

## Using the Gestalt Manager

The **Gestalt Manager** includes three functions-**Gestalt**, **NewGestalt**, and **ReplaceGestalt**. You can use the **Gestalt** function to obtain information about software or hardware components available on the current machine. You can use **NewGestalt** to register new software modules (such as drivers and patches) with the Operating System. Use **ReplaceGestalt** to replace the function associated with a particular selector code by some other function.

**Note:** Most applications do not need to use either **NewGestalt** or **ReplaceGestalt**.

### Determining Features of the Operating Environment

When your application needs information about a specific software or hardware feature that can be provided by the **Gestalt** function, your application can pass **Gestalt** a selector code (or selector) as one of the parameters. The selector code is simply an indication of what information your application currently needs. There are two types of selector codes-predefined selector codes that are always recognized by **Gestalt**, and application-defined selector codes that applications may register with **Gestalt** by calling the **NewGestalt** function.

If **Gestalt** can determine the requested information, it returns that information in its second parameter, known as the response parameter. If **Gestalt** cannot obtain the desired information, it returns a result code indicating the cause of the error; in that case, the value of the response parameter is undefined. You should *always* check the result code returned by **Gestalt** to make sure that the response parameter contains meaningful information.

**Note:** When passed one of the predefined selector codes, **Gestalt** does not move or purge memory and therefore may be called at any time, even at interrupt time. However, selector functions associated with application-defined selector codes may move or purge memory, and applications can alter **Gestalt**'s predefined selector functions. As a result, it is safest to assume that **Gestalt** might always move or purge memory. The **NewGestalt** function may move memory and should not be called at interrupt time.

There are two types of predefined selector codes: codes that return information that your application can use to guide its actions (known as environmental selectors), and codes that provide information only and should never be used as an indication of some feature's existence (known as informational selectors).

It is particularly important that you understand the difference between environmental and informational selectors. The response returned by **Gestalt** when it is passed an informational selector is for your (or the user's) edification only and should *never* be used by your application as a means of determining whether some specific software or hardware feature is available. For example, you can use **Gestalt** to test for the version of the ROM installed on a particular machine, but you should never use this information to guide any of your application's actions. Routines you expect to be in ROM may actually be in RAM; hence, you cannot determine that some routine usually

found in ROM is not present simply by looking at the ROM version. Also, routines contained in ROM may have been patched by the system at startup time, in which case the system might not have the features that you think it has on the basis of the reported ROM version. Similar remarks apply to other informational selectors such as ROM size, machine type, and System file version.

You can use the following environmental selectors to determine information about the operating environment.

|                                   |                                   |
|-----------------------------------|-----------------------------------|
| <u>gestaltAddressingModeAttr</u>  | addressing mode attributes        |
| <u>gestaltAliasMgrAttr</u>        | <b>Alias Mgr</b> attributes       |
| <u>gestaltAppleEventsAttr</u>     | Apple events attributes           |
| <u>gestaltAppleTalkVersion</u>    | AppleTalk version                 |
| <u>gestaltAUXVersion</u>          | A/UX version if present           |
| <u>gestaltConnMgrAttr</u>         | Connection Mgr attributes         |
| <u>gestaltCRMAttr</u>             | Comm Resource Mgr attrs           |
| <u>gestaltCTBVersion</u>          | <b>Comm Toolbox</b> version       |
| <u>gestaltBAccessMgrAttr</u>      | <b>Data Access Mgr</b> attrs      |
| <u>gestaltDITLExtAttr</u>         | <b>Dialog Mgr</b> extensions      |
| <u>gestaltEasyAccessAttr</u>      | Easy Access attributes            |
| <u>gestaltEditionMgrAttr</u>      | <b>Edition Manager</b> attributes |
| <u>gestaltExtToolboxTable</u>     | Ext Toolbox trap table base       |
| <u>gestaltFindFolderAttr</u>      | FindFolder attributes             |
| <u>gestaltFontMgrAttr</u>         | <b>Font Mgr</b> attributes        |
| <u>gestaltFPUType</u>             | FPU type                          |
| <u>gestaltFSAttr</u>              | file-system attributes            |
| <u>gestaltFXfrMgrAttr</u>         | File Transfer Mgr attrs           |
| <u>gestaltHardwareAttr</u>        | hardware attributes               |
| <u>gestaltHelpMgrAttr</u>         | <b>Help Mgr</b> attributes        |
| <u>gestaltKeyboardType</u>        | keyboard type                     |
| <u>gestaltLogicalPageSize</u>     | logical page size                 |
| <u>gestaltLogicalRAMSize</u>      | logical RAM size                  |
| <u>gestaltLowMemorySize</u>       | low-memory area size              |
| <u>gestaltMiscAttr</u>            | miscellaneous attributes          |
| <u>gestaltMMUType</u>             | MMU type                          |
| <u>gestaltNotificationMgrAttr</u> | <b>Notification Mgr</b> attrs     |
| <u>gestaltNuBusConnectors</u>     | NuBus connector bitmap            |
| <u>gestaltOSAttr</u>              | O/S attributes                    |
| <u>gestaltOSTable</u>             | O/S trap table base               |
| <u>gestaltParityAttr</u>          | parity attributes                 |
| <u>gestaltPhysicalRAMSize</u>     | physical RAM size                 |
| <u>gestaltPopupAttr</u>           | pop-up CEF attributes             |
| <u>gestaltPowerMgrAttr</u>        | <b>Power Mgr</b> attributes       |
| <u>gestaltPPCToolboxAttr</u>      | <b>PPC Toolbox</b> attributes     |
| <u>gestaltProcessorType</u>       | processor type                    |
| <u>gestaltQuickdrawVersion</u>    | <b>QuickDraw</b> version          |
| <u>gestaltResourceMgrAttr</u>     | <b>Resource Mgr</b> attributes    |
| <u>gestaltScriptCount</u>         | # of active script systems        |
| <u>gestaltScriptMgrVersion</u>    | <b>Script Mgr</b> version         |
| <u>gestaltSerialAttr</u>          | serial hardware attributes        |
| <u>gestaltSoundAttr</u>           | sound attributes                  |
| <u>gestaltStandardFileAttr</u>    | <b>Standard File</b> attributes   |
| <u>gestaltStdNBPAttr</u>          | StandardNBP attributes            |
| <u>gestaltTermMgrAttr</u>         | Terminal Mgr attributes           |
| <u>gestaltTextEditVersion</u>     | <b>TextEdit</b> version           |

|                              |                           |
|------------------------------|---------------------------|
| <u>gestaltTimeMgrVersion</u> | <b>Time Mgr</b> version   |
| <u>gestaltToolboxTable</u>   | Toolbox trap table base   |
| <u>gestaltVersion</u>        | <b>Gestalt</b> version    |
| <u>gestaltVMAttr</u>         | virtual memory attributes |

The following informational selectors are provided for informational purposes only. You can display the information returned when using these selectors, but you should never use this information as an indication of what software features or hardware may be available.

|                             |                          |
|-----------------------------|--------------------------|
| <u>gestaltMachineIcon</u>   | machine ICON/cicn res ID |
| <u>gestaltMachineType</u>   | machine type             |
| <u>gestaltROMSize</u>       | ROM size                 |
| <u>gestaltROMVersion</u>    | ROM version              |
| <u>gestaltSystemVersion</u> | System file version      |

**Interpreting Gestalt Responses** explains the exact meaning of each of these selectors and of the values returned by **Gestalt** in each case.

### Determining Whether Gestalt Is Available

Because the **Gestalt Manager** currently exists only in system software versions 6.0.4 and later (and in ROM on the Macintosh IIci, the Macintosh Portable, and later machines), you should make certain that it is actually available before attempting to call it. You can do this by using the **TrapAvailable** function defined previously in **Determining Whether a Trap Is Available** under **About Compatibility**. The Listing below uses that function to determine whether the **Gestalt Manager** is available.

#### Determining whether **Gestalt** is available

```
// Determining whether Gestalt is available.
// See About Compatibility for the definition of TrapAvailable.
#include <Traps.h>

#define _Gestalt 0xA1AD

Boolean GestaltAvailable(void)
{
    return TrapAvailable(_Gestalt);
}
```

**Note:** If you are using THINK C 5.0 or later, THINK Pascal 4.0 or later, or the MPW development system version 3.2 or later, then you do not need to perform this check because these versions provides glue routines that allow you to call **Gestalt**, **NewGestalt**, and **ReplaceGestalt** even if they are not in ROM or in the System file. However, if you are programming in assembly language, this glue is not provided (and you still need to check that the trap is available before calling it). See the individual descriptions of **Gestalt**, **NewGestalt**, and **ReplaceGestalt** for more information about the behavior of the glue routines under systems which do not support the **Gestalt Manager**.

If you need to know at several different places in your application whether **Gestalt** is available, it may be more efficient to define a global **Boolean** variable that you can test before calling **Gestalt**. The listing below illustrates how to do this. Once again, this code uses the **TrapAvailable** function defined earlier.

### Using **Gestalt** to determine the **Time Manager** version

```
// Using Gestalt to determine the Time Manager version
// See About Compatibility for the definition of TrapAvailable
#include <Traps.h>
#include <GestaltEqu.h>

#define _Gestalt 0xA1AD

Boolean    gHasGestalt;  // true if Gestalt is implemented

main()
{
    OSErr myErr;
    long   myFeature;      // will receive Gestalt return value

    gHasGestalt = TrapAvailable(_Gestalt);

    if (gHasGestalt) {    // Not necessary under development systems
                        // which contain Gestalt glue
        myErr = Gestalt(gestaltTimeMgrVersion, &myFeature);
        if (myErr != noErr)
            DoError(myErr);
    }
}
```

This sample code returns (in the myFeature parameter) the version of the **Time Manager** available on the current machine. Before using that information, however, you should test the result code to make sure that **Gestalt** was able to determine the requested information.

### Interpreting Gestalt Responses

When your application calls **Gestalt** to get information about the operating environment, the meaning of the value that **Gestalt** returns in the response parameter depends on the selector code with which it was called. For example, if you call **Gestalt** using the gestaltTimeMgrVersion selector, it returns a version code in the low-order byte of the response parameter. In this case, a returned value of 3 indicates that the extended **Time Manager** is available.

In almost all cases, the last few characters in the selector's symbolic name form a suffix that indicates what type of value you can expect **Gestalt** to return in the response parameter. For example, if the final characters in a **Gestalt** selector are Size, then **Gestalt** returns a size in the response parameter. The following list shows the meaningful suffixes.

| Suffix | Meaning |
|--------|---------|
|--------|---------|

|         |   |
|---------|---|
| Attr    | The returned value is a range of 32 bits, the meaning of which must be determined by comparison with a list of constants. Note that bit 0 is the least significant bit of the long word.  |
| Count   | The returned value is a number indicating how many of the indicated type of item exist.   |
| Size    | The returned value is a size. Sizes reported by <b>Gestalt</b> are usually in bytes.  |
| Table   | The returned value is the base address of a table.  |
| Type    | The returned value is an index describing a particular type of feature.   |
| Version | The returned value is a version number. Implied decimal points may separate digits of the returned value. For example, a value of 0x0605 returned in response to the <u>gestaltSystemVersion</u> selector indicates that system software system 6.0.5 is present. |

Selectors that have the suffix Attr deserve special attention; they cause **Gestalt** to return a bit field that your application must interpret in order to determine whether a desired feature is present. For example, the gestaltOSAttr selector requests information about a number of Operating System features. To determine whether a particular Operating System feature is available, you need to read the appropriate bit in the response parameter, as the Listing below illustrates.

### Interpreting a bit field response

```
// Interpreting a bit field response
#include <GestaltEqu.h>

short  myBit, myErr;
long   myFeature;      // will receive Gestalt return value
Boolean gHasGestalt;

if (gHasGestalt) {
    myErr = Gestalt(gestaltOSAttr, &myFeature);
    if (myErr != noErr)
        DoError(myErr);
    else {
        myBit = gestaltTempMemSupport;
        if (BitTst(&myFeature, 31-myBit))
            printf("temporary memory support available\n");
        else
            printf("temporary memory support not available\n");
    }
}
```

This code uses the Toolbox utility function **BitTst** to determine whether the appropriate bit in **Gestalt**'s response is set to 1. Notice that because bit numbering with **BitTst** is the opposite of the usual MC680x0 numbering

scheme used by **Gestalt**, the bit to be tested must be subtracted from 31. Also, the first parameter to **BitTst** is a pointer to a byte; hence the use of the & operator. Your development system may have other ways of testing the appropriate bit.

### Interpreting Responses to Environmental Selectors

**Gestalt** returns one of the following responses when passed a predefined environmental selector.

| Selector                         | Meaning  |
|----------------------------------|--|
| <u>gestaltAddressingModeAttr</u> | Returns information about the current addressing mode.   |
| <u>gestalt32BitAddressing</u>    |  |
| <u>gestalt32BitSysZone</u>       |  |
| <u>gestalt32BitCapable</u>       | The <u>gestalt32BitAddressing</u> attribute indicates that the machine started up with 32-bit addressing. The <u>gestalt32BitSysZone</u> attribute indicates that the system heap has 32-bit clean block headers (regardless of the type of addressing the machine started up in). See the <b>Memory Manager</b> for more information about 32-bit addressing.   |
| <u>gestaltAliasMgrAttr</u>       | Returns information about the <b>Alias Manager</b> .   |
| <u>gestaltAliasMgrPresent</u>    |  |
| <u>gestaltAppleEventsAttr</u>    | Returns information about Apple events.  |
| <u>gestaltAppleEventsPresent</u> |  |
| <u>gestaltAppleTalkVersion</u>   | Returns the version number of the AppleTalk driver currently installed. In particular, it returns the version number of the .MPP driver. The version number is placed into the low-order byte of the result, so you should ignore the three high-order bytes of the result. If an AppleTalk driver is not currently open, the response parameter is 0. The driver does not open until the user requests a network service (for example, by running the Chooser). |
| <u>gestaltAUXVersion</u>         | Returns the version of A/UX if it is currently executing. The result is placed into the lower word of the response parameter. If A/UX is not executing, <b>Gestalt</b> returns <u>gestaltUnknownErr</u> .  |
| <u>gestaltConnMgrAttr</u>        | Returns information about the Connection Manager.  |

|                                  |  |
|----------------------------------|--|
| <u>gestaltConnMgrPresent</u>     |  |
| <u>gestaltConnMgrCMSearchFix</u> |  |
|                                  | The <u>gestaltConnMgrCMSearchFix</u> bit flag indicates whether a fix is present that allows the CMAAddSearch routine to work over the mAttn channel.  |
| <u>gestaltCRMAttr</u>            | Returns information about the Communications Resource Manager.   |
|                                  | <u>gestaltCRMPresent</u>   |
| <u>gestaltCTBVersion</u>         | Returns the version number of the <b>Communications Toolbox</b> .  |
| <u>gestaltDBAccessMgrAttr</u>    | Returns information about the <b>Data Access Manager</b>   |
|                                  | <u>gestaltDBAccessMgrPresent</u>   |
| <u>gestaltDITLExtAttr</u>        | Returns information about the <b>Dialog Manager</b> .  |
|                                  | <u>gestaltDITLExtPresent</u>   |
|                                  | If this flag bit is TRUE, then the <b>Dialog Manager</b> extensions are available. See "Manipulating Dialog Item Lists" for details about the <b>Dialog Manager</b> extensions included in system 7.0. |
| <u>gestaltEasyAccessAttr</u>     | Returns information about the status of Easy Access.   |
|                                  | <u>gestaltEasyAccessOff</u><br><u>gestaltEasyAccessOn</u><br><u>gestaltEasyAccessSticky</u><br><u>gestaltEasyAccessLocked</u>  |
| <u>gestaltEditionMgrAttr</u>     | Returns information about the <b>Edition Manager</b> .   |
|                                  | <u>gestaltEditionMgrPresent</u>  |
| <u>gestaltExtToolboxTable</u>    | Returns the base address of the extended Toolbox trap table.   |
| <u>gestaltFindFolderAttr</u>     | Returns information about the <b>FindFolder</b> function.  |
|                                  | <u>gestaltFindFolderPresent</u>  |
| <u>gestaltFontMgrAttr</u>        | Returns information about the <b>Font Manager</b> .  |

|                            |   |
|----------------------------|---|
|                            | <u>gestaltOutlineFonts</u>  |
| <u>gestaltFPUType</u>      | Returns a value that indicates the type of floating-point coprocessor currently installed, if any.  |
|                            | <u>gestaltNoFPU</u><br><u>gestalt68881</u><br><u>gestalt68882</u>   |
| <u>gestaltFSAttr</u>       | Returns information about the file system.  |
|                            | <u>gestaltFullExtFSDispatching</u><br><u>gestaltHasFSSpecCalls</u>  |
|                            | The bit <u>gestaltFullExtFSDispatch</u> indicates that all the routines selected through the <u>_HFSDispatch</u> macro are available to external file systems.  |
| <u>gestaltFXfrMgrAttr</u>  | Returns information about the <b>File Transfer Manager</b> .  |
|                            | <u>gestaltFXfrMgrPresent</u>  |
| <u>gestaltHardwareAttr</u> | Returns information about the hardware configuration of the machine.  |
|                            | <u>gestaltHasVIA1</u><br><u>gestaltHasVIA2</u><br><u>gestaltHasASC</u><br><u>gestaltHasSCC</u><br><u>gestaltHasSCSI</u>   |
| <u>gestaltHelpMgrAttr</u>  | Returns information about the <b>Help Manager</b> .   |
|                            | <u>gestaltHelpMgrPresent</u>  |
| <u>gestaltKeyboardType</u> | Returns a value that indicates the type of keyboard that is currently attached to the system.   |
|                            | <u>gestaltMacKbd</u><br><u>gestaltMacAndPad</u><br><u>gestaltMacPlusKbd</u><br><u>gestaltExtADBKbd</u><br><u>gestaltStdADBKbd</u><br><u>gestaltPrtblADBKbd</u><br><u>gestaltPrtblISOKbd</u><br><u>gestaltStdISOADBKbd</u><br><u>gestaltExtISOADBKbd</u><br><u>gestaltADBKbdII</u><br><u>gestaltADBIKbdII</u><br><u>gestaltPwrBookADBKbd</u><br><u>gestaltPwrBookISOADBKbd</u> |



|                                   |   |
|-----------------------------------|---|
|                                   | If the Apple Desktop Bus is in use, there may be multiple keyboards or other ADB devices attached to the machine. <b>Gestalt</b> returns the type of the keyboard on which the last keystroke occurred.   |
| <u>gestaltLogicalPageSize</u>     | Returns the logical page size. This value is an unknown on 68000-based machines because such machines do not have logical pages. On those machines, <b>Gestalt</b> returns an error.  |
| <u>gestaltLogicalRAMSize</u>      | Returns the amount of logical memory available. This value is the same as that returned by <u>gestaltPhysicalRAMSize</u> when virtual memory is not installed. On some machines, however, this value might be less than the value returned by <u>gestaltPhysicalRAMSize</u> because some RAM may be used by the video display and the Operating System. |
| <u>gestaltLowMemorySize</u>       | Returns the size (in bytes) of the low-memory area. The low-memory area is used for vectors, global variables, and dispatch tables.   |
| <u>gestaltMiscAttr</u>            | Returns information about miscellaneous pieces of the Operating System or hardware configuration.   |
|                                   | <u>gestaltScrollingThrottle</u><br><u>gestaltSquareMenuBar</u>  |
| <u>gestaltMMUType</u>             | Returns a value that indicates the type of MMU currently installed, if any.<br><br><u>gestaltNoMMU</u><br><u>gestaltAMU</u><br><u>gestalt68851</u><br><u>gestalt68030MMU</u>  |
| <u>gestaltNotificationMgrAttr</u> | Returns information about the <b><u>Notification Manager</u></b> .  |
|                                   | <u>gestaltNotificationPresent</u>   |
| <u>gestaltNuBusConnectors</u>     | Returns information about the NuBus slot connector locations. The value returned is a bitmap. For example, the value returned on a Macintosh II would have bits 9 through E set, indicating that 6 NuBus slots are present (having locations 9 through E).  |
| <u>gestaltOSAttr</u>              | Returns general information about the Operating System, such as whether temporary memory handles are real handles. The low-order bits of the response parameter are interpreted   |

as bit flags. A flag is set to 1 to indicate that the corresponding feature is available. Currently, the following bits are significant:

gestaltSysZoneGrowable  
gestaltLaunchCanReturn  
gestaltLaunchFullFileSpec  
gestaltLaunchControl  
gestaltTempMemSupport  
gestaltRealTempMemory  
gestaltTempMemTracked  
gestaltIPCSupport  
gestaltSysDebuggerSupport

See the **Memory Manager** section for a full explanation of the temporary memory features, and see the **Process Manager** section for a full explanation of the launch control features.

gestaltOSTable

Returns the base address of the Operating System trap table.

gestaltParityAttr

Returns information about the parity-checking abilities of the machine.

gestaltHasParityCapability  
gestaltParityEnabled

Note that parity is not considered to be enabled unless *all* installed memory is parity RAM.

gestaltPhysicalRAMSize

Returns the number of bytes of physical RAM currently installed.

gestaltPopupAttr

Returns information about the pop-up control definition.

gestaltPopupPresent

If the gestaltPopupPresent bit is set, the system 7.0 pop-up control definition procedure is present. See **Creating Pop-Up Menus** under

**Running in System 7.0**

for details about creating pop-up menus.

gestaltPowerMgrAttr

Returns information about the **Power Manager**, if present.

gestaltPMgrExists  
gestaltPMgrCPUIdle  
gestaltPMgrSCC  
gestaltPMgrSound

gestaltPPCToolboxAttr

Returns information about the capabilities of the **PPC Toolbox**.

gestaltPPCToolboxPresent

|                                |   |
|--------------------------------|---|
| <u>gestaltProcessorType</u>    | Returns a value that indicates the type of processor that is currently running.<br><br><u>gestalt68000</u> = 1;<br><u>gestalt68010</u> = 2;<br><u>gestalt68020</u> = 3;<br><u>gestalt68030</u> = 4;<br><u>gestalt68040</u> = 5;   |
| <u>gestaltQuickdrawVersion</u> | Returns a 2-byte value indicating the version of <b>QuickDraw</b> currently present. The high-order byte of that number represents the major revision number, and the low-order byte represents the minor revision number. For example, the Macintosh IIci contains <b>QuickDraw</b> version 2.01 in ROM; on that machine, <b>Gestalt</b> returns the value 0x0201.<br><br><u>gestaltOriginalQD</u> = 0x000;<br><u>gestaltOriginalQD1</u> = 0x001;<br><u>gestalt8BitQD</u> = 0x100;<br><u>gestalt32BitQD</u> = 0x200;<br><u>gestalt32BitQD11</u> = 0x210;<br><u>gestalt32BitQD12</u> = 0x220;<br><u>gestalt32BitQ13</u> = 0x230;<br><br>Values having a major revision number of 1 or 2 indicate that <b>Color QuickDraw</b> is available, in either the 8-bit or 32-bit version. These results do not, however, indicate whether a color monitor is attached to the system. You need to use high-level <b>QuickDraw</b> routines to obtain that information. |
| <u>gestaltResourceMgrAttr</u>  | Returns information about the capabilities of the <b>Resource Manager</b> .<br><br><u>gestaltPartialRsrcs</u>   |
| <u>gestaltScriptCount</u>      | Returns the number of script systems currently active.  |
| <u>gestaltScriptMgrVersion</u> | Returns the version number of the <b>Script Manager</b> .   |
| <u>gestaltSerialAttr</u>       | Returns information about the serial hardware of the machine (such as whether or not the GPIa line is connected and can be used for external clocking).<br><br><u>gestaltGPIaToDCa</u><br><u>gestaltGPIaToRTxCa</u><br><u>gestaltGPIaToDCb</u>  |
| <u>gestaltSoundAttr</u>        | Returns information about the sound capabilities of the machine.  |

gestaltStereoCapability  
gestaltStereoMixing  
gestaltSoundIOMgrPresent  
gestaltBuiltInSoundInput  
gestaltHasSoundInputDevice

If the bit gestaltStereoCapability is TRUE, the available hardware can play stereo sounds. The bit gestaltStereoMixing indicates that the sound hardware of the machine mixes both left and right channels of stereo sound into a single audio signal for the internal speaker. The gestaltSoundIOMgrPresent bit indicates that the new sound input routines are available, and the gestaltBuiltInSoundInput bit indicates that a built-in sound input device is available. The gestaltHasSoundInputDevice bit indicates that some sound input device is available.

gestaltStandardFileAttr

Returns information about the **Standard File Package**.

gestaltStandardFile58

If this flag bit is set to 1, you can call the four new procedures **StandardPutFile**, **StandardGetFile**, **CustomPutFile**, and **CustomGetFile**. (The name of the constant reflects the enabling of selectors 5-8 on the trap macro that handles the **Standard File Package**.)

gestaltStdNBPAAttr

Returns information about the call **StandardNBP** (Name-Binding Protocol).

gestaltStdNBPPresent

gestaltTermMgrAttr

Returns information about the **Terminal Manager**.

gestaltTermMgrPresent

gestaltTextEditVersion

Returns a value that indicates which version of **TextEdit** is present.

gestaltTE1  
gestaltTE2  
gestaltTE3  
gestaltTE4  
gestaltTE5

See **TextEdit** for further information on the capabilities of the enhanced versions of **TextEdit**.

gestaltTimeMgrVersion

Returns a value that indicates the version of

the **Time Manager** that is present.

gestaltStandardTimeMgr

gestaltRevisedTimeMgr

gestaltExtendedTimeMgr

See the **Time Manager** for a complete explanation of the capabilities of each of these three versions.

gestaltToolboxTable

Returns the base address of the Toolbox trap table.

gestaltVersion

Returns the version of **Gestalt**. The current version is 1, corresponding to a returned value of 0x0001.

gestaltVMAttr

Returns information about virtual memory.

gestaltVMPresent

### Interpreting Responses to Informational Selectors

**Gestalt** returns the following responses when passed a predefined informational selector.

**Warning:** Never infer the existence of certain hardware or software features from the responses that **Gestalt** returns to your application when you pass it these selectors.

| Selector                   | Meaning  |
|----------------------------|--|
| <u>gestaltMachineIcon</u>  | Returns an icon family resource ID for the current type of Macintosh.  |
| <u>gestaltMachineType</u>  | Returns one of the following values, indicating the type of machine on which the application is currently running. |
| <u>gestaltClassic</u>      | Macintosh 128K   |
| <u>gestaltMacXL</u>        | Macintosh XL   |
| <u>gestaltMac512KE</u>     | Macintosh 512KE  |
| <u>gestaltMacPlus</u>      | Macintosh Plus   |
| <u>gestaltMacSE</u>        | Macintosh SE   |
| <u>gestaltMacII</u>        | Macintosh II   |
| <u>gestaltMacIIx</u>       | Macintosh IIx  |
| <u>gestaltMacIIcx</u>      | Macintosh IIcx   |
| <u>gestaltMacSE030</u>     | Macintosh SE/30  |
| <u>gestaltPortable</u>     | Macintosh Portable   |
| <u>gestaltMacIIci</u>      | Macintosh IIci   |
| <u>gestaltMacIIfx</u>      | Macintosh IIfx   |
| <u>gestaltMacClassic</u>   | Macintosh Classic  |
| <u>gestaltMacIIsi</u>      | Macintosh IIsi   |
| <u>gestaltMacLC</u>        | Macintosh LC   |
| <u>gestaltQuadra900</u>    | Macintosh Quadra900  |
| <u>gestaltPowerBook170</u> | Macintosh PowerBook170   |
| <u>gestaltQuadra700</u>    | Macintosh Quadra700  |

|                            |                         |
|----------------------------|-------------------------|
| <u>gestaltClassicII</u>    | Macintosh ClassicII     |
| <u>gestaltPowerBook100</u> | Macintosh PowerBook100  |
| <u>gestaltPowerBook140</u> | Macintosh PowerBook140  |
| <u>gestaltMacLC</u>        | Macintosh LC            |
| <u>gestaltQuadra900</u>    | Macintosh Quadra 900    |
| <u>gestaltPowerBook170</u> | Macintosh PowerBook 170 |
| <u>gestaltQuadra700</u>    | Macintosh Quadra 700    |
| <u>gestaltClassicII</u>    | Macintosh Classic II    |
| <u>gestaltPowerBook100</u> | Macintosh PowerBook 100 |
| <u>gestaltPowerBook140</u> | Macintosh PowerBook 140 |
| <u>gestaltQuadra950</u>    | Macintosh Quadra 950    |
| <u>gestaltMacLCII</u>      | Macintosh LCII          |
| <u>gestaltPowerBook145</u> | Macintosh PowerBook 145 |

Macintosh Technical Note #129 points out the following: Under System 7.0.1 **Gestalt** will return a value of gestaltMacLC for the Macintosh LC II. The LC II can be distinguished from the LC by calling **Gestalt** with the gestaltProcessorType selector. A response of gestalt68020 will be returned for the LC and a response of gestalt68030 will be returned for the LC II. However, under System 7.1 and later **Gestalt** will return the gestaltMacLCII response. Responses for the PowerBook 145 exhibit the same behavior. Under System 7.0.1, a response of gestaltPowerBook140 will be returned. The PowerBook 145 may be distinguished in the same way as the LC II, since it has a 68030 processor while the PowerBook 140 has a 68020 processor. Under all systems later than 7.0.1, the correct response of gestaltPowerBook145 will be returned.

To obtain a string containing the machine's name, you can pass the returned value to **GetIndString** as an index into the resource of type 'STR#' in the System file having resource ID defined by the constant kMachineNameStrID.

#### kMachineNameStrID

|                             |  |
|-----------------------------|--|
| <u>gestaltROMSize</u>       | Returns the size of the installed ROM. The value is returned in a word.  |
| <u>gestaltROMVersion</u>    | Returns the version number of the installed ROM.   |
| <u>gestaltSystemVersion</u> | Returns the version number of the currently active System file. This number is represented as two byte-long numbers. For example, if your application is running in system 6.0.4, then <b>Gestalt</b> returns the value 0x0604. You should ignore the high-order word of the returned value. |

### **Adding Gestalt Selectors**

You can add a new selector code to those already understood by **Gestalt** by calling the **NewGestalt** function. The **NewGestalt** function requires two parameters. The first parameter is the new selector to be registered. The

second parameter is the address of a selector function. **Gestalt** executes the selector function when it needs to determine what value to pass back when it is called after the new selector code.

The selector code is a four-character sequence of type `OStype`. For example, Carl's Object-Oriented Linker might register itself using the selector code 'COOL'. If you have registered a creator string with Apple, you are strongly encouraged to use that sequence as your selector code.

**Note:** Apple reserves for its own use all four-character sequences consisting solely of lowercase letters and nonalphabetic ASCII characters.

The selector function whose address you specify when registering a new **Gestalt** selector code can be any function that resides in the system heap and whose calling syntax conforms to that defined in **Specifying Gestalt Selector Functions** below. The Listing below illustrates how to install a simple function into the system heap and pass its address to **NewGestalt**.

```
// Installing a selector function into the system heap
// #inclusion of <MacHeaders.h> assumed

#include <GestaltEqu.h>           // for Gestalt
#include <Traps.h>                // for trap numbers
#include <stdio.h>                // for printing

#define mySelector                'COOL' // Gestalt function selector
#define gstFuncRsrcType          'GDEF' // Gestalt function resource type
#define gstFuncRsrcID            128    // Gestalt function resource ID

main()

{
    OSErr    gestaltErr;           // error returned by Gestalt
    Handle    gstFuncHandle;       // handle to Gestalt function
    ProcPtr    oldGestaltFunc;     // pointer to old function

    // first make sure that Gestalt is available

    if (!TrapAvailable(_Gestalt)) {
        printf("Gestalt is not implemented.\n");
        Exit(1);
    }

    // load Gestalt function resource into system heap

    gstFuncHandle = GetResource(gstFuncRsrcType, gstFuncRsrcID);
    if (gstFuncHandle == nil) {
        printf("Could not load Gestalt function resource.\n");
        Exit(1);
    }

    // detach it from the resource map so it stays around

    DetachResource(gstFuncHandle);
```

```

// add the new selector; first assume that it doesn't already exist

gestaltErr = NewGestalt(mySelector, (ProcPtr)(*gstFuncHandle));
if (gestaltErr != noErr){
    printf("Could not add as a new selector.\n");

    // try to replace existing selector

    gestaltErr = ReplaceGestalt(mySelector,
        (ProcPtr)(*gstFuncHandle), &oldGestaltFunc);
    if (gestaltErr != noErr) {
        printf("Could not replace selector either.\n");
        Exit(1);
    }
}
printf("Selector installed.\n");
DisposHandle(gstFuncHandle);
}

```

You can ensure that the new **Gestalt** selector function is installed into the system heap by defining it as a resource (in this case, of type 'GEF') whose resource attributes are resSysHeap and resLocked (in other words, lock the resource into the system heap). You can set the resource attributes by using ResEdit.

The Listing below shows the actual function definition.

```

// Defining a new Gestalt function
#include <GestaltEqu.h>                                // for Gestalt

#define    myResult    0x87654321                      // Gestalt function response

OSErr gestaltCool(OSType gestaltSelector, long *gestaltResponse)

{
    *gestaltResponse = myResult;
    return noErr;
}

```

Because the new selector function resides in the system heap, **Gestalt** recognizes and responds to the new selector until the machine restarts, even if your application terminates before that time. As a result, you might want your selector function to determine whether your application is still running before returning a value to **Gestalt**. If your application has terminated, the selector function should return an error.

Note that if you try to register a selector that has already been registered with **Gestalt**, an error results.

### Modifying Gestalt Selectors

You can use the **ReplaceGestalt** function to modify the function that **Gestalt** executes when passed a particular selector code. As with the function whose address is passed to **NewGestalt**, the new function must reside in the system



heap and have a calling syntax that conforms to that defined in the following section, **Specifying Gestalt Selector Functions**. Listing "Installing a selector function into the system heap" above illustrates how to replace a **Gestalt** selector function to allow the new function to call the function previously associated with the selector in question. The **ReplaceGestalt** function returns the address of the previous function.

If you attempt to redefine a selector that is not yet defined, an error is returned; in that case, the address of the previous function is undefined. Accordingly, you should always test the result code of **ReplaceGestalt** before calling **Gestalt** with the selector in question.

**Note:** If you modify the function associated with a predefined **Gestalt** selector, do not use any bits in the response parameter that are not documented. Apple reserves all undocumented bits in the response parameter returned by predefined **Gestalt** selectors.

### Specifying Gestalt Selector Functions

When you call the **NewGestalt** and **ReplaceGestalt** functions, you need to supply the address of a selector function that is called when some application passes the specified new or replacement selector to **Gestalt**. This selector function should have the following syntax and must reside in the system heap.

**OSErr mySelectorFunction** ( *OSType selector*, *long \*response* )

When you pass the new or replacement selector to **Gestalt**, **Gestalt** calls the specified selector function to determine the information that **Gestalt** should pass back to the calling software. Your function should place the result into the *long* integer pointed to by the response parameter and should return the result code that **Gestalt** will return. This function should be as simple as possible and cannot use global variables in the **A5 world** unless A5 is set up explicitly and then restored upon exit. (See the **Memory Manager** for an explanation of setting up and restoring the **A5 world**.)

Your selector function can, if necessary, call **Gestalt** and pass it other selector codes. Note that the response variable parameter is the address into which your function should place the information requested. You cannot depend on that address containing useful information when your selector function is called.