use this application to either encrypt a file with a password or decrypt an encrypted file. The Finder's **Encrypt** menu item is dependent upon the Apple File Security application residing within the Applications folder. Apple File Security does not have the ability to compress and encrypt a folder. This precludes it from encrypting a package.

Cryptographic Message Syntax (CMS)

CMS is a high-level digital signature API. For more information about CMS, refer to the Cryptographic Message Syntax Standard documentation provided by RSA Laboratories, and for information about the APIs provided in Apple's implementation, refer to the documentation at the Mac OS Security and Cryptography web site.

A beta version of an API for accessing CMS has been introduced with Mac OS 9.

Keychain Access 2.0

The Keychain provides a secure repository for passwords. After entering a master password, any of the other passwords stored in the Keychain are automatically available to application programs, saving the user the need to remember multiple passwords. Keychain 2.0 supports the following features:

- Multiple open Keychains
- Storing cryptographic keys and certificates
- Integration with Multiple Users

Related Materials:

The Mac OS Security and Cryptography web site

Back to top

Carbon Support in Mac OS 9

Carbon is the set of programming interfaces for use in Mac OS applications. Carbon includes about 70 percent of the legacy Mac OS APIs. With Mac OS 9 comes the first installation of the Carbon APIs. The following sections summarize Carbon APIs available in Mac OS 9.

Carbon Control Manager

The Carbon Control Manager introduces some changes to existing Control Manager APIs. These new services include:

- New routines `GetControlID` and `SetControlID` have been added. These routines allow controls to be referenced by ID number.
- A new control definition packaging model has been introduced in which controls are stored as CFM Code Fragments rather than as standalone code resources.
- It's now possible to write a control definition to support contextual menus.
- Attributes have been added to control properties. Control properties can be marked as persistent so their data will be saved when a control is flattened.
- Six new Control Manager APIs have been introduced that allow controls to support the drag-and-drop feature. This new feature can be turned on either on a per-control or on per-window basis.

Related Materials:

Q&A TB39, "Appearance Versions"
The High Level Toolbox Technote Collection
The High Level Toolbox Q&A Collection
The Control Manager Reference chapter of *Inside Macintosh: Mac OS 8 Toolbox Reference*
The Control Manager chapter of *Inside Macintosh: Macintosh Toolbox Essentials*
Inside Macintosh: Macintosh OS 8 Human Interface Guidelines
The Appearance SDK

Back to top

Carbon Menu Manager

The Carbon Menu Manager introduces some changes to existing Menu Manager APIs. These new services include:

- Menus are now stored as opaque data.
- Automatic handling of the Apple Menu.
- New APIs have been added for defining custom menu definitions.

Related Materials:
Carbon Menu Manager Documentation (coming soon)

Back to top

Carbon Printing Manager

The Carbon Printing Manager provides an entirely new set of APIs for applications to use for generating printed output. These new services include:

- New APIs or managing virtually every aspect of the printing process.
- Extended printer settings structures for storing more user settings.

Related Materials:
Carbon Printing Manager Documentation

Back to top

Carbon Scrap Manager

The Carbon Scrap Manager introduces new APIs for accessing the scrap. These new services include:

- The ability to provide "promised" scrap data, allowing applications to defer the creation of data until if and when it is requested. This is a concept similar to the Drag Manager's "promised" flavor model.

Related Materials:
Carbon Scrap Manager Documentation (coming soon)

Back to top

Carbon Window Manager

The Carbon Window Manager extends and enhances the existing Window Manager APIs, and introduces several new and interesting window management facilities to the Mac OS. These new services include:

- Window Classes and Layering
- New APIs for defining custom windows

Related Materials:
Carbon Window Manager Documentation (coming soon)

Back to top

CarbonLib 1.0

CarbonLib is a shared library that allows Carbon-based applications to run in Mac OS 9. Carbon applications link against CarbonLib instead of linking to InterfaceLib, etc. as CFM-based applications do today.

- CarbonLib 1.0 is new for Mac OS 9.

COMPATIBILITY NOTE
CarbonLib 1.0 with only load with Mac OS 9. It will not load with earlier versions of the Mac OS, nor will it load with later versions of the Mac OS.

Related Materials:
Carbon online documentation and resources

Back to top

CoreFoundation 1.2

CoreFoundation extracts various APIs and services from Mac OS X's Cocoa APIs and makes them available to multiple stacks on multiple platforms using C APIs and pure C implementation.

- String type to provide a unified and simple way to deal with Unicode strings. Easy ways to convert between these and Pascal and C strings in various 8-bit encodings are provided.
- Additional collection classes (Dictionary, Array, Data) that, along with String, provide "property list" functionality.
- XML support for reading and writing property lists.
- Abstractions for Date/Time along with some utilities.
- Preferences support for reading/writing user preferences that can be per user and/or per app.
- Bundle APIs to manage Carbon application wrappers containing executables and arbitrary localizable resource files.

COMPATIBILITY NOTE
For future compatibility, developers should begin using these string types in their software for all international string data.

Related Materials:

Core Foundation Documentation (coming soon).

Back to top

Summary of New Carbon APIs

The following summarizes new APIs introduced by CarbonLib 1.0 that is installed as a part of Mac OS 9.

- Apple event Manager
  - `AEDesc.dataHandle` is opaque
  - `AEDesc` accessors
- Alias Manager
  - New APIs for resolving aliases without displaying a UI
- Appearance Manager
  - `GetThemeMetric`
- Control Manager
  - Custom control definition support using `ProcPtrs` instead of code resources
  - Control property attributes
  - `ControlRecord` is opaque
  - Control accessors
  - Control IDs
  - Control definitions may now support:
    - Contextual menus
    - Drag Manager
    - Setting the cursor
- Dialog Manager
  - Editing a dialog item list
  - `DialogRecord` is opaque
  - Dialog accessors
- Event Manager
  - New APIs to get the current mouse position and keyboard modifiers
- Icon Services
  - Icon file registration
- List Manager
  - Custom list definition support using `ProcPtrs` instead of code resources
  - List accessors
- MacHelp Manager
  - New context-sensitive help API that replaces Balloon Help - not recommended for use yet in CarbonLib 1.0
- Memory Manager
  - New APIs for `Ptr`, `Handle`, and `heap` validation.
- Window Manager
  - Custom window definition support using `ProcPtrs` instead of code resources
  - Ability to change attributes of an existing window
  - Window property attributes
  - Window iteration by class
  - `WindowRecord` is opaque
  - Window accessors
- Menu Manager
  - Custom menu definition support using `ProcPtrs` instead of code resources
  - Menu property attributes

    - Menu and menu item attributes
    - Mass menu item enabling and disabling
    - `CommandID`-based access to menu items
    - Standard font menu
    - `MenuInfo` is opaque
    - Menu accessors
- QuickDraw
    - `GrafPort`, `QDGlobals`, and `Region` are opaque
    - `GrafPort`, `QDGlobals`, and `Region` accessors
- Resource Manager
    - Resource chain manipulation APIs
- Scrap Manager
    - All new API that replaces the classic Scrap API
- String Comparison
    - `TypeSelect` APIs
- `TextEdit`
    - Accessors for some `TextEdit` low-memory globals
- Text Utilities
    - `p2cstrcpy` and friends

Back to top

Inside the System File

The System file contains code and resources for running the computer. This section describes new features, changes, and bug fixes in the System file. The functionality provided by the System file is always present under Mac OS 9, even when the shift key is held down at system startup.

ATSUI (Apple Type Services for Unicode Imaging) 1.2

Unicode is a character set which encodes the characters of many languages in a flat 16-bit number space. ATSUI extends the QuickDraw API and makes it possible for Mac OS applications to draw Unicode text. ATSUI provides both low-level services for drawing Unicode text as well as much of the high-end typographical control previously provided by QuickDraw GX.

- Native support for all flavors of Postscript Type 1 fonts has been added. Type 1 font formats supported include:
    - sfnt unimaster
    - sfnt multimaster
    - sfnt CID (primarily 2-byte)
    - LWFN + FOND unimaster
    - OCF (Original composite font for double-byte scripts)
    - naked CID
- Memory requirements for ATSUI have been reduced.
- Breakline is now a linguistic break based on the unicode utilities.
- Improved performance breakline operation is faster, along with faster layout.
- Added tab character handling.
- Many new features have been added to the APIs.
- Bug fixes.

Related Materials:
Technote TN1102, "Mac OS 8"
Technote TN1121, "Mac OS 8.1"
Technote TN1142, "Mac OS 8.5"
Technote TN1163, "Mac OS 8.6"
Inside Macintosh: Text Encoding Conversion Manager
The Text Encoding Converter 1.5 SDK
The ATSUI SDK

Back to top

Code Fragment Manager 68K (CFM-68K) 4.0

The 68K Code Fragment Manager (CFM-68K) is the 68K implementation of the Code Fragment Manager. The 68K Code Fragment Manager is the part of the Mac OS on 68K machines that loads 680x0 code fragments into memory and prepares them for execution. CFM-68K is not appropriate for use on PowerPC machines.

- Several CFM-68K fragments have been removed from the system. As Mac OS 9 only runs on PowerPC-based systems, these are no longer needed.

Related Materials:

Technote TN1084, "Running CFM-68K Code at Interrupt Time: Is Your Code at Risk?"
Technote TN1077, "Calling CFM Code From Classic 68K Code, or There and Back Again, A Mixed Mode Magic Adventure"
The CFM-68K SDK

Back to top

Communications Toolbox

The Communications Toolbox provides an API for basic communications and connection services.

- The Communications Toolbox Resource Manager (CRM) was reporting the incorrect maximum allowable serial port speed. In Mac OS 9, the CRM has been revised so it reports the correct maximum allowable port speed.

Related Materials:

Apple Computer, Inc. *Inside the Macintosh Communications Toolbox* . Addison Wesley
Communications Toolbox Online Documentation

Back to top

Device Manager

The Device Manager provides a programming interface for communications between applications and device drivers (typically code that communicates with particular hardware devices, although some device drivers do not actually drive devices).

- Calls which have an asynchronous variant (`_Control`, `_Status`, `_Read`, `_Write`) are now accessible synchronously from MP tasks. To test for the availability of this functionality, developers can use the `gestaltMPCallableAPIsAttr Gestalt` selector.

Related Materials:

The Multitasking/Multiprocessing Section
The Devices Technote Collection
The Devices Q&A Collection
The Device Manager chapter of *Inside Macintosh: Devices*
Designing PCI Cards & Drivers
Minow, M. The New Device Drivers: Memory Matters. Develop 24
The PCI DDK

Back to top

Dialog Manager

The Dialog Manager manages user interactions with dialogs. New features in the Dialog Manager include:

- Carbon applications no longer have access to a dialog's item list handle. New routines have been added so Carbon applications can edit items in an item list.

Related Materials:

Q&A TB39, "Appearance Versions"
The High Level Toolbox Technote Collection
The High Level Toolbox Q&A Collection
The Dialog Manager Reference chapter of *Inside Macintosh: Mac OS 8 Toolbox Reference*
The Dialog Manager chapter of *Inside Macintosh: Macintosh Toolbox Essentials*
Inside Macintosh: Macintosh OS 8 Human Interface Guidelines
Franke, N. Multipane Dialogs. Develop 23
Lavoie, M. & Johnson, B. KON & BAL's Puzzle Page: AppendDITL Apoplexy. Develop 29
The Appearance SDK

Back to top

Drag Manager

The Drag Manager supports drag-and-drop operations between windows and applications.

- A new API has been introduced allowing applications to enable or disable various Drag Manager behaviors during particular drag and drop operations. The new routine is shown in Listing 1.

```
OSErr ChangeDragBehaviors(
     DragRef theDrag,
     DragBehaviors inBehaviorsToSet,
     DragBehaviors inBehaviorsToClear);
```

  - `theDrag` is the drag reference that the behavior changes should be applied to. The specified change in drag behavior will be used the next time `TrackDrag` is called.
  - `inBehaviorsToSet` contains a set of flags indicating what behaviors should be enabled for the drag reference.
  - `inBehaviorsToClear` contains a set of flags indicating what behaviors should be turned off for the drag reference.

Listing 1. `ChangeDragBehaviors` declaration.

`ChangeDragBehaviors` allows callers to adjust some properties of the user interaction that will occur for a particular drag reference the next time the drag reference is passed to TrackDrag. In Mac OS 9, `ChangeDragBehaviors` can be used to turn off or on the zoom-back animation that occurs when a drag and drop operation is rejected by the destination. To turn off zoom-back animation for a drag reference, provide the `kDragBehaviorZoomBackAnimation` flag in the `inBehaviorsToClear` parameter. To turn zoom-back animation back on for that drag reference, call `ChangeDragBehaviors` again specifying the `kDragBehaviorZoomBackAnimation` in the `inBehaviorsToSet` parameter.

To specify that no behaviors are to be set or cleared, provide the `kDragBehaviorNone` constant in either the `inBehaviorsToSet` or the `inBehaviorsToClear` parameter.

COMPATIBILITY NOTE
Developers using the `ChangeDragBehaviors` symbol and linking against newer versions of DragLib who would like to have their products run with previous versions of the system software should weak-link against the `ChangeDragBehaviors` symbol. Unless this is done, the Code Fragment Manager will refuse to launch applications using `ChangeDragBehaviors` when an older version of DragLib is being used. Developers weak-linking against the `ChangeDragBehaviors` symbol in DragLib should check to ensure that the routine is defined before calling it.

Related Materials:

Technote TN1085, "Using the Drag Manager to Interact with and Manipulate File System Entities"
The Drag Manager Online Documentation
The Drag Manager SDK

Back to top

File Manager

The File Manager provides services for storing and retrieving disk-based information. New features for the File Manager include:

- The HFS+ API has been introduced with this version of the File Manager. It includes support for large files (forks over 2GB - up to a theoretical limit of 2^{63} bytes in size), and long Unicode names. Named forks are supported if they are supported by the underlying file system implementation (none of the file systems included with Mac OS 9 support forks beyond the resource and data forks; however, it is possible for a FSM plug-in to implement support for this feature).
- The maximum number of open files has been increased to 8169.
- Parameter block calls which have an asynchronous variant, and high-level calls that are based on such calls, can now be made synchronously from an MP task. Calls which are always executed synchronously, such as `PBMountVol`, cannot be called from MP tasks.
- The File Manager has been revised so it is no longer possible for malfunctioning software to accidentally overwrite the catalog or extents overflow files on HFS and HFS Plus volumes.
- The File Manager now supports a new error code being returned from disk drivers, `driverHardwareGoneErr (-503)`. Disk drivers should return this error in response to an I/O request to a drive whose device has been disconnected and can't be reconnected. This is especially useful for USB disk drivers. FireWire disk drivers should continue to use the FireWire routine `FWWaitForDeviceRePlug` to force the user to reconnect the device.

COMPATIBILITY NOTE (System Error 119)
It is no longer possible to access the FCB table directly. Software that calls the PowerPC low-memory accessor functions for the FCB table will now halt the system with a `dsMustUseFCBAccessors` (119) system error. Developers needing to access FCB data should do so using the appropriate routines, as described in DTS Technote TN1184, "FCBs, Now and Forever."

COMPATIBILITY NOTE
Developers patching the existing File Manager programming interface in order to see all File Manager requests to a volume, will have to revise their software to patch the HFS Plus APIs as well.

Related Materials:

The File System Manager Section
Technote TN1184, "FCBs, Now and Forever"
The File Manager section in Technote TN1041, "Files Errata"
The Files Technote Collection
The Files Q&A Collection
The File Manager chapter of *Inside Macintosh: Files*

Back to top

File System Manager

The File System Manager provides a general means by which foreign file systems can be installed, identified, and interfaced to the operating system.

- New APIs have been added which allow FSM plug-ins to quickly locate FCBs when a large number of files have been opened.

- New APIs have been added, allowing FSM plug-ins to both validate FCB pointers and map FCB pointers to file reference numbers.

- A new API has been added that FSM plug-ins can use for checking permissions when using fork IDs.

- File Manager now implements a compatibility layer so that FSM plug-ins don't have to support the new HFS Plus API calls. As long as your FSM plug-in does not set the `kVCBFlagsHFSPlusAPIsBit` in the `vcbFlags` fields of the VCB, the compatibility layer will emulate the HFS Plus call on your volume by making one or more old-style calls to your plug-in. If you want to support HFS Plus APIs directly, you can prevent the emulation by setting this bit.

Related Materials:

The File Manager Section
The File System Manager 1.2 SDK

Back to top

Folder Manager

The Folder Manager provides facilities for locating "special" folders (such as, for example, the Extensions folder) without relying on the names of those folders. This aids developers in application localization.

- The Folder Manager has been revised to support multiple users.
- A new extended version of the `FindFolder` routine has been added that accepts new flags and data parameters, allowing for more precise control of the `FindFolder` operation.
- `FindFolder` now follows aliases to folders.
- Notification callbacks have been added along with other facilities for managing multiple user settings.
- `IdentifyFolder` performance has been improved.
- A new volume selector, `kOnAppropriateDisk`, has been added. This selector allows developers to establish the location of a particular folder when it is possible for the folder to be located on a number of different volumes.

Related Materials:

The FindFolder description in the Finder Interface chapter of *Inside Macintosh: Macintosh Toolbox Essentials*
The Using the System Folder and Its Related Directories section in the Finder Interface chapter of *Inside Macintosh: Macintosh Toolbox Essentials*
The File Manager chapter of *Inside Macintosh: Files*

Back to top

Font Manager

The Font Manager provides system-wide services for the retrieval of glyphs for display of textual information.

- The maximum number of active font suitcases in the Fonts folder has been increased from 128 to 512.
- Under Mac OS 8.6 the Font Manager was creating incorrectly formatted FOND resources and adding them to resource files. In Mac OS 9, the Font Manger has been revised so that it no longer does this. Furthermore, there is an update, Font Manager Update 1.0, available for use with Mac OS 8.6 that includes an extension to correct this problem, and a utility, Font First Aid, to fix fonts corrupted by this problem.
- A new Font Management API has been added to the Font Manager. This is a public API for accessing and modifying the font collection to support the basic tasks of font management. Functionality provided by the new Font Management API includes:
    - Enumerating fonts and font families.
    - Accessing information about font families, including name, character encoding, and member fonts.
    - Accessing information about fonts, including technology, format, and name.
    - Accessing font data.
    - Creating and managing a basic font menu.
    - Activating and deactivating fonts.
    - Notifying clients of changes in the font database.

Related Materials:
Mac OS 9 Font Management Documentation (coming soon)

Back to top

Mac OS USB

Mac OS USB is built into the Mac OS ROM file, and enables USB support for those Macintosh systems that have USB built in. The changes for v1.3 are described as follows.

- Added support for driver-requested timeouts.
- Implemented support in the HID driver to support vendor-specific drivers for Immersion-based force feedback joysticks.
- Implemented support for the USB Expert to get the international strings from a USB device.
- Provide USB suspend and resume-wake up support.

- Fixed bug where device names were not showing up when the "not enough power" dialog was displayed.

- `USBExpertStatus` calls are now re-entrant. This should solve the "missing messages" reported by some driver writers.

- Added new functions allowing USB Class Drivers to get and set the Expert Status level. This will allow developers to debug specific problems they are experiencing and not overflow the log. This functionality is currently only used by USB Prober.

- Added support for `kNotifySystemSleepRevoke` message.

- Improved the performance of isochronous and interrupt transfer routines under Virtual Memory.

- Fixed a problem that could permit dereferencing nil if we ran out of isoch descriptors.

- Implemented the `USBGetVersion` call to obtain the version of USB present. This call is provided as an alternative for using `Gestalt` so that you does not need to link with InterfaceLib in order to obtain the version of USB present. Listing 2 contains an example of how one could use this call to determine the installed USB version. For this example, you weak link the USBServicesLib as part of the project. The result of the call is a 4-byte `'vers'` response exactly like what would be returned by using the `Gestalt` call with the `'usbv'` selector.

```
#include <USB.h>  // ensure that you include the
                  // USB.h v1.3 header file

UInt32 MyUSBGetVersion(void) {
   UInt32 version;
   if ((Ptr) USBGetVersion != (Ptr) kUnresolvedCFragSymbolAddress)
      version = USBGetVersion();
   else
      version = 0;    // version of USB is less than 1.3
   return version;
}
```

Listing 2. Determining the installed version of USB.

- Fixed the `Isochronous frStatus` error codes. In USB 1.2 (and earlier) the wrong error codes were returned. For example a `DataOverRun` error would appear as a `Stall` error, and `DeviceNotResponding` was reported as a `BitStuff` error.

- Fixed a problem with the `OHCIUIM` in which multiple overcurrent dialogs appeared when the root hub detected an overcurrent situation.

- Made sure that the `kNotifySystemSleepDemand` messages are sent to the driver modules.

- Fixed the Mouse driver to function better on USB CardBus-equipped PowerBooks.

- Implements support for USB Audio Class devices, both speakers and microphones and for generic HID devices on USB speakers - volume, mute, bass, and treble adjustment.

- Implements support for USB Communication Class devices that comply with the USB Communication Class Abstract Control Model.

- Calls to the USB manager no longer return the USBPending result code. Now calls will always return either noErr or some other error code.

- The following changes have been implemented in the USB API:
    - `USBOpenPipe` call is no longer supported. Developers should now use the `USBConfigureInterface` call to set the active interface for the device. This call will also open all pipes in the interface. Use the `USBFindNextPipe` call to access the pipe references that you will need in passing data through the pipe.
    - The `USBOpenDevice` call has been replaced with `USBSetConfiguration`. Developers should use `USBSetConfiguration` instead of a device request to perform a `set_configuration` operation. The USL issues a status message to the `USBProber` log whenever a `set_configuration` `USBDeviceRequest` is made.

Related Materials:

The Latest USB DDK

Back to top

Memory Manager

The Memory Manager is the part of the operating system responsible for managing memory allocation requests.

- `NewHandle` was crashing when the size parameter was a value between `0xFFFFFFE1` and `0xFFFFFFFF` rather than returning `NULL` with `MemError` returning `memFullErr`. This problem has been corrected - now, for this size range, `NewHandle` returns `NULL` and `MemError` returns `memFullErr`.
- Memory in Memory Manager free blocks is now released with the Virtual Memory function `ReleaseMemoryData`. This can greatly reduce the number of disk accesses when Virtual Memory needs to handle page faults. However, this also means that programs that attempt to use pointer or handle data after disposing it (a programming error) will likely get garbage data.

Related Materials:

The Virtual Memory Manager Section.
Technote IM_ERRATA 02, "Inside Macintosh - Memory Errata"
The Memory Technote Collection
The Memory Q&A Collection
The Memory Manager chapter of *Inside Macintosh: Memory*

Back to top

Mixed Mode Manager

The Mixed Mode Manager provides services for switching between 68k emulation and PowerPC instruction execution environments.

- Accelerated code resources no longer crash if they are moved in memory after they are initialized.

Related Materials:

Technote 1077, "Calling CFM Code From Classic 68K Code, or There and Back Again - A Mixed Mode Magic Adventure"
The Mixed Mode Manager chapter of *Inside Macintosh: PowerPC System Software*

Back to top

Multitasking/Multiprocessing 2.1

The Multitasking/Multiprocessing Library allows applications to perform preemptive multiprocessing operations on PowerPC computers equipped with one or more processors. Several new features have been added to this release of the Multitasking/Multiprocessing Library including:

- Parameter block-based File and Device Manager requests which have an asynchronous variant, and high-level calls which are based on such calls, can now be called synchronously from MP tasks. (Other file manager and device manager routines that are always executed synchronously, such as `PBMountVol`, cannot be called from MP tasks.)
- MP tasks can now call `WakeUpProcess`.
- MP tasks can now call `DTInstall`.
- The MultiProcessing library now loads on all machines where Mac OS 9 can be installed.
- Greater than 1GB RAM Support.

Related Materials:

Multiprocessing SDK
Multi processing Documentation

Back to top

Navigation Services 2.0

Navigation Services provides new **Open** and **Save** dialogs, allowing users to locate and select files and other resources. In Mac OS 9, Navigation Services 2.0 is built in to the System file. Features new to Navigation Services 2.0 include:

- A new function has been added which simplifies the process of attaching a preview resource to a document (`NavCreatePreview`).
- Support has been added for Carbon application packages. Packages appear as files to the user if the client requests them (`kNavSupportPackages`). An option is also available to allow packages to appear as folders, allowing users navigate into them (`kNavAllowOpenPackages`).
- A new option is available to turn off the bevel frame around the custom control area (`kNavDontUseCustomFrame`), and another option prevents opened files from being added to the **Recent** menu (`kNavDontAddRecents`).
- A new event selector notifies the client when the user has opened the current selection (`kNavCBOpenSelection`).
- File filtering has been relaxed, allowing the component signature in the `NavTypeList` to have a wild-card value (`kNavGenericSignature`).
- A number of new custom control selectors have been added, allowing the client to automate and restrict certain navigation actions (`kNavCtlBrowserSelectAll`, `kNavCtlGotoParent`, `kNavCtlSetActionState`, `kNavCtlBrowserRedraw`, `kNavCtlTerminate`).

Related Materials:

Q&A TB38, "Navigation Services Versions"
Q&A TB40, "Navigation Services and memFullErr"
The Navigation Services SDK.

Back to top

Notification Manager

The Notification Manager provides a mechanism for programs which are not the front-most application to notify the user of specific events.

- The Notification Manager now displays alert notifications using a floating window. This allows the current application's event loop to continue running while a Notification Manager alert is on the screen. Rather than using the cancel button as in the older style "modal" alerts, these new Notification Manager alerts are dismissed by clicking in the window's close box. The new non-blocking Notification Manager fixes both of the problems described in TN1026, "The Notification Manager: Problems & Fixes".

- The Notification Manager now plays custom alert sounds (specified with the `nmSound` field of the request record) asynchronously.

- The Notification Manager no longer throws away notification requests that were posted during early boot time.

COMPATIBILITY NOTE
Some applications were built to depend on the event-loop-blocking behavior of Notification Manager alerts, and assumed that after posting an alert notification, the posting application would not run again until the user had dismissed the alert. This assumption has never been correct; there has always been a small window during which the posting application could be activated before the alert notification was displayed. With the new non-blocking behavior of the Notification Manager, it becomes very easy for the user to activate the posting application without dismissing the alert. Applications should be aware of this possibility and not reuse a request record before the user has dismissed the notification. An application may also wish to explicitly cancel an alert notification with `NMRemove` when the application becomes active.

As in previous versions of the Notification Manager, the notification response procedure will not be called until the user dismisses the alert. If your application reuses request records, it should always provide a response procedure that marks a request record as no longer in use. This will prevent your application from reusing the request record before the alert is dismissed.

Related Materials:

Technical Q&A OPS22, "Notification Manager Reinitialized During Boot"
Technote PS505, "Notification Manager Q&As"
Technote TN1026, "The Notification Manager: Problems & Fixes"
The Notification Manager chapter in *Inside Macintosh:Processes* .

Back to top

Process Manager

The Process Manager shares the processor among multiple applications.

- A problem in Mac OS 8.6 that could cause slower Apple event performance has been corrected.

COMPATIBILITY NOTE
There is no process in the system that is guaranteed to remain operational from system startup to system shutdown. As a result, control panels, system extensions, and applications should not assume that state variables belonging to other processes, such as, for instance, a processes' A5 value, will remain constant across calls to `WaitNextEvent`; and, nor should any process assume that it is safe to allocate persistent storage in another processes' heap zone and expect it to remain intact across calls to `WaitNextEvent`. Processes requiring persistent storage outside of their own heap zone can allocate memory in the system heap zone.

Related Materials:
Technote TN1070, "Background-Only Applications"
Technote PS510, "Process Manager Q&As"
The Processes Q&A Collection
The Process Manager chapter in *Inside Macintosh:Processes*

Back to top

Power Manager 2.0

The Power Manager provides facilities for controlling electrical power usage on the computer.

- The new Power Manager includes several new APIs that allow for finer control over energy consumption.

Related Materials:
Technote TN1190, "Power Manager 2.0"

Back to top

PPC Toolbox

The Program-to-Program Communications Toolbox provides process-to-process communications facilities both for processes running on the same machine, and for processes running on different computers connected by either an AppleTalk network or a TCP/IP network.

- The PPC Browser window now provides facilities for selecting processes on a remote machine over TCP/IP. Users can target a machine for display in the PPC Browser window by providing an address in either the domain name format or the dotted-decimal address format.

COMPATIBILITY NOTE
Guest access is not available for PPC over TCP/IP connections.

- A new routine has been added for canceling outstanding list ports calls (`IPCListPorts`). The new routine `IPCKillListPorts` takes the parameter block from a pending `IPCListPorts` call and cancels the call immediately.

- URLs specifying the target machine for a PPC over IP connection may specify the IP address by using either a domain name or an IP dotted-decimal address. For example, the following are both valid URLs for PPC over IP connections:

  eppc://mymac.mycompany.com

  eppc://127.76.94.105

  COMPATIBILITY NOTE
  The implementation of PPC over TCP/IP listens at port number 3031 for incoming PPC over TCP/IP connections. Although it is not possible to change the listening port number, it is possible to specify a port number in outgoing PPC over TCP/IP connection requests.

  This ability is useful in situations where multiple hosts share a single IP address. For example, on the outside of a network address translator (NAT) a number of different machines may have the same IP address, while on the inside of the NAT they may each have a unique IP address. In these cases, it is necessary for external clients to refer to different machines using different port numbers instead of unique IP addresses. The NAT itself must do the translation from port numbers to internal IP addresses using port number 3031.

  Using this same example, external clients can select a specific target machine by specifying the machine's port number in the outgoing PPC over TCP/IP connection by providing the port number in the destination address. For example, the address `eppc://myhost.mycompany.com:1234` specifies port number 1234.

- `LocationNameRec` structure has been extended to support PPC Toolbox over TCP/IP. The new structure's definition can be found in `"PPCToolbox.h"` in Universal Interfaces 3.3 and higher. A new `PPCLocationKind` constant has been added (`ppcXTIAddrLocation`) to indicate the `location` field contains an XTI address. Also, a new union member in the `location` field, `xtiType`, has been added that holds the XTI address itself. The XTI address is stored in the same format used by Open Transport routines (such as `OTConnect`); for PPC Toolbox over TCP/IP, you can use either an IP address (`AF_INET`) or a DNS address (`AF_DNS`).

  COMPATIBILITY NOTE
  The maximum length of the XTI address stored `LocationNameRec` is 98 bytes. This restriction is required so as to not change the overall length of the `LocationNameRec` structure.

  This is not a problem when connecting to an IP address, because the IP address structure (`InetAddress`) is always a fixed size (16 bytes). It is potentially a problem for DNS addresses, which can be up to 255 characters long. If your software passes DNS addresses to PPC Toolbox and it receives a DNS address string longer than 96 bytes (98 minus the two-byte `fAddressType` field), it must resolve the DNS address to an IP address (using `OTInetStringToAddress`) before passing it to PPC Toolbox.

The code snippet shown in Listing 3 shows how you might target an Apple event over TCP/IP or AppleTalk by setting up the `location` field of the `TargetID` structure appropriately. Note that

OTInitDNSAddress is an Open Transport routine that initializes an XTI address (AF_DNS) from a C-style string and returns the XTI address length.

```
TargetID   target;
AEDesc     targetDesc;
AppleEvent event;
AppleEvent reply;

// Set up sessionID and name fields of target as before.

if (useAppleTalk) {

    // target "The Count"

    target.location.locationKindSelector = ppcNBPLocation;
    PLstrcpy(target.location.u.nbpEntity.objStr, "\pThe Count");
    PLstrcpy(target.location.u.nbpEntity.typeStr, "\pPPCToolBox");
    PLstrcpy(target.location.u.nbpEntity.zoneStr, "\p*");

} else {

    // target "eppc://the-count.apple.com"

    target.location.locationKindSelector = ppcXTIAddrLocation;
    target.location.u.xtiType.Reserved[0] = 0;
    target.location.u.xtiType.Reserved[1] = 0;
    target.location.u.xtiType.Reserved[2] = 0;
    target.location.u.xtiType.xtiAddrLen =
    OTInitDNSAddress((DNSAddress *)&target.location.u.xtiType.xtiAddr,
             "the-count.apple.com");
}
err = AECreateDesc(typeTargetID, &target,
    sizeof(target), &targetDesc);
if (err == noErr) {
    err = AECreateAppleEvent('syso', 'beep',
        &targetDesc, kAutoGenerateReturnID,
        kAnyTransactionID, &event);
}
if (err == noErr) {
    err = AESend(&event, &reply,
    kAENoReply, kAENormalPriority, kNoTimeOut, nil, nil);
}
```

Listing 3. Targeting and Apple event over TCP/IP.

- The PPC Toolbox can now utilize Keychain Access to automate secure connections. This allows background-only applications to initiate secure PPC Toolbox sessions and to send Apple events to processes on machines requiring authentication. In the past, the user interaction required for authentication made it impossible for background-only applications to establish secure connections requiring authentication. Now, in Mac OS 9, the PPC Toolbox routine StartSecureSession (also used by the Apple Event Manager) works together with Keychain Access to allow any process to open one or more secure sessions without any user interaction.

  If there is an unlocked keychain that contains an entry for the target machine, then a secure session can be established without user interaction; otherwise, if a keychain entry has not been created for the target machine, or if one exists but the keychain containing it is locked, then an error will be returned indicating a user interaction error has occurred.

Related Materials:

The discussion of Open Transport TCP/IP stack transition events in the Open Transport section of this note

Back to top

QuickDraw

QuickDraw is the part of the Mac OS used for drawing and displaying graphical information on the screen and other raster devices.

- Large cursor support has been added. It is now possible for developers to build cursors larger than the 16-by-16 pixel cursor of the past. In addition, it is possible to create animated color cursors that are updated at interrupt level. The new cursor facilities allow for hardware acceleration, which allows cursors to be rendered inside of the screen buffer and for switching between hardware and software rendering. To take advantage of this new facility, developers must create a cursor plug-in component. For more information consult the cursor plug-in component SDK (coming soon).
- New "add noise" and Color XOr transfer modes have been added.
- A flag has been added to tell `NewGWorld` to allocate memory for the raster data on the video card or AGP (Accelerated Graphics Port).
- Facilities for adding user-defined transfer modes are now available. Users can specify new transfer modes by providing a callback procedure that implements the operations required.
- Corrects a problem that could occur when scaling raster images for landscape printing where a column of pixels would not appear in the printed image.
- Performance of the `LockPixels` and `SetCPixel` routines have been improved.
- Accessor routines have been added for all QuickDraw structures.

COMPATIBILITY NOTE
Developers should begin using the new public accessors rather than accessing QuickDraw structures directly to ensure future compatibility of their source code.

Related Materials:

The QuickDraw Technote Collection
The QuickDraw Q&A Collection
Inside Macintosh: Imaging with QuickDraw

Back to top

Sound Manager

The Sound Manager provides facilities for playback and recording of digitized sounds.

- A problem where clicking sounds could occur during sound playback on some PowerBook models when VM is turned on has been removed.
- The Sound Manager has been re-implemented. The new implementation features improved performance and robustness.
- In the new implementation it is only possible to have one sound input source active at a time.

Related Materials:

Technote TN1048, "Some Sound Advice: Getting the Most Out of the Sound Manager"
Technote TN1124, "New Sound Input Driver Features"
Technote TN1108, "Unknown Sound Features"
The Sound Manager chapter of *Inside Macintosh: Sound*
Olson, K. Sound Secrets. Develop 24

Back to top

Standard File

Standard File provides a consistent user interface for accessing files by way of the **Open** and **Save As...** dialog boxes.

- There have been no changes in Standard File; however, developers using Standard File in their applications should be aware that doing so makes their applications incompatible with Mac OS 9 packages.

COMPATIBILITY NOTE
Mac OS 9 packages are not displayed correctly by the Standard File dialogs. The dialogs displayed by Standard File allow users to navigate down into Mac OS 9 packages. Navigation Services handles display of packages correctly. Developers interested in providing a consistent user interface with other Mac OS 9 applications should replace their calls to Standard File with calls to Navigation Services.

Related Materials:

The Navigation Services Section
The Standard File Package chapter of *Inside Macintosh: Files* .
The Navigation Services SDK

Back to top

StdCLib 3.5

StdCLib is a shared library located in the ROM on all PowerPC-based Mac OS compatible computers providing functionality for programs compiled using the C language. StdCLib has been present on all PowerPC models, and prior to System 7.6 updates were provided via the StdCLibInit system extension.

- A problem in signal handling where native tools could crash when interrupted by Command-"." (period) has been corrected.
- Support for AltiVec has been added, including new memory allocation routines that allocate memory on 16-byte boundaries for vector data, `setjmp`/`longjmp` variants that save and restore the processor's AltiVec state, and new `printf` and `scanf` format size specifiers for reading and writing AltiVec data. The new entry points `vec_malloc`, `vec_calloc`, `vec_realloc`, and `vec_free` have been added to support 16-byte aligned allocations for vector data.

  COMPATIBILITY NOTE
  A bug was found in the printing of vector data using the `printf` family in the 3.5 version of StdCLib shipped in Mac OS 9.0. In some cases, vector data could be misaligned on the stack and the output would be garbled.

- The standard C library's string functions have been optimized for PowerPC.
- Support for native 64-bit long integers (`long long`) has been added. The routines `printf`, `scanf`, `atoll`, `strtoll`, and others support this data type.
- The C library's `malloc` implementation has been replaced. New features include improved performance, the ability to allocate more than 8MB in a single call, and a more intelligent scheme for returning sub-allocation blocks to the Macintosh Memory Manager - when memory is running low, the new `malloc` will return empty sub-allocation blocks to the Macintosh Memory Manager.
- New `FSSpec` I/O routines have been added. These routines allow the use of Macintosh file specification records in conjunction with the standard C library I/O routines. The new routines are defined in `FSpio.h` header file. Routines provided include `FSp_creat`, `FSp_access`, `FSp_fopen`, `FSp_freopen`, `FSp_fsetfileinfo`, `FSp_open`, `FSp_remove`, `FSp_rename` and `FSp_unlink`. They function identically to their standard C library counterparts - the only difference being that they accept a `FSSpec` pointer rather than a string to identify the file.
- `<time.h>` functions have been enhanced/fixed. The `tm_isdst` field is set correctly in `mktime` and `localtime`. The `%p` format specifier is now supported in `strftime`. The calculation of leap year has been corrected in `localtime`. The `gmtime` function is now fully supported.
- `fgets` now recognizes line endings for Mac, DOS, and UNIX files.
- All I/O functions requiring C-style path names to identify files now work correctly with names longer than 255 characters.
- `memchr` and `strchr` now handle 8-bit characters correctly.
- Two bugs in `strncat` were fixed. The first occurred when `n % 4 == 3` and `n < strlen(s2)` - the last character was chopped off. The second occurred when `n % 4 == 0` and `n <= strlen(s2) <= n + 3` - the final three characters were chopped off.
- Fixes to `PLstr` suite of routines. Specifically, the functions `PLstrchr`, `PLstrrchr` and `PLstrspn` had bugs where these functions would scan one character beyond the length of the target string (`PLstrchr`, `PLstrrchr`) or one character beyond the character set specifier (`PLstrspn`). These bugs were fixed. Also, a bug was fixed in `PLstrncpy` in which the

destination string was mishandled when `len(SRC) < n <= len(DST)`.

- Fixed an off-by-one error in `[S|U]Int64ToStr` which could cause an extra (garbage) character to be output.
- Double-rounding in `scanf` has been corrected.
- Performance of `getenv` has been improved.

Related Materials:
The StdCLib section in Technote TN1102, "Mac OS 8.0"

Back to top

Multilingual Text Editor 1.1

MultiLingual Text Editing (MLTE) supplies a C language interface for creating and editing Unicode text documents. Although MLTE may appear to be functionally similar to TextEdit in many ways, MLTE provides many features that are not provided in TextEdit. These features include:

- A full unicode text rendering engine (ATSUI) for drawing text.
- The ability to add embedded objects like sound, graphics, and QuickTime movies.
- Document-wide tabs.
- Full justification of text.
- More than 32K of text.
- Built-in support for Drag & Drop.
- Built-in scroll bar handling.
- Built-in printing support.
- Built-in support for inline input.
- Support for advanced ATSUI font features.
- Support for multiple levels of undo.

Related Materials:
MLTE Documentation

Back to top

Thread Manager

The Thread Manager is part of the operating system providing multi-programming services for use within application programs.

- The routines `GetDefaultThreadStackSize`, `GetSpecificFreeThreadCount`, and `GetFreeThreadCount` were returning `noErr` when called with the `kPreemptiveThread` constant on a PowerPC. This was incorrect as it is not possible to create preemptive threads using the Thread Manager on PowerPC computers. This problem has been present in the Thread Manager since Mac OS 8.1, and it has been corrected in Mac OS 9.0. Now, these routines will return the `paramErr` error code as expected.

- The Thread Manager routine `SetThreadReadyGivenTaskRef` now works correctly when given a reference to a stopped thread. This fix applies to the PowerPC version of the Thread Manager - it has not been implemented for the 68k version. In past versions of the Thread Manager, this routine did not work when it was called with a reference to a stopped thread.

Related Materials:

The Thread Manager 2.1 SDK
The Concurrent Programming With the Thread Manager. Develop 17

Back to top

Unicode Text Utilities 1.1

The Unicode Text Utilities provide a set of APIs for working with Unicode text. These APIs provide services for sorting and collating Unicode text, finding word breaks, establishing character properties, and so on.

- New character properties APIs have been added.

- Unicode text boundaries/break discovery APIs have been added.

- Bug fixes associated with the `UCKeyTranslate` routine tbd.

Related Materials:

Online Unicode Utilities Documentation

Back to top

UTC Utilities

The UTC (Coordinated Universal Time) Utilities provide facilities for accessing the clock chip and APIs for converting dates between various formats.

- New 64-bit routines are now available for accessing the system's clock in both UTC and Local Time formats. Converting a value returned by `GetDateTime` to a value usable by the new routines is just a matter of copying the value returned by `GetDateTime` into the low seconds value in the new time structure (along with initializing the other fields to their appropriate values).
- Facilities have been provided for conversion between UTC (Coordinated Universal Time) and local time. With this release, these routines operate on dates between the years 1904 and 2108.
- The new routines include provisions for calculating fractions of seconds. This fractional value is stored as a 16-bit value in the new time structure and is interpreted as seconds = value / `0xFFFF`.

Related Materials:

Technote TN1049, "Approaching the Millennium: The Mac and the Year 2000"
Q&A OPS01, "System Clock"
Technote TN1064, "Inside Macintosh: Operating System Utilities: Addendum to Chapter 4 - Determining If Daylight Savings Time is in Effect"
The Date, Time, and Measurement Utilities chapter of *Inside Macintosh: Operating System Utilities*
Minow, M. Timing on the Macintosh. Develop 29

Back to top

Virtual Memory Manager

The Virtual Memory Manager provides virtual memory services for the Mac OS. The following changes have been made in the Virtual Memory Manager:

- Fixed problem where cleaning a dirty page marked it "recently used" when non-DMA disk drivers access the page's data to write it to disk.
- New API, `GetVolumeVirtualMemoryInfo`, for getting information on the volume's Virtual Memory uses for backing storage. This call will not be supported under Carbon.
- A bug in the Virtual Memory Manager that was causing data corruption in a very rare case has been fixed. This bug was only seen on very fast striped hard disk arrays, where the stripe boundary for a read operation fell in the middle of a page.

Related Materials:

The Memory Manager Section.
Technote TN1094, "Virtual Memory Application Compatibility"
Technote TN1104, "Interrupt Safe Routines"
Technote IM_ERRATA 02, "Inside Macintosh - Memory Errata"
Q&A DV32, "PrepareMemoryForIO and Execution Levels"
The Memory Technote Collection
The Memory Q&A Collection
The Virtual Memory Manager chapter of *Inside Macintosh: Memory*
Minow, M. The New Device Drivers: Memory Matters. Develop 24

Back to top

Control Panels

Control Panels provide user interface facilities for custom configuration of the system.

Old-style control panel files (files of type `'cdev'`) are now launched in a separate Process Manager partition. By default, the size of this partition is about 200K; however, the size of this partition can be customized by including a `'SIZE'` resource with ID = `-16474`. When such a `'SIZE'` resource is present, the minimum and suggested sizes will be used.

The Process Manager will choose to launch a control panel file as an application if it contains either a `'CODE'` ID = 0 resource (68K app), or it has a `'cfrg'` resource with a fragment descriptor marked as an "application"type. Control Panels that meet these criteria will not have their `'cdev'` resources called by the system.

For backwards compatibility with some control panels that must run in the Finder's partition, it is possible to open a control panel in the Finder layer by holding Command and Control keys while double-clicking the control panel icon. This technique only works for control panel files containing a `'cdev'` resource. As this feature is only present for compatibility issues in Mac OS 9, it will likely be removed in the next system revision.

Appearance

The Appearance control panel provides a user interface for configuring the appearance of graphical user interface elements.

- The Appearance Control Panel no longer leaks a large amount of memory in a reserved area of the system heap each time it is opened.

Related Materials:

Q&A TB39, "Appearance Versions"
The Appearance Manager Reference chapter of *Inside Macintosh: Mac OS 8 Toolbox Reference*
Inside Macintosh: Macintosh OS 8 Human Interface Guidelines
The Appearance SDK

Back to top

ColorSync 3.0

ColorSync provides system-level color management that enables publishing software to achieve repeatable, reliable, and consistent color on-screen, in print, and for electronic delivery.

- The new control panel provides a greater range of configuration options allowing association of ColorSync profiles with different functional uses.
- Allows named settings allowing users to save and restore control panel settings.

Related Materials:

The ColorSync Manager Documentation Suite
Hayward, D. Printing Hints: Syncing Up With ColorSync 2.0. Develop 23
ColorSync web pages
The ColorSync 3.0 SDK

Back to top

Date & Time

The Date and Time control panel provides a user interface for setting the system clock and configuring the display format for both dates and times.

- By default, dates are now displayed and entered using four-digit rather than two-digit centuries.

Related Materials:

Technote TN1049, "Approaching the Millennium: The Mac and the Year 2000"
Q&A OPS01, "System Clock"
Technote TN1064, "Inside Macintosh: Operating System Utilities: Addendum to Chapter 4 - Determining If Daylight Savings Time is in Effect"
The Date, Time, and Measurement Utilities chapter of *Inside Macintosh: Operating System Utilities*
Minow, M. Timing on the Macintosh. Develop 29

Back to top

File Exchange 3.0.2

File Exchange provides automatic file name extension to Mac OS file type mapping (for example, mapping the file name extension `".TXT"` to the Mac OS file type `'TEXT'`) along with other services associated with using volumes created for other file systems with Mac OS. The File Exchange control panel allows users to modify settings used by this facility, add new file mappings, mount DOS SCSI disks and drive containers, and set File Translation preferences. New features for this version of File Exchange include:

- File exchange now provides better support for 120MB SuperDisk disks in PC format.
- Corrects a problem where an incorrect volume size could be assigned to a volume formatted by File Exchange.
- File Exchange now informs the File Manager when a volume mounted by File Exchange has been unplugged (so long as the underlying disk driver supports the new `driverHardwareGoneErr` error code described in the File Manager section).
- Some problems with reformatting PC format volumes into Mac OS format while PC Exchange is turned on have been corrected.
- File Exchange no longer allows the name of a DOS disk to be changed to " " (blank).
- Clicking on the **Mount Now** button in the File Exchange control panel's window now provides additional information and feedback in cases where no volume could be mounted.
- Has been updated to support features of the new HDI driver, version 1.4.1.

Related Materials:
Technote PT515, "Apple File Exchange Q&As"

Back to top

File Sharing

The File Sharing control panel allows for end user configuration of file sharing services on the computer.

- Adds support for Personal File Sharing over IP. Personal file sharing over TCP/IP is implemented by a new background process called "ShareWay IP Personal Bgnd." It works in conjunction with the File Sharing Extension, and was licensed from Open Door Networks. AppleTalk must be active to use File Sharing over IP.
- Adds support for turning on or off Program Linking over IP.
- Passwords are now entered using masked text for better security.
- The Users and Groups control panel has been integrated into the File Sharing control panel. AppleScripts that were designed to work with the Users & Groups control panel will have to be modified to target the File Sharing control panel.

Related Materials:

The File System Manager, File Manager, and AppleShare Client Sections
The Networking Technote Collection
The File Sharing section in Technote TN1102, "Mac OS 8"
Technote TN1111, "Programmatic Mounting of AppleShare Volumes"
Technote FL20, "File Sharing and Shared Folders"
Technote FL28, "PBShare, PBUnshare, and PBGetUGEntry"
Technote FL37, "You Want Permission to do What?!!"
AppleShare IP web pages

Back to top

FontSync

FontSync provides a common method for identifying fonts based upon the content of the font rather than just the name as a document is moved from machine to machine. In Mac OS 9, FontSync is installed in the "Apple Extras" folder. To use it, the user needs to drag the control panel into the System Folder.

- This is a new feature introduced with Mac OS 9.

Related Materials:

FontSync Documentation

Back to top

Memory 8.1.1

The Memory control panel provides a user interface for the configuration of virtual memory, RAM disk storage, and the size of the disk cache.

- Introduces AppleScript support for configuring memory options.
- PowerPC implementation added.

Back to top

Monitors

The Monitors control panel provides facilities for configuring monitors attached to the computer.

- In Mac OS 9 the Monitors & Sound control panel has been divided into two different control panels. The Monitors control panel provides the monitor configuration user interface elements formerly found in the Monitors & Sound control panel.

Back to top

Mouse

The Mouse control panel provides facilities for configuring the mouse's tracking speed and double-click time interval.

- The new Mouse control panel features an updated graphic interface.
- AppleScript support has been added.

Back to top

Text

The Text control panel provides user interface facilities for configuring international text handling.

- On systems with bidirectional scripts installed, the Text control panel now only affects the line direction for applications localized into those scripts. Applications localized in non-bidirectional scripts are not affected by the direction setting. The default setting is now right-to-left.

Back to top

Sound

The Sound control panel provides facilities for configuring alert sounds and configuring sound input and output preferences.

- In Mac OS 9 the Monitors & Sound control panel has been divided into two different control panels. The Sound control panel provides the sound configuration user interface elements formerly found in the Monitors & Sound control panel.

Back to top

Software Update

Software Update provides a way to find and install the latest system software for your computer by way of the Internet.

- Software Update is new for Mac OS 9. There are no public APIs for this facility.

Back to top

System Extensions

System extensions are located in the Extensions folder inside the System Folder.
For a system extension to be active, it must be located in the Extensions folder
when the system starts up. Extensions are not loaded if the Shift key is held down while the system is starting up.

AppleScript 1.4

AppleScript is a component of the operating system that provides a scripting interface for users to automate actions that would normally require use of the menus and keyboard. Even more importantly, it allows users to access functionality of applications which would be difficult or impossible to access by hand.

- When AppleScript 1.4 tries to locate an application, it will search the Scripting Additions folder for applications and application aliases. For example, if you put an alias to FileMaker Pro in the Scripting Additions folder, AppleScript 1.4 will be able to use that alias to locate the application even if the application is not running.

  When searching for applications and application aliases in the Scripting Additions folder, AppleScript 1.4 will use the name of the item in the Scripting Additions folder (even if the item is an alias and it refers to a file with a different name). In many circumstances, this can be a convenience as it allows script developers to create aliases whose names are abbreviations for applications and then use those abbreviations in their scripts. For example, one might create an alias named "FM" referring to the application "FileMaker Pro" and place it in the Scripting Additions folder. After creating that alias, "FM" can be used in a script to refer to the application "FileMaker Pro".
- "Tell" blocks can now accept URLs in the "machine" parameter, allowing delivery of AppleEvents to remote applications via TCP/IP protocol.
- A facility for targeting particular applications in scripts (specifically, the "using terms from" construct) has been added replacing the double-tell block application targeting method described in the Mac OS 8.6 technote.
- Script Editor is now a Carbon application.
- The "file busy" flag to has been added to the `FileInfo` result to allow a script to wait until a file operation is complete (i.e., until the file is no longer busy).
- Try blocks with optional "on error." It is no longer necessary to include the "on error" section before the "end try." This requirement has been removed.

COMPATIBILITY NOTE
The "mount volume" scripting addition provided with Mac OS 9 uses a `PBVolumeMount` that requests 2-Way Randnum Exchange. Some scripts may have been built that assume the "mount volume" scripting addition will fall back to using clear text passwords for logging into servers that only support clear text passwords (such as, for instance, Mac OS X servers or Windows NT servers). As described in the AppleShare Client section, this "fallback to a lower security mode" no longer occurs in `PBVolumeMount` calls. As a result, the "mount volume" scripting addition cannot be used to log in to servers that only support clear text passwords.

COMPATIBILITY NOTE
AppleScripts that use the new features of AppleScript 1.4 (specifically the "using terms from" clause and IP addressing) will not run on earlier versions of AppleScript or Mac OS.

COMPATIBILITY NOTE
AppleScript now loads at component registration time, not INIT time, so beware of load-order dependencies.

Related Materials:

Technote TN1164, "Native Scripting Additions"
The discussion of AppleScript in chapters 7 (Introduction to Scripting), 8 (Apple Event Terminology Resources), 9 (Recording Apple Events), and 10 (Scripting Components) of *Inside Macintosh: interapplication Communication*
Inside Macintosh: AppleScript Finder Guide
Inside Macintosh: AppleScript Language Guide
Inside Macintosh: AppleScript Scripting Additions Guide
The AppleScript SDK

Back to top

AppleShare Client 3.8.5

The AppleShare Client provides file sharing services for computers networked with Apple Filing Protocol (AFP) file servers.

- Adds support for Keychain.
- The method used for mounting server volumes at startup has changed. Rather than using the older method where mounting information was stored in the AppleShare Prep file, a new folder entitled "servers" is placed in the Startup Items folder containing aliases of server volumes to mount at startup. Aliases of type `'srvr'` will auto-route to this folder.
- Support for Wake-on-LAN capable machines dozing functionality has been added.
- A DHX User Authentication Module has been added. This allows for encrypted logins to Mac OS Server devices.

COMPATIBILITY NOTE
The `PBVolumeMount` call (for AppleShare Servers only) no longer negotiates a less secure connection than was requested in the `UAMType` field. This means that a `PBVolumeMount` call that requests 2-Way Randnum Exchange (`UAMType` = 6) will not fall back to using ClearText (`UAMType` = 2). This can be a problem when accessing servers that typically support ClearText only, such as servers running on UNIX, NT, or NetWare. The client will negotiate to more secure authentication methods if they are available.

COMPATIBILITY NOTE
AppleShare Client 3.8.5 or later is required for Mac OS 9. Earlier versions of the AppleShare Client will not work with Mac OS 9.

Related Materials:

Technote TN1111, "Programmatic Mounting of AppleShare Volumes"
Technote FL20, "File Sharing and Shared Folders"
Technote FL28, "PBShare, PBUnshare, and PBGetUGEntry"
Technote FL37, "You Want Permission to do What?!!"
AppleShare IP web pages

Back to top

ColorSync 3.0

ColorSync provides system-level color management that enables publishing software to achieve repeatable, , and consistent color on-screen, in print, and for electronic delivery.

- APIs have been added for identifying profiles and assigning them according to:
  - Functional area: input, output, proof, display.
  - Input devices such as scanners and digital cameras.
  - Printers, including press.
  - A default proofing device.

COMPATIBILITY NOTE
Starting with ColorSync 3.0, the name of the ColorSync profiles folder on Japanese system software will change from "ColorSync Tokusei" to "ColorSync Profiles" (in Kana). Software that relies on the name of the ColorSync Profiles folder needs to be updated. Developers should use the `'prof'` Folder Manager selector introduced with Mac OS 8.5.

Related Materials:
Technote TN1185, "New ColorSync 3.0 APIs"
The ColorSync Manager Documentation Suite
ColorSync web pages
The ColorSync 3.0 SDK

Back to top

DrawSprocket 1.7

DrawSprocket provides drawing services coordinated with the display manager that are useful in game development. This version of DrawSprocket provides a number of corrections and fixes over previous versions.

- Improved blitting performance.
- Routines have been added for better context selection control.
- Routines have been added allowing direct switching between two different DrawSprocket contexts.

Related Materials:

Apple Game Sprockets SDK

Back to top

Find By Content

Provides system-wide search facilities for finding related files based on their content.

- Find By Content's overall performance has been improved for both indexing and searching.
- A new API has been introduced allowing for the immediate indexing of new or altered files on demand. The new routine is declared as shown in Listing 3.

```
OSErr FBCIndexItems(
     FSSpecArrayPtr theItems,
     UInt32 itemCount);
```

- `theItems` is a pointer to an array of file specification records referring to the files to be indexed.
- `itemCount` is the number of items in the array of file specification records.

Listing 3. `FBCIndexItems` declaration.

`FBCIndexItems` indexes (or re-indexes) the files referred to in the array of file specification records passed as a pointer in the first parameter. If the volume containing a file already has an index, the document is added or re-indexed; and, if the volume does not contain an index, a new index is created.

Normally you will call `FBCIndexItems` after saving a file (or updating a file) on a volume containing an index. This will allow users to keep their indexes up to date without any additional effort. For more information about how to determine if a volume contains an index, refer to the Sherlock technote.

- Expanded Language Support for indexing volumes has been added. Earlier versions of Find by Content supported a set of Western European languages, Japanese, and Korean. In Mac OS 9, support has been added for all other Roman-alphabet languages and for languages using the Arabic, Hebrew, Greek, and Cyrillic alphabets. The specific list of supported languages (and groups of languages) is Afrikaans, Arabic, Farsi, Urdu, Catalan, Croatian, Cryllic languages, Czech, Danish, Dutch, English, Estonian, French, German, Greek, Hebrew, Hungarian, Icelandic, Italian, Japanese, Korean, Latvian, Lithuanian, Norwegian, Other (Roman alphabet), Polish, Portuguese, Romanian, Slovak, Slovene, Spanish, Swedish, Turkish, and Yiddish.

COMPATIBILITY NOTE
Once an index has been created with a particular set of languages it is not possible to change the set of languages used in the index. To change the set of languages used in the index it will be necessary to delete the index file and create a new one with a new set of language settings.

Related Materials:
Technote TN1141, "Extending and Controlling Sherlock"
Technote TN1180, "Sherlock's Find By Content Library"
Technote TN1181, "Sherlock's Find by Content Text Extractor Plug-ins"

Back to top

HTMLRenderingLib 1.0

HTMLRenderingLib is Apple's lightweight HTML rendering engine capable of displaying HTML 3.2 files.

- Provides a convenient and easy to use API for displaying HTML files.
- Clicking on HTTP links automatically launches the default browser to display a web page.

Related Materials:
HTML Rendering Library Documentation

Back to top

InputSprocket 1.7

InputSprocket provides a consistent set of interfaces to various input devices such as joysticks that are useful in game development. This version of InputSprocket provides a number of corrections and fixes over version 1.3 as follows:

- Adds universal USB support for all USB HID-compliant devices. This means almost any USB gaming device will work off the shelf, without a custom driver.
- It is now possible for InputSprocket to use a mouse with more than two axes or more than three buttons.
- Adds support for calibration of devices and provides an interface for doing so in the InputSprocket configuration dialog.
- All input sprockets have been consolidated into one file.

Related Materials:

Apple Game Sprockets SDK

Back to top

LaserWriter 8 version 8.7

LaserWriter 8 provides the ability to print to PostScript printers. The following features have been added to the LaserWriter 8 printer facility:

- Improved ColorSync Integration and Easy of Use with a simplified color matching panel in the print dialog.
- Application Support for Scriptable Printing, allowing specification of printing options when printing from AppleScripts. New APIs have been introduced to support this feature.

  COMPATIBILITY NOTE
  Developers wanting to support the new scriptable printing features will need to make a few minor modifications to their applications.

- It is now possible to create PostScript filters capable of modifying the PostScript data stream and locate them in the PostScript data pipeline between the LaserWriter 8 printer driver and the PostScript device. Furthermore, to allow for configuration of a filter's print job parameters, it is possible for a PostScript filter to add one or more dialog panels to the LaserWriter 8 print job dialog.
- In earlier versions of LaserWriter 8, partial font downloading was only available for fonts with the particular style being requested defined in their type. Now, if the particular style variant is not defined for the font's type it will be synthesized "on the fly" from the plain text style of the font.

Related Materials:

Technote 1178, "Scriptable Printing"
Technote 1179, "PostScript Output Filters for LaserWriter 8.7"
Technote 1177, "Introducing the LaserWriter 8 Driver Version 8.7"
Technote 1144, "Writing Custom Hoses for LaserWriter 8.6"
Technote 1146, "LaserWriter 8 and Fonts"

Back to top

Mac OS Runtime for Java 2.1.4

Mac OS Runtime for Java (MRJ) is Apple's implementation of the Java runtime environment. MRJ allows Java functionality to be embedded in Mac OS applications. Also, applications using MRJ's JManager API are able to run Java applets and applications. MRJ is a runtime environment and does not include development tools. If you are interested in Java development, you need a Java development environment and the MRJ Software Development Kit (MRJ SDK). For more information, see the MRJ developer page.

- Mac OS Runtime for Java (MRJ) 2.1.4 is Bronze certified for running applications built with Oracle Developer 1.6.1 and deployed on Oracle Developer Server using Apple Applet Runner 2.1.4. For information on obtaining Applet Runner 2.1.4 and Oracle Bronze certification, see http://www.apple.com/java/.
- MRJ 2.1.4 addresses several general Java security problems reported to us by Sun Microsystems, fixes a number of memory leaks, and provides solutions to a number of outstanding developer issues.
- MRJ 2.1.4 is compliant with Sun Microsystem's Java 1.1.7 specification.
- MRJ 2.1.4 is compatible with both Swing 1.0.3 and Swing 1.1. For more information or a copy of the Swing software, see the Sun Microsystems web site at http://java.sun.com/. To run Swing-based applets and applications, place the "swingall.jar" file in the MRJClasses folder, in the MRJ Libraries folder in the Extensions folder (inside the System Folder).

Related Materials:

Programming With MRJ Toolkit
Programming With JManager for MRJ
Using JDirect to Access Mac OS Code from Java for MRJ
Using JBindery 2.0.1
Main Java web site
Apple's Java Developer web site
The Mac OS Runtime for Java SDK

Back to top

Multiple Users 1.0

Multiple Users is a multi-user software extension that provides the capability for a number of users to access a single computer. Each user logs in to her own environment that provides her own preferences and some file security.

- Allows users to have their own preferences when they log in. This includes, but not limited to: Preferences, Apple Menu Items, Startup Items, Desktop Items, and Favorites.
- A voice authentication method has been provided for Multiple Users 1.0.

Related Materials:

Technote 1186, "Being a good multiuser citizen"

Back to top

NetSprocket 1.7

NetSprocket provides a fast, lightweight client-server messaging service on top of the Open Transport API that are useful within games played over an AppleTalk or TCP/IP network. NetSprocket includes the following corrections:

- NetSprocket 1.7 is a bug-fix and feature enhancement release. In addition to increasing the stability of the product, this new version adds new APIs that provide client applications with more flexibility in the way that they manage the players and teams within their NetSprocket applications.
- API additions to allow a host to drop players from a game.
- API additions to allow developers the ability to set time-out values.
- Consolidated into a single file rather than two.

Related Materials:

Apple Game Sprockets SDK

Back to top

Network Services Location 1.1

Network Services Location (NSL) allows you to register and search for services on a network in a protocol-independent fashion. New features for NSL 1.1 include:

- LDAP plug-in (support for LDAP protocol; implementation based on Netscape LDAP server).
- NBP plug-in (support for NBP protocol).
- Standard user interface that will allow users to browse for a service via the dialog provided in `NSLStandardGetURL` and then launch an appropriate application, obtained by Internet Config, with the resulting URL.
- URL drag-and-drop support (drag URL from NSL dialog or any browser to desktop, creating a file on the desktop, which when double-clicked will launch the default browser and go directly to that URL).
- Added `OTCfgEncrypt` and `OTCfgDecrypt` routines for encrypting and decrypting ARA passwords.
- Other stability and performance improvements.

Related Materials:
NSL Documentation

Back to top

OpenGL 1.1

OpenGL for Macintosh is Apple's implementation of Silicon Graphics' industry-standard 3D graphics API.

- Mac OS 9 ships with OpenGL 1.1 that supports the OpenGL API v1.1.

Related Materials:
The Apple OpenGL web page.
OpenGL SDK (Found at the Apple OpenGL web page)
The OpenGL web page

Back to top

Open Transport 2.5

Open Transport is the networking technology for Mac OS 9.

- OT has been revised to include the implementation of Mentat Portable Streams version 3.3 and Mentat TCP version 3.5. As a result, clients to the Open Transport API can expect improved performance and stability.

- The new implementation provides faster endpoint creation and disposal.

- Open Transport 2.5 supports a new message that network drivers can send to the network stack. When a driver calls `OTChangePortState(OTPortRef, ref, kOTPortNetworkChange, 0)` it tells Open Transport that the network environment may have changed. If Open Transport is set to obtain its address using DHCP, then it treats this as a hint that it might now be on a different IP network, so it goes through INIT-REBOOT processing to verify that its address is still valid, and obtains a new one if necessary. The AirPort wireless driver uses this call every time it changes its base station connection. A wired Ethernet driver could use this call every time it detects a link state change, indicating that a cable has been connected, and this would allow the Mac to automatically acquire a new IP address whenever it is connected to a new network, without having to sleep or reboot the machine.

- Copy stall problems should not occur any more over TCP/IP connections while transferring large blocks of data.

- The number of Open Transport library files has been reduced to two, thereby reducing clutter in the extensions folder.

- It is now possible to keep a protocol stack open across a sleep cycle.

- Mac OS 9 includes a client/server version of ARA.

- Dial-on-demand performance has been improved.

- Corrects some problems affecting various third-party wireless modems.

- Some internal timeout constants have been adjusted to prevent pausing behavior when attempting to obtain or renew a DHCP lease.

- DHCP packets are now more compatible with Windows NT DHCP servers and other types of DHCP servers.

  COMPATIBILITY NOTE
  This implementation of Open Transport reduces its dependency on ASLM. Developers should avoid using ASLM in their products as it may not be used in future versions of Open Transport (and therefore may not be present in future versions of Mac OS).

  COMPATIBILITY NOTE
  It is possible for ARA to return error code `-7100`. This error code is also defined as an error code returned by the printing software.

- OpenTransport now sends messages to registered clients providing information about changes in the status of the TCP/IP protocol stack. This allows clients to be aware of changes in the stack's state and avoid making inappropriate calls when the stack is down. The new events, listed in Table 1, are sent to registered Open Transport clients. Using these new events together with the `OTInetGetInterfaceInfo` routine (as discussed in Technote TN1145, "Living in a Dynamic TCP/IP Environment") it is now possible for your application to monitor the state of the TCP/IP protocol stack.

Table 1. TCP/IP protocol stack transition event codes.

Constant Name & Value	Description
`kOTStackIsLoading` (OTEventCode) 0x27000001	Sent before Open Transport attempts to load the TCP/IP protocol stack.
`kOTStackWasLoaded` (OTEventCode) 0x27000002	Sent after the TCP/IP stack has been successfully loaded.
`kOTStackIsUnloading` (OTEventCode) 0x27000003	Sent before Open Transport unloads the TCP/IP stack.

In order to receive these messages, a client must register itself with Open Transport by calling `OTRegisterAsClient`. Once registered in this way, the notifier function installed by `OTRegisterAsClient` will be called with the messages listed in Table 1 in the `OTEventCode` parameter. Also, the notifier's `cookie` parameter will identify the protocol stack that sent the message (currently defined as `kOTPRotocolTypeTCPv4` = `'ipv4'`).

When a `kOTStackIsLoading` message is received, the notifier's `OTResult` parameter contains a value indicating the type of hardware link the stack will be using once it has been loaded (these values are listed in Table 2).

Table 2. TCP/IP protocol stack transition event codes.

Constant Name & Value	Description
`kOTHardwareInterfaceIsEthernet` = 1	TCP/IP running on top of Ethernet.
`kOTHardwareInterfaceIs802` = 6	TCP/IP running on top of 802.
`kOTHardwareInterfaceIsPPP` = 7	TCP/IP running on top of a PPP dial-up connection.

COMPATIBILITY NOTE
In the future, other protocols may be revised to send stack transition events to registered Open Transport clients. As such, clients should check the notifier's `cookie` parameter to ensure the event refers to the particular protocol it is monitoring. For example, a client monitoring the TCP/IP protocol would check the cookie parameter to ensure it contains the value `kOTPRotocolTypeTCPv4`.

Related Materials:

Inside Macintosh: Networking With OpenTransport
The OpenTransport web pages
The OpenTransport SDK
The OpenTransport/PPP SDK

Back to top

PlainTalk 2.0.2

PlainTalk provides both speech recognition services and services for the translation of text into audible speech. The Mac OS 9 installation ships with English Text-to-Speech 2.0.2, Plaintalk Speech Recognition 2.0, and Mexican Spanish Text-to-Speech 2.0.2. English Text-to-Speech is installed as part of the default Mac OS 9 installation. Plaintalk Speech Recognition 2.0 can be installed as a custom install option. On some localized OS releases, individual Plaintalk components, may be moved to the CD Extras folder on the System Software CD or removed entirely from the OS.

- Facilities have been provided to allow speakable items to be designated as either recognizable when a particular application is active or recognizable when any application is active.
- Developers can install Speakable Actions for use when their application is active by creating a folder in the Apple Menu Items:Speakable Items:Application Speakable Items: hierarchy. Inside of this folder, add an invisible alias file referring to the application along with items (AppleScript applets or standard applications) associated with voice commands. The items in this folder will become speakable in when the application is active (i.e., is the front-most application).
- The Speech Listener Application enables Applescript writers to listen for specific words as part of their scripts.
- Speakable items commands have been updated to use Mac OS 9 features.
- Built-in commands have been added to allow for application switching and launching based on the running processes, the contents of the "Recent Applications" folder and the contents of the "Favorites" folder.
- Built-in commands allow for launching web browser and email client based on the selections in the Internet control panel.
- A new dynamic help view that shows all available spoken commands - this can be accessed by way of the "Show Speakable commands" speakable command.
- A set of new application speakable items designed specifically for Internet Explorer, Netscape Communicator, and Outlook Express.
- Speaking the names of push buttons, radio buttons, and check boxes works in (most) applications.
- AppleScript terminology for the `SpeakableItems` application is now available.

Related Materials:

The Speech Manager chapter of *Inside Macintosh: Sound*
The Speech Recognition Manager 1.5.1 SDK
The Speech Synthesis Manager 1.5 SDK

Back to top

SoundSprocket 1.7

SoundSprocket extends the functionality of the Sound Manager so that some sound channels appear to be coming from a specific location and distance, and moving, in space. This effect can be presented over speakers, over headphones, or over headphones with head tracking.

- Provides superior 3D sound experience over previous versions.
- API additions have been included so that clients of the OpenGL 3D graphics API can more easily use SoundSprocket.

Related Materials:
Apple Game Sprockets SDK

Back to top

Text Encoding Converter (TEC) 1.5

The Text Encoding Converter enables the conversion of text data from one encoding to another (e.g., Mac OS Roman to Unicode). The following new features are present in the TEC that ships with Mac OS 9:

- A new API, `NearestMacTextEncodings`, has been introduced that allows callers to determine the best possible Macintosh character encoding that can be used to represent any arbitrary character encoding.
- A new routine, `UCGetCharProperty`, has been added to the TEC interfaces. This new routine allows callers to establish character property information for UTF-16 character sequences.
- A new API, `TECSetBasicOptions`, allows callers to set some of the options available in the UnicodeConverter to also be used by way of the TextEncodingConverter interfaces.
- A new Unicode Converter option, `kUnicodeTextRunHeuristics`, has been added. This new option allows for finer control over conversion involving text runs including multiple character sets.
- Support for the Euro currency symbol has been added for the Mac Cyrillic encodings.
- Support for conversion from arbitrary Unicode to maximally decomposed Unicode has been added.
- Fixed problems in building conversions involving Unicode variants, including UTF-8.
- Fixed bugs related to booting and initialization, especially on localized systems.
- Fixed bugs relating to locking tables for use at interrupt level.
- Added support for the encodings ISO 8859-3 (Latin-3), ISO 8859-4 (Latin-4), ISO 8859-15

(Latin-9), Windows CP 1257 (Baltic Rim), Windows CP 1258 (Vietnamese).

Related Materials:
T EC Documentation
Text Encoding Converter (TEC) 1.5 SDK

Back to top

Apple Menu Items

Items in the Apple Menu Items folder appear in the Apple Menu.

Network Browser

The Network Browser application provides a convenient and easy to use graphical user interface for connecting to network-based file servers.

- Network Browser now supports FTP connections to servers on TCP/IP networks.
- Browsing services are now based on Network Services Location (NSL), allowing users to browse services on the internet.

Related Materials:
The Network Services Location (NSL) section.

Back to top

Sherlock 2

Sherlock provides search facilities for Mac OS computers. In addition to the traditional capabilities of the Find application, Sherlock supports searching by content and Internet searches.

- Adds the concept of Sherlock Channels, allowing internet search sites to be organized into discrete categories. Categories include People, Shopping, News, and general Internet searches.
- Find by Content and Find File have been integrated into a single Files channel, allowing users to perform more complex searches than in previous versions of Sherlock.
- The Sherlock application now provides a simple user interface for selecting which languages are to be supported for indexing. Thus, a user could select English, French, German, and Italian, for example, or German, Polish, and Russian. Of course, selecting additional languages will increase the size of index files.

Related Materials:

The Find By Content section.
Technote TN1141, "Extending and Controlling Sherlock"
Technote TN1180, "Sherlock's Find By Content Library"
Technote TN1181, "Sherlock's Find by Content Text Extractor Plug-ins"

Back to top

Utilities

Utilities include items useful for maintaining your computer. Normally, these items are placed in the Utilities folder on your startup volume when you install your system software, but some of them can be found in the "Utilities" folder on the Install CD.

Package First Aid

The Utilities folder on the Mac OS 9 CD contains an application called Package First Aid. This utility allows you to reset folders that have been incorrectly configured as application packages.

Back to top

Drive Setup

Drive Setup provides facilities for updating hard disk drivers and for formatting hard drives.

- Commands have been added allowing volumes to be formatted as UFS and other common UNIX file system types.

Back to top

Apple Extras

Apple Extras include a number of items placed in on your hard drive during system installs. These items may include utilities, tools, or programs meant for specific purposes.

Palm Desktop 2.5

Palm Desktop is a updated version of Claris Organizer that Palm Computing acquired from Apple. It provides Personal Information Management (PIM) functionality and synchronization to Palm OS-based devices.

- The primary features are unchanged from the currently shipping MacPac 2.1.

Back to top

Items no longer supported

The following items are not supported in Mac OS 9. Attempts to install these components in Mac OS 9 and may lead to undesirable results. These components have not been tested with Mac OS 9.

Apple Telecom software

Apple Telecom software is not supported in Mac OS 9.

Color StyleWriter 4000 series printers

Color StyleWriter 4000 series printers are not compatible with Mac OS 9, and will be removed by the installer if it is present in a System Folder that is being updated to Mac OS 9.

Energy Saver API

The Energy Saver API, as documented in TN1086, "Power Management & The Energy Saver API," is not supported with Mac OS 9. The Energy Saver extension is still present in among the control panels - only the API is no longer supported.

COMPATIBILITY NOTE
Applications attempting to utilize this API in Mac OS 9 will not work.

LaserWriter 8f

LaserWriter 8f is not compatible with Mac OS 9 and will be removed by the installer if it is present in a System Folder that is being updated to Mac OS 9.

QuickDraw GX

QuickDraw GX is not supported in Mac OS 9 (QuickDraw GX printing was phased out in Mac OS 8.0. Graphics, line layout, and typography are now also phased out).

Back to top

Downloadables

Acrobat version of this Note (204K).

To contact us, please use the Contact Us page.

Updated: 24-April-2000

Technotes | Contents
Previous Technote | Next Technote