



US 20140157162A1

(19) **United States**

(12) **Patent Application Publication**
Chaudhri et al.

(10) **Pub. No.: US 2014/0157162 A1**
(43) **Pub. Date: Jun. 5, 2014**

(54) **METHOD AND APPARATUS FOR
DISPLAYING A WINDOW FOR A USER
INTERFACE**

(52) **U.S. Cl.**
CPC *G06F 3/0481* (2013.01)
USPC *715/768*

(71) Applicant: **APPLE INC.**, Cupertino, CA (US)

(72) Inventors: **Imran Chaudhri**, San Francisco, CA (US); **Bas Ording**, San Francisco, CA (US)

(73) Assignee: **APPLE INC.**, Cupertino, CA (US)

(21) Appl. No.: **14/061,660**

(22) Filed: **Oct. 23, 2013**

Related U.S. Application Data

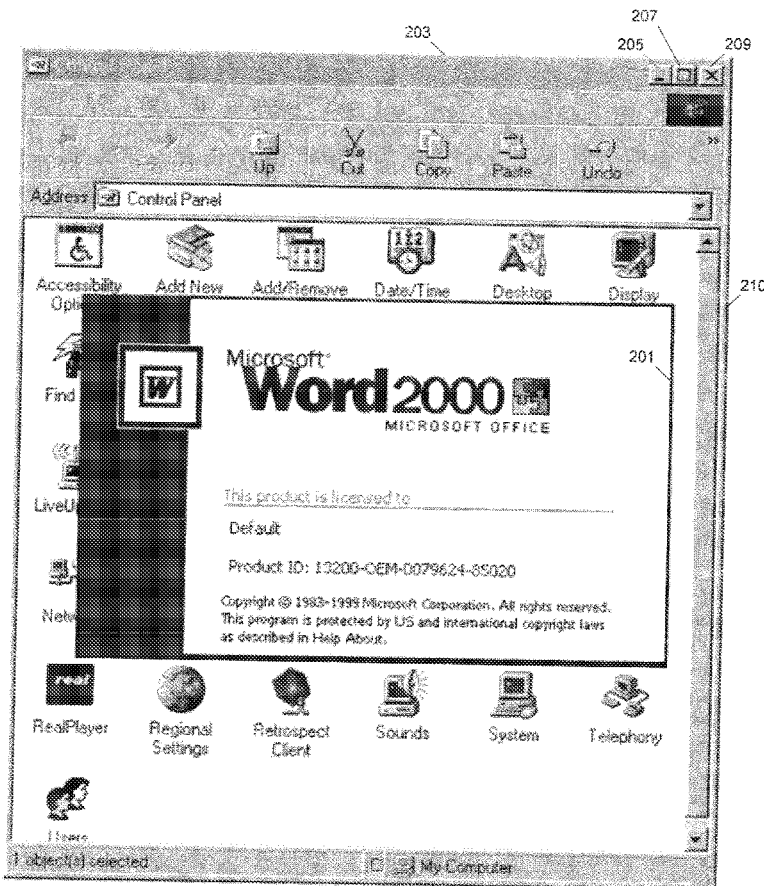
(63) Continuation of application No. 11/635,847, filed on Dec. 8, 2006, now Pat. No. 8,601,384, which is a continuation of application No. 10/193,573, filed on Jul. 10, 2002, now Pat. No. 7,343,566.

Publication Classification

(51) **Int. Cl.**
G06F 3/0481 (2006.01)

(57) **ABSTRACT**

Methods and apparatuses to display windows. In more than one embodiments of the invention, a window is closed automatically (e.g., after a timer expires, or when a condition or criterion is met, or a system input is received) without user input. In some examples, the window is translucent so that the portion of another window, when present, is visible under the window. In some examples, the image of the window is faded out before the window is closed and destroyed. In some examples, the window does not close in response to any input from a user input device. In some examples, the window is repositioned (or hidden) automatically when another translucent window is displayed. The degree of translucency, the speed for fading out, the discrete levels of translucency for fading out, the time to expire, and/or other parameters for controlling the display of the window may be set by the user or adjusted by the system (or application software programs) automatically according to system conditions or other criteria.



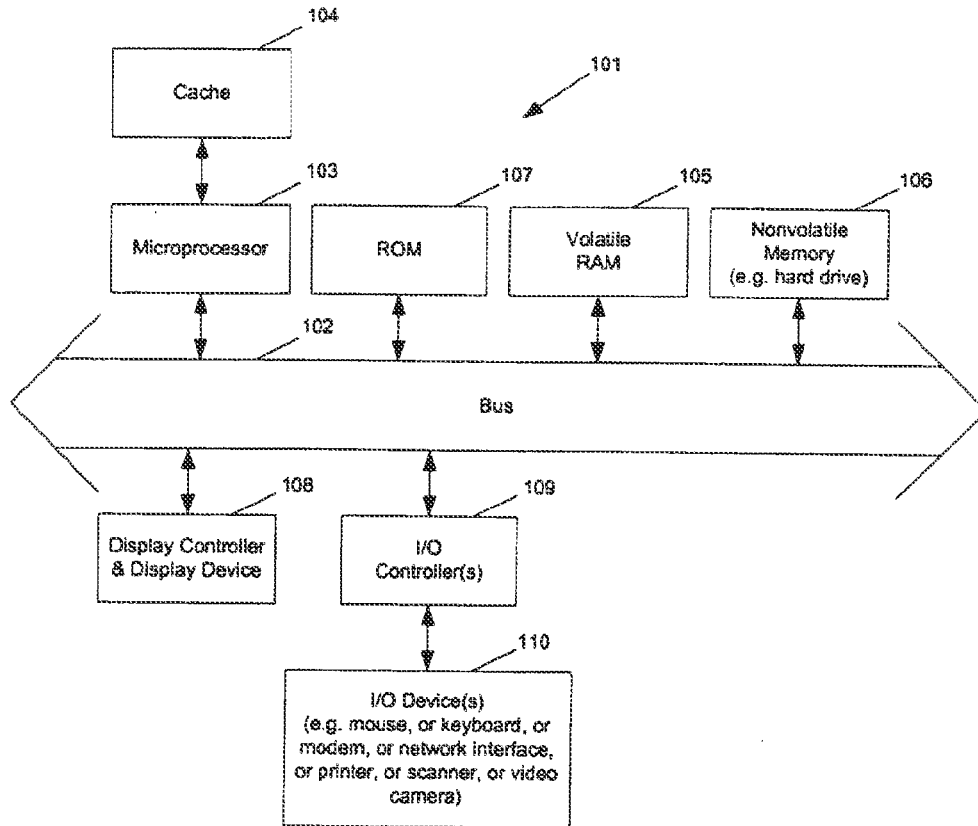


Fig. 1
(Prior Art)

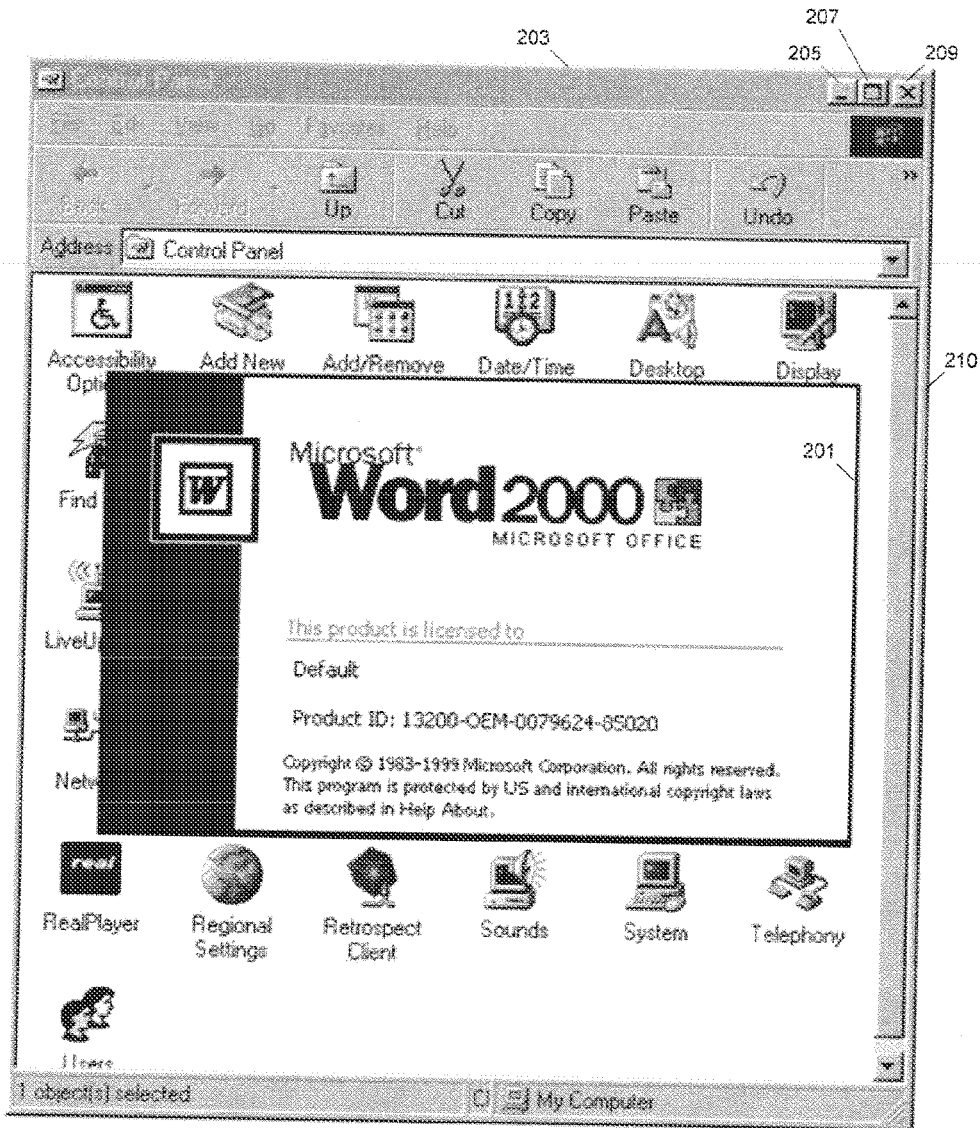


Fig. 2

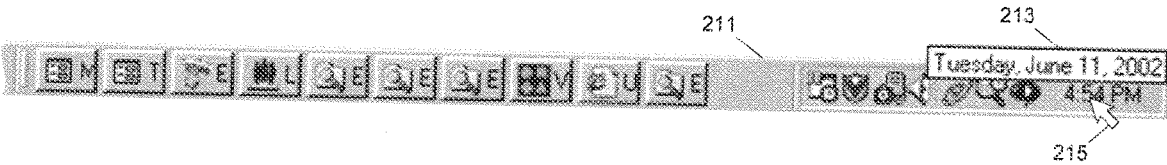


Fig. 3

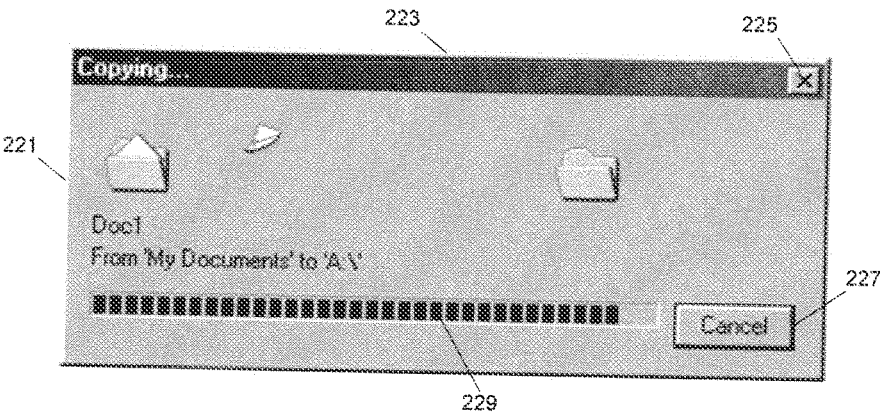


Fig. 4

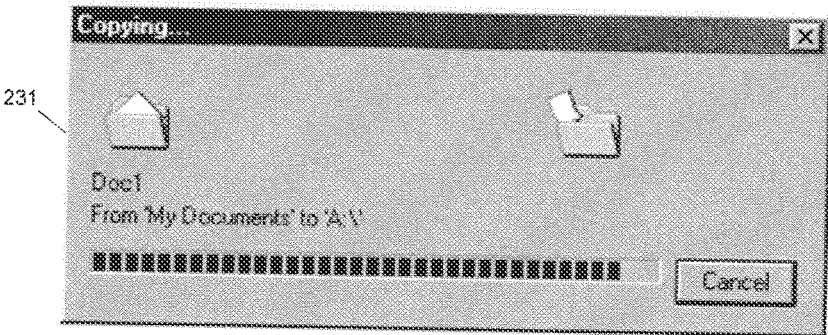


Fig. 5

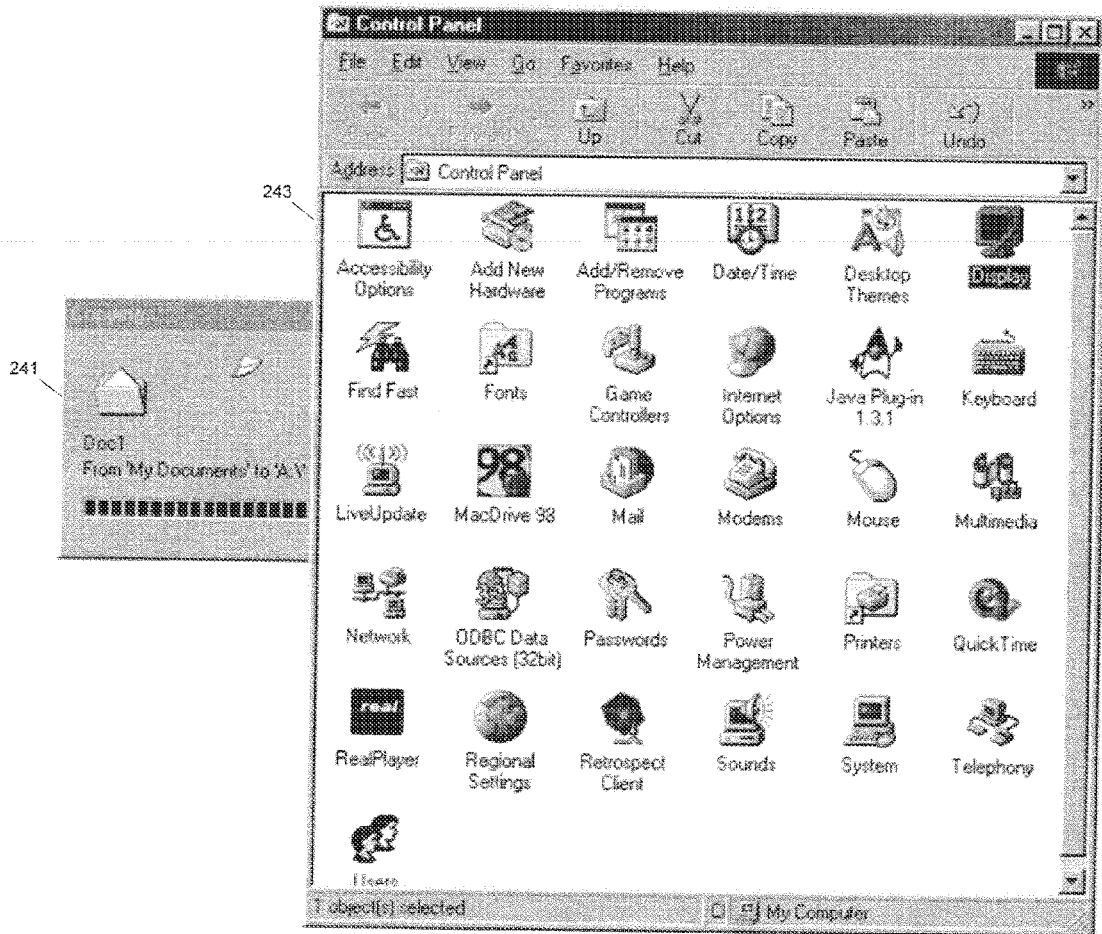


Fig. 6

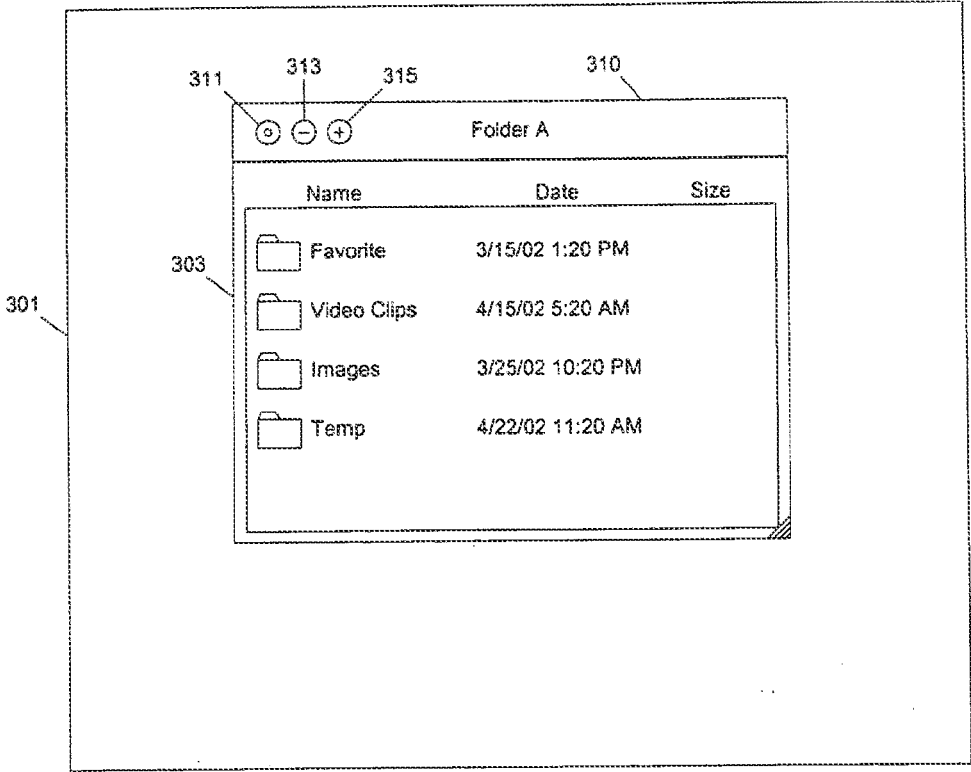


Fig. 7

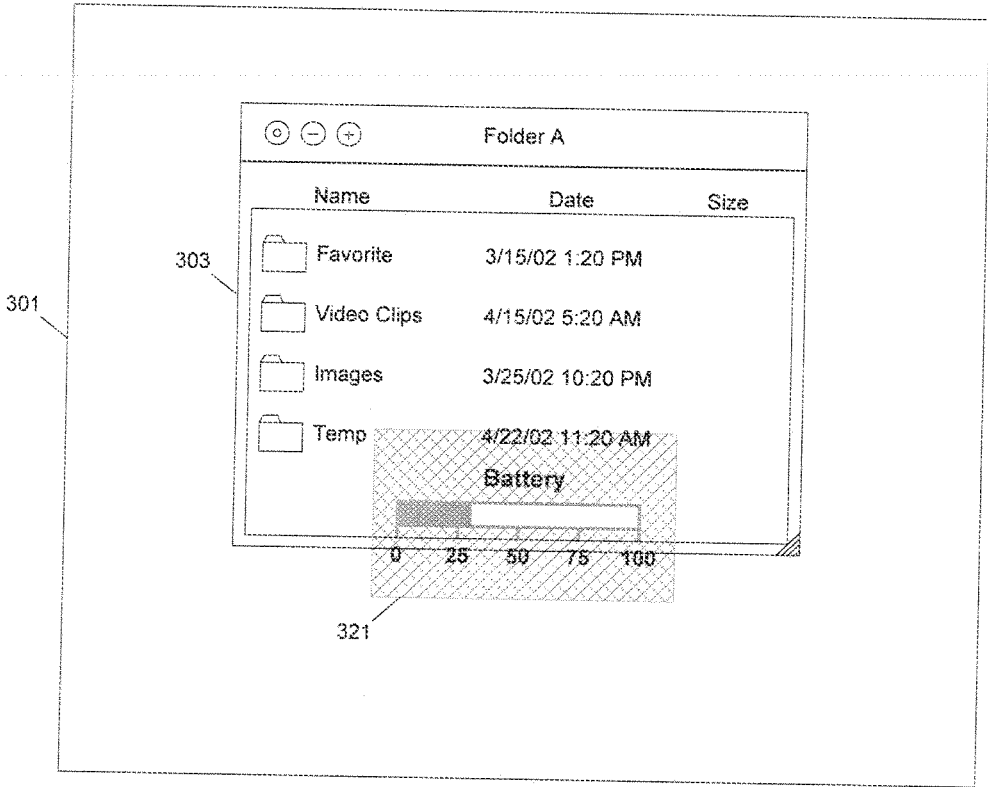


Fig. 8

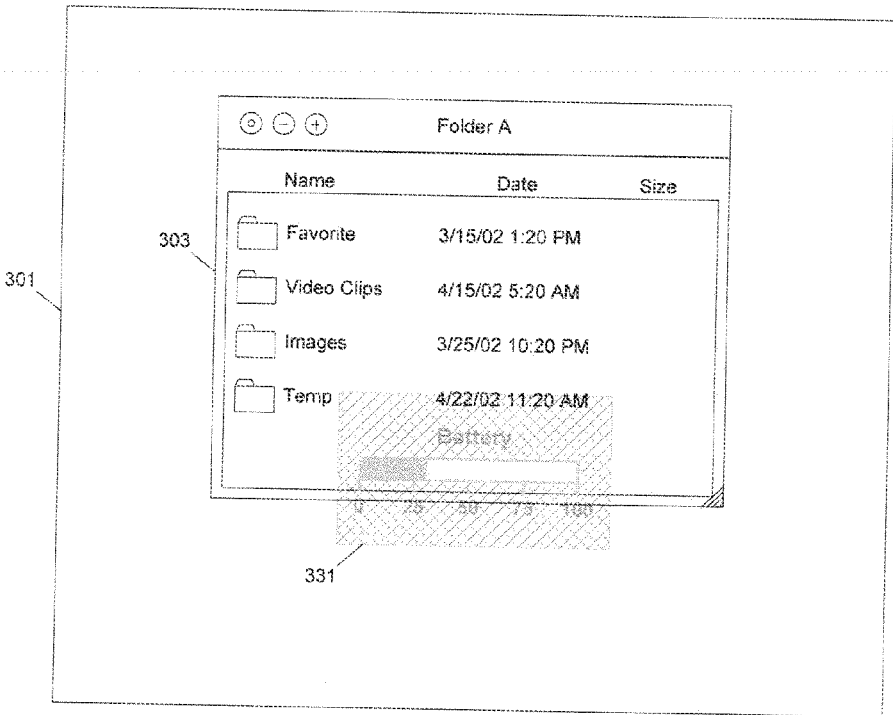


Fig. 9

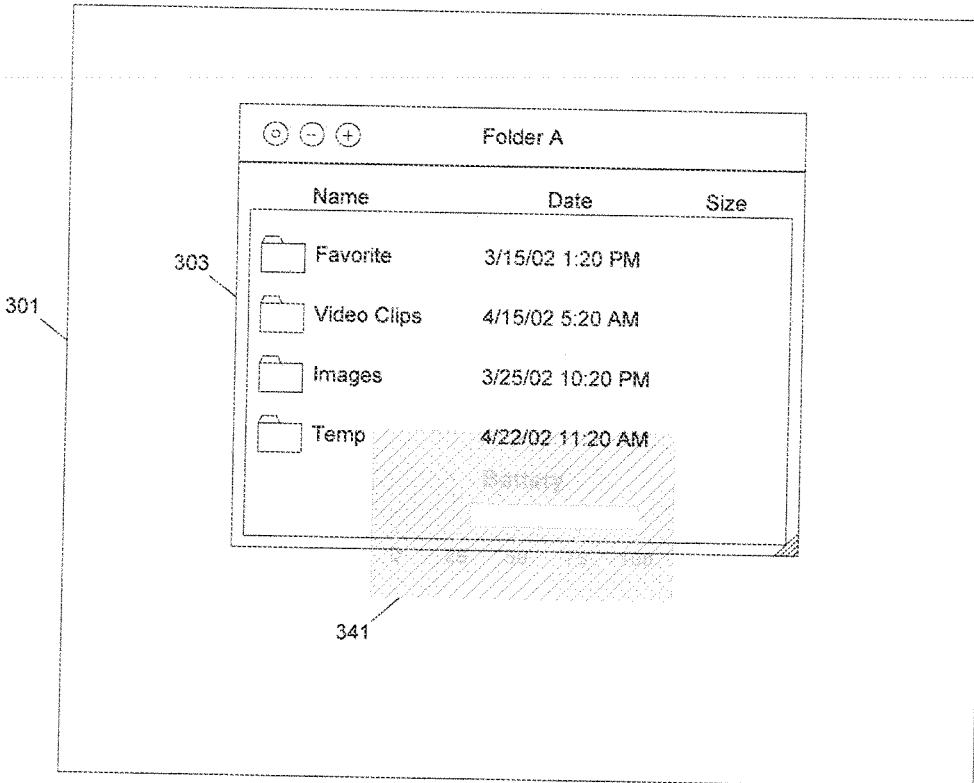


Fig. 10

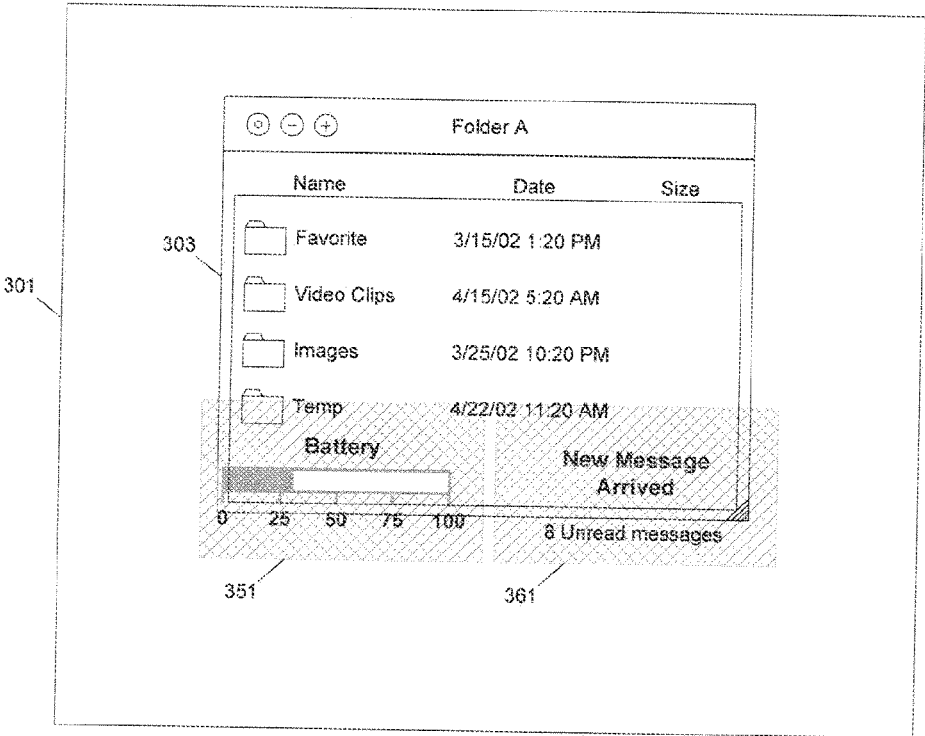


Fig. 11

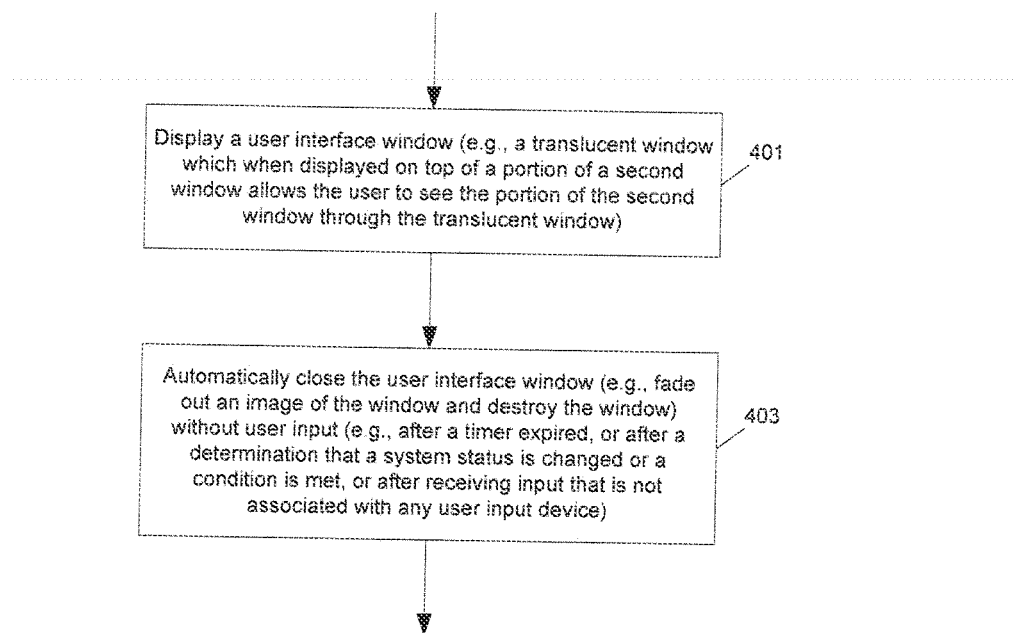


Fig. 12

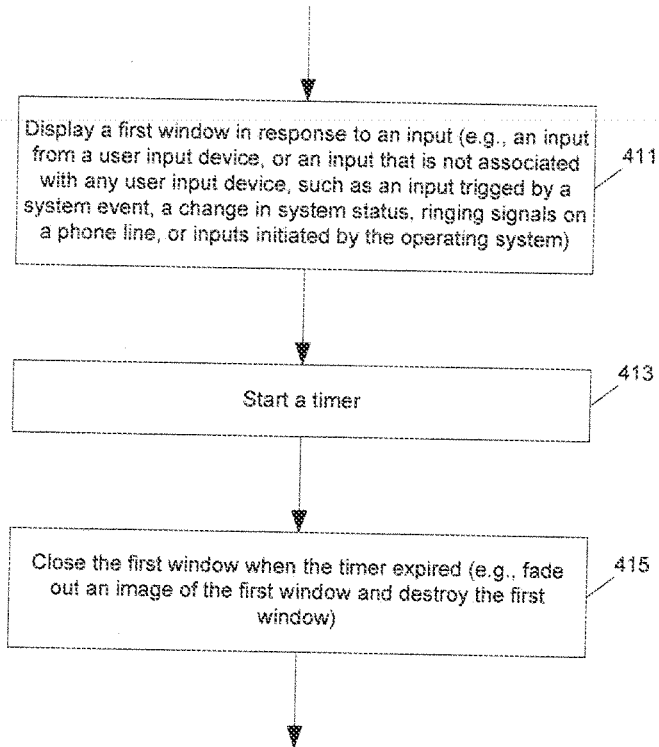


Fig. 13

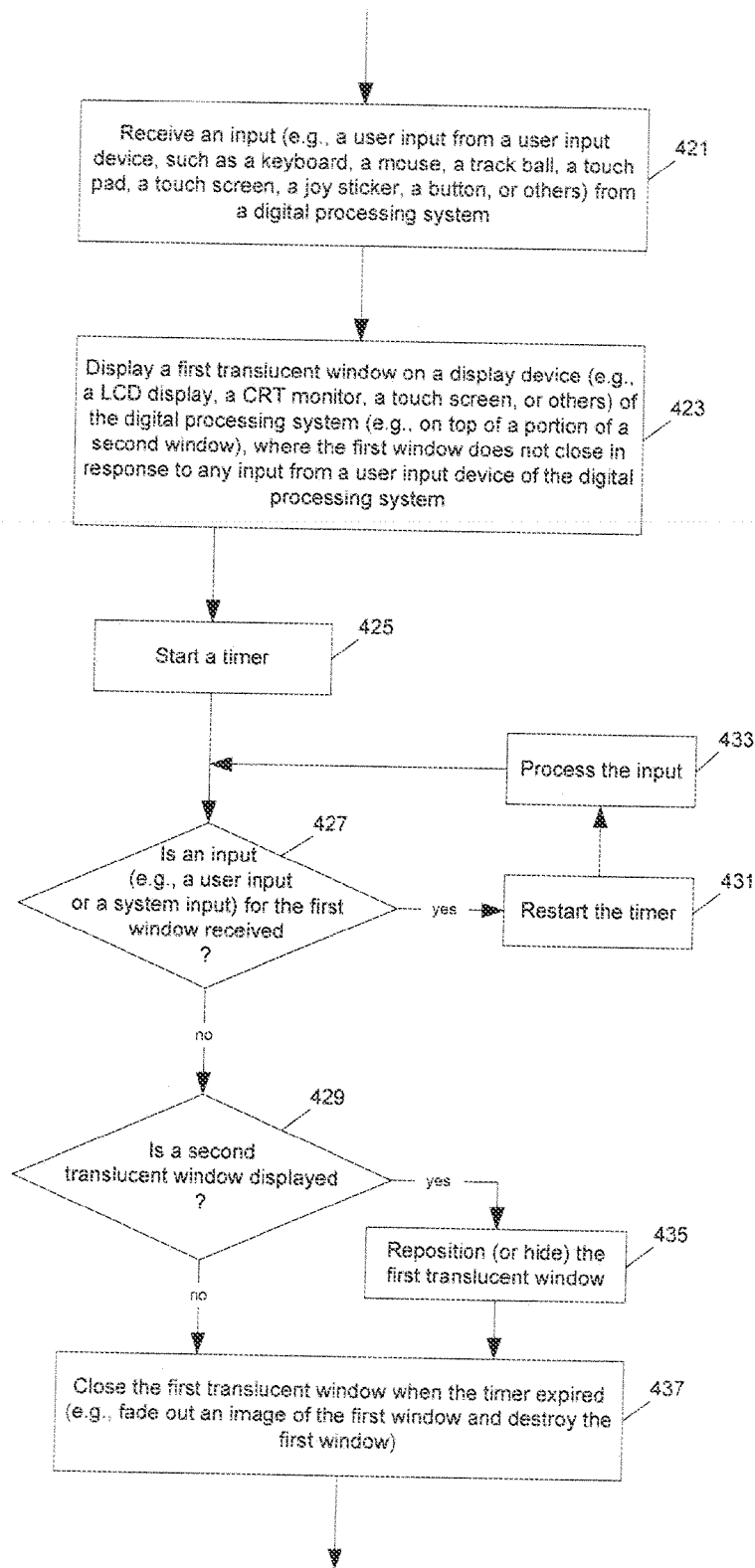


Fig. 14

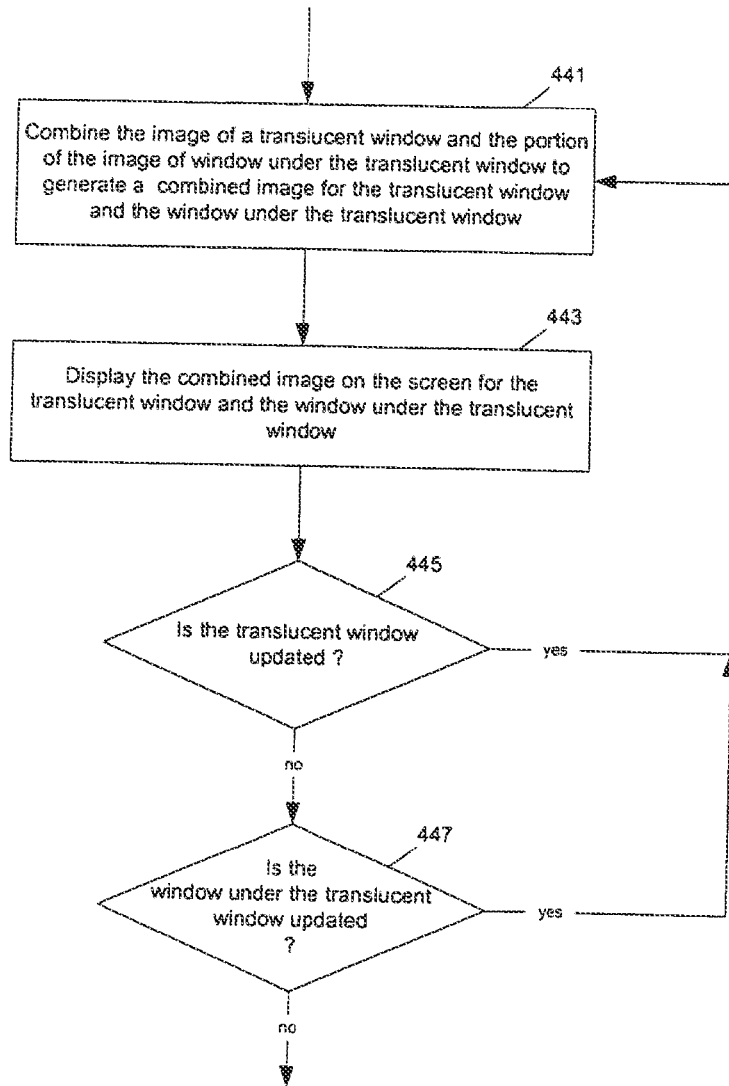


Fig. 15

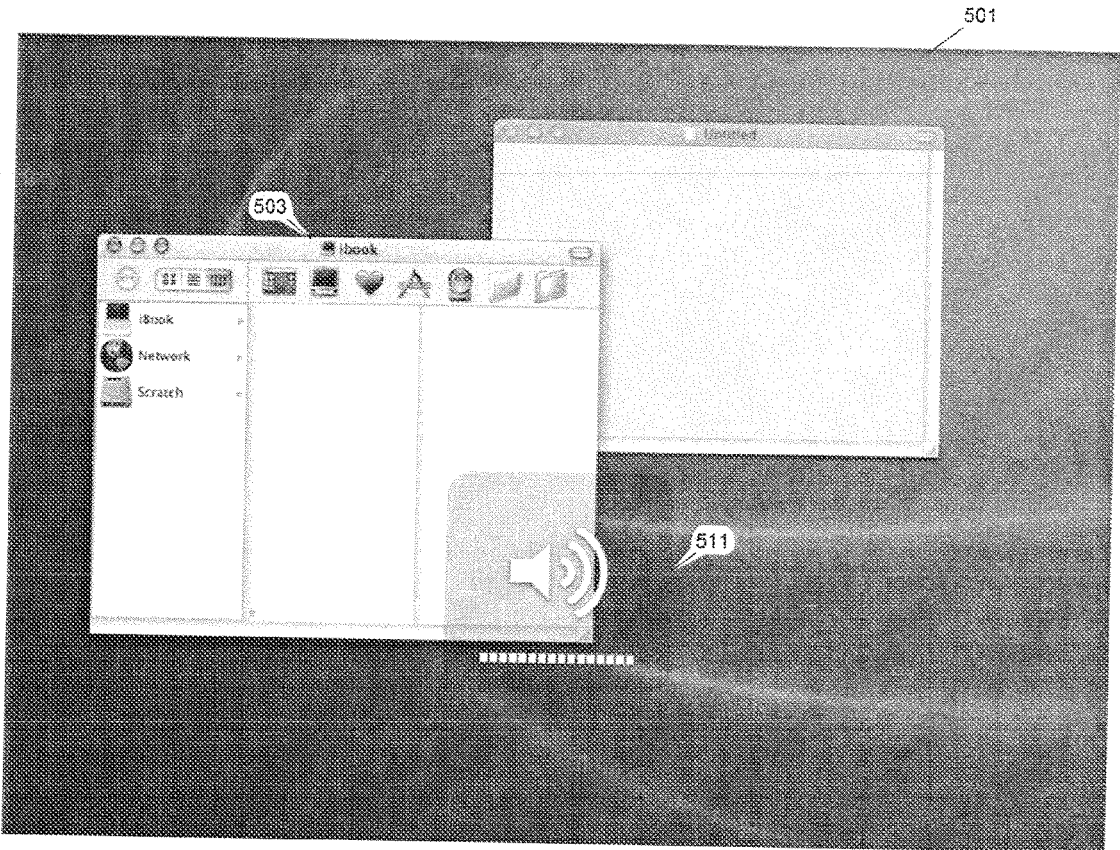


Fig. 16

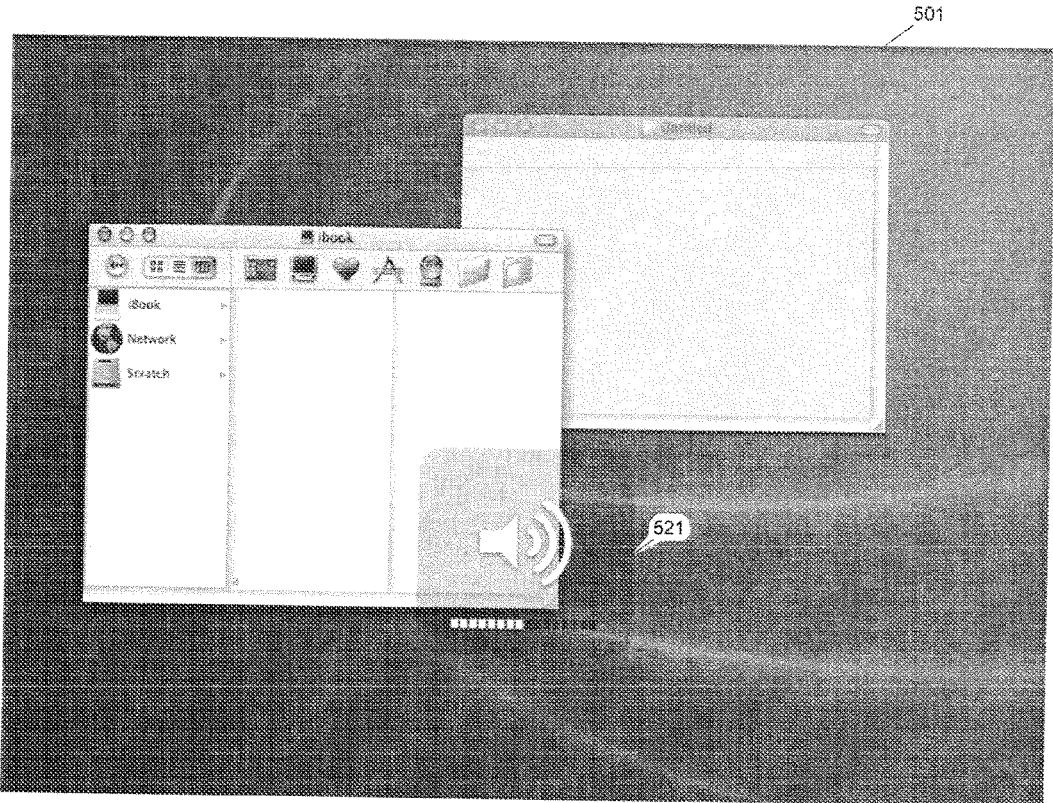


Fig. 17

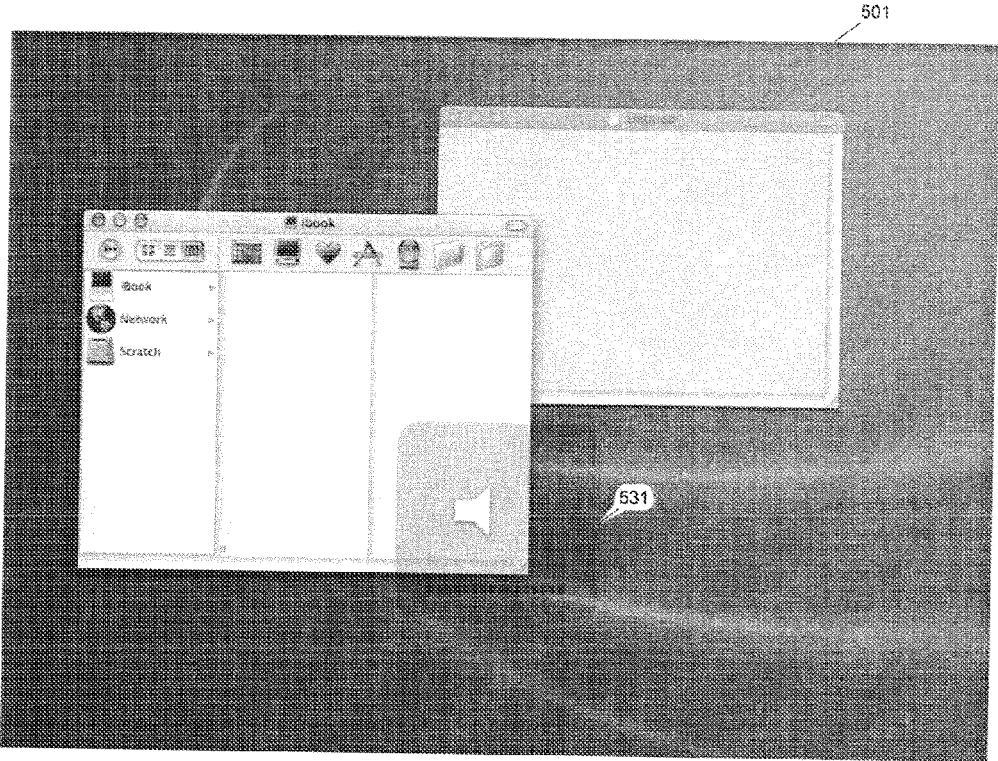


Fig. 18

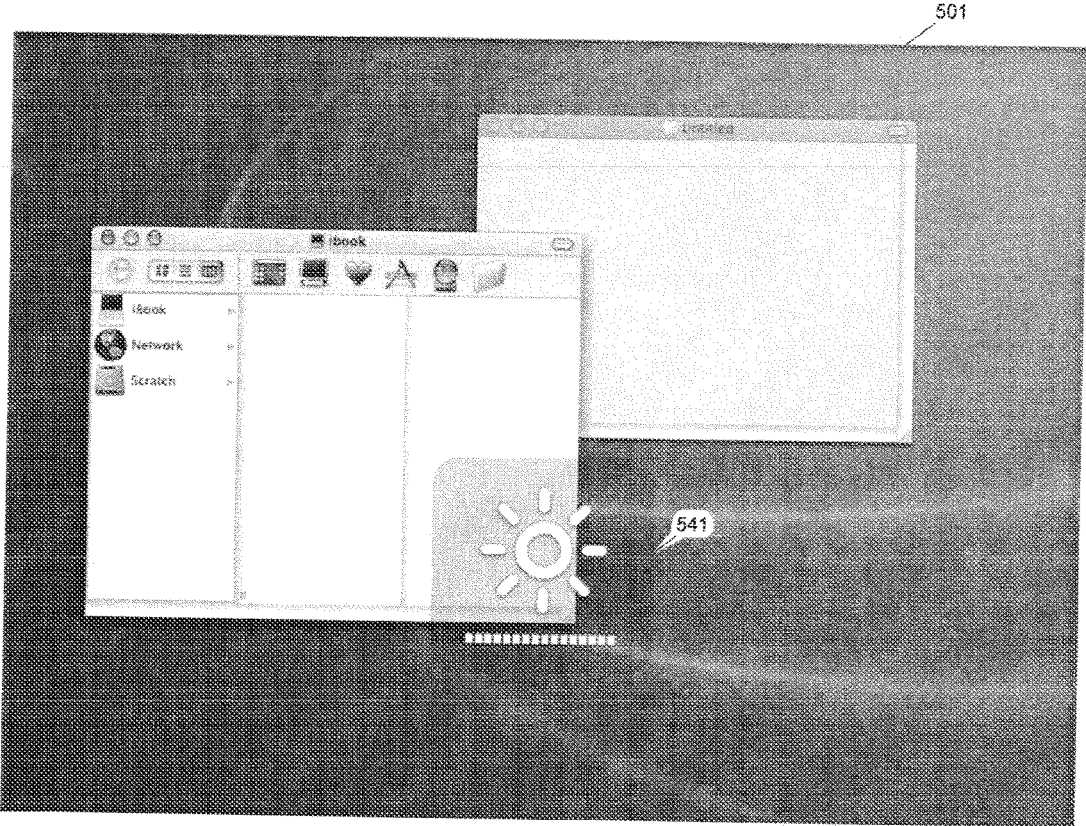


Fig. 19

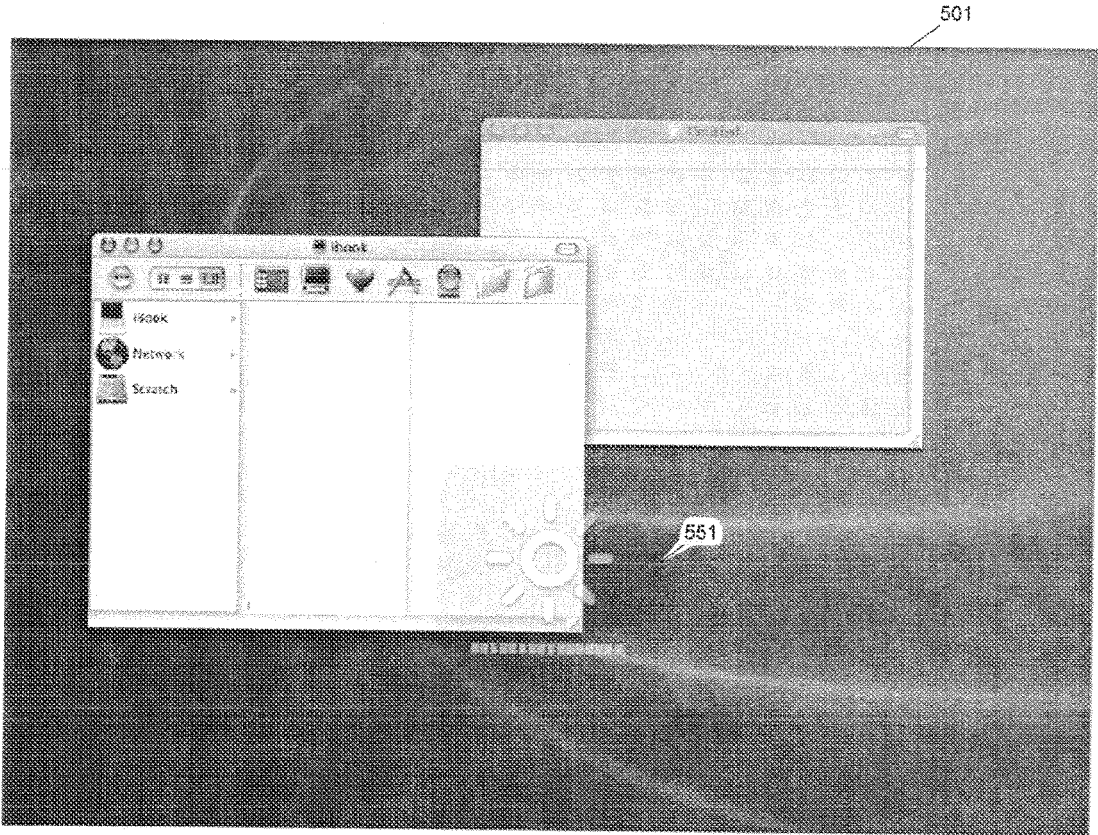


Fig. 20

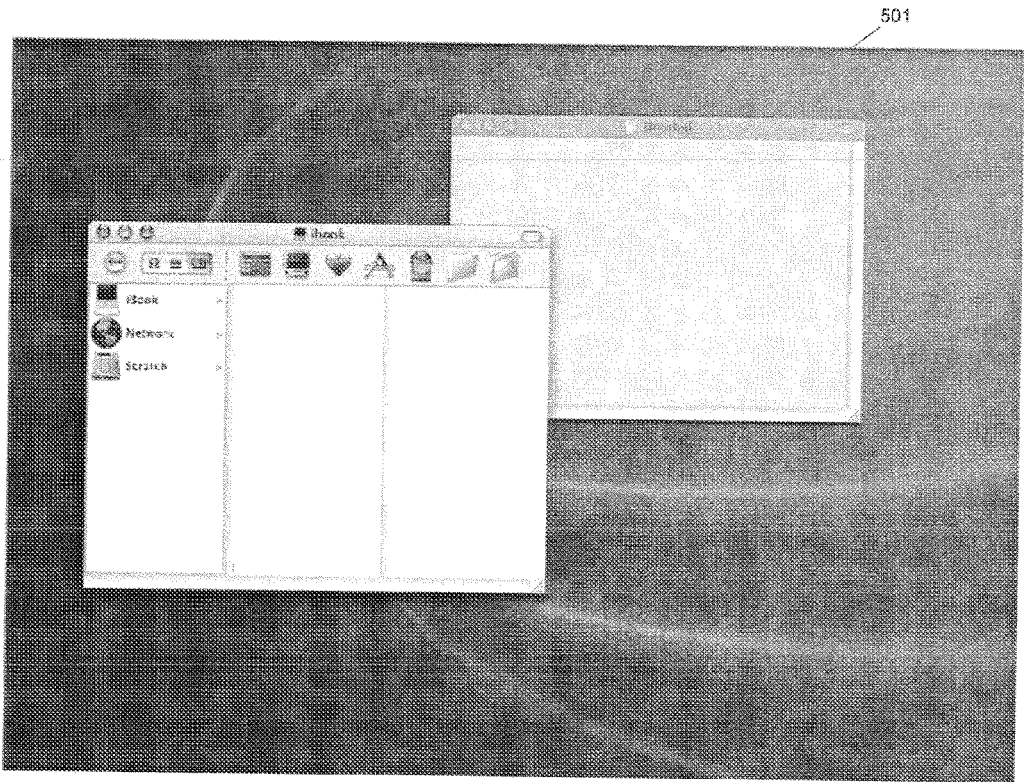


Fig. 21

METHOD AND APPARATUS FOR DISPLAYING A WINDOW FOR A USER INTERFACE

[0001] This application is a continuation of U.S. patent application Ser. No. 10/193,573, filed on Jul. 10, 2002.

FIELD OF THE INVENTION

[0002] The invention relates to graphical user interfaces, and more particularly to such interfaces with windows.

[0003] A portion of the disclosure of this patent document contains material which is subject to copyright protection. The copyright owner has no objection to the facsimile reproduction by anyone of the patent document or the patent disclosure, as it appears in the Patent and Trademark Office patent file or records, but otherwise reserves all copyright rights whatsoever. Copyright Apple Computer, Inc., 2002

BACKGROUND OF THE INVENTION

[0004] Many digital processing systems use window-based graphical user interfaces for users to interact with software programs. Images of windows are displayed on a display device to show the user the states of the software programs; and user input devices (e.g., a keyboard and a mouse) are used to accept user inputs. In addition to user input devices, a digital processing system may have other devices (e.g., sensors) for accepting system inputs, such as phone line status, power supply status, storage disk usage status, communication connection status, execution status of software programs, and others that are not directly related to user inputs (signals associated with user input devices).

[0005] FIGS. 2-6 illustrate examples of traditional windows. FIG. 2 shows a typical window 210 that has title bar 203 and buttons 205, 207 and 209 for minimizing, maximizing, and closing the window. The title bar and the buttons on the title bar can be used to manipulating the position and size of the window. For example, title bar 203 may be clicked (e.g., pressing a button of a mouse while the cursor is on the title bar) to bring the window to the top level of the window displaying hierarchy so that if there are any other windows displayed at the same location of window 210, these windows will be hidden under window 210.

[0006] When a user starts an application program, a window (e.g., window 201) may be displayed to show the license/copyright information while the components of the software program are being loaded. After the components of the software program are fully loaded, the license/copyright window is closed automatically so that the user can start to use the software program without interference from the license/copyright window.

[0007] FIG. 3 shows task bar 211 with flash help window 213. When the user pauses cursor 215 at a location of the task bar for a short period of time, flash help window 213 appears. If the user does not move the cursor for another short period of time while window 213 is displayed, flash window 213 disappears. If the user moves cursor 215 slightly (e.g., using a mouse, a track ball, or a touch pad) and pauses the cursor 215 again, flash help window may appear again.

[0008] FIGS. 4-6 show a window that displays the progress of copying a file. When a file is copied from one location to another location, window 221 is displayed to indicate the progress. Button 227 is provided for canceling the copy operation; and button 225 is for closing the progress window. The progress of the operation is indicated by progress bar 229

and an animation showing that the document is going from one folder to another. Windows 221 and 231 in FIGS. 4 and 5 show two snap shots of the animation. A user may drag title bar 223 (e.g., pressing down and holding a button of a mouse and moving the mouse while holding down the button) to drag the window from one location on a screen to another; and the user can click on the title bar to bring the window to the top level when the window is partially covered by another window (e.g., when window 241 is partially covered by window 243, as shown in FIG. 6). When the copy operation completes, the progress window closes automatically.

[0009] Traditional windows typically provide strong user interactions, which may cause distractions. For example, a user waits for window 201 to disappear to view window 210 in FIG. 2; the user manipulates a cursor control device (e.g., a mouse, a track ball, or a touch pad) to view or dismiss flash help window 213 in FIG. 3; and, the user interaction is provided to relocate the progress window or change the window displaying hierarchy to see the progress of window 241 in FIG. 6.

SUMMARY OF THE INVENTION

[0010] Methods and apparatuses to display windows are described here. There are many different embodiments which are described here. Some of these embodiments are summarized in this section.

[0011] In more than one embodiment of the invention, a window is closed automatically (e.g., after a timer expires, or when a condition or criterion is met, or system input is received) without user input. In some examples, the window is translucent so that the portion of another window, when present, is visible under the window. In some examples, the image of the window is faded out before the window is closed and destroyed. In a further example, the level of translucency, the speed for fading out, the discrete levels of translucency for fading out, the time to expire, and/or other parameters for controlling the display of the window may be set by the user or adjusted by the system (or application software programs) automatically according to system conditions or other criteria.

[0012] In one embodiment of the invention, a method to display a user interface window for a digital processing system includes: displaying a first window in response to receiving a first input from a user input device (e.g., a keyboard, mouse, track ball, touch pad, touch screen, joy stick, button, or others) of the digital processing system which is capable of displaying at least a portion of a second window under the first window; starting a timer; and closing the first window in response to a determination that the timer expired. The first window does not close in response to any input from a user input device of the digital processing system. In one example according to this embodiment, the first window is translucent; the portion of the second window is visible while under the first window; and the first window is at a top level in a window displaying hierarchy. In one example, an image of the first window is faded out on the screen before the first window is destroyed to close the first window. In one example, the second window, if displayed, closes in response to an input from a user input device of the digital processing system; and the first window does not respond to any input from a user input device of the digital processing system. In one example, the first window is repositioned in response to a third window (e.g., an alert window or a translucent window) being displayed; in another example, the first window is hidden in

response to a third window being displayed at a location where the first window is displayed. In one example, the first window is repositioned on a display in response to a second input for the first window (e.g., an input indicating that a third window is displayed, or an input from a user input device of the digital processing system to reposition the window, such as dragging and dropping the window); and a position of the first window in a window displaying hierarchy can be adjusted in response to a third input (e.g., bringing another window in front of the first window). In one example, the first window is displayed at a position on a display of the digital processing system that is independent of a position of a cursor on the display (e.g., a position centered horizontally on the display); and the timer is restarted in response to receiving a second input for the first window (e.g., from a user input device of the digital processing system).

[0013] In another embodiment of the invention, a method to display a user interface window for a digital processing system includes: displaying a first translucent window such that if a portion of a second window is displayed on the digital processing system under the first window, the portion of the second window is visible under the first window; and closing the first window without user input. In one example according to this embodiment, a timer is started so that when the timer expires the first window is closed (e.g., fading out an image of the first window and destroy the first window). In another example, the first window is closed in response to an input that is not associated with a user input device of the digital processing system. In a further example, the first window is closed in response to a determination that a system condition is met (e.g., a system status is changed, or other criteria).

[0014] In a further embodiment of the invention, a method to display a user interface window for a digital processing system includes: displaying a first window in response to receiving a first input that is not associated with a user input device of the digital processing system; starting a timer; and closing the first window in response to a determination that the timer expired (e.g., fading out an image of the first window; and destroying the first window). In one example, the first window does not close in response to any input from a user input device of the digital processing system (e.g., the first window does not respond to any input from a user input device of the digital processing system); and the first window is translucent such that a portion of a second window is visible when displayed under the first window. In one example, the first window is repositioned on a display without user input (e.g., in response to a third window being displayed). In another example, the timer is restarted in response to receiving a second input for the first window; and the second input is received from a user input device of the digital processing system.

[0015] The present invention includes apparatuses which perform these methods, including data processing systems which perform these methods and computer readable media which when executed on data processing systems cause the systems to perform these methods.

[0016] Other features of the present invention will be apparent from the accompanying drawings and from the detailed description which follow.

BRIEF DESCRIPTION OF THE DRAWINGS

[0017] The present invention is illustrated by way of example and not limitation in the figures of the accompanying drawings in which like references indicate similar elements.

[0018] FIG. 1 shows a block diagram example of a data processing system which may be used with the present invention.

[0019] FIGS. 2-6 illustrate examples of traditional windows.

[0020] FIGS. 7-11 illustrate example scenarios of displaying a window according to one embodiment of the present invention.

[0021] FIG. 12 shows a flow diagram of a method to display a window according to one embodiment of the present invention.

[0022] FIG. 13 shows a flow diagram of a method to close a window according to one embodiment of the present invention.

[0023] FIG. 14 shows a detailed flow diagram of a method to control a translucent window according to one embodiment of the present invention.

[0024] FIG. 15 shows a method to display a translucent window according to one embodiment of the present invention.

[0025] FIGS. 16-21 show example screen images of windows displayed according to one embodiment of the present invention.

DETAILED DESCRIPTION

[0026] The following description and drawings are illustrative of the invention and are not to be construed as limiting the invention. Numerous specific details are described to provide a thorough understanding of the present invention. However, in certain instances, well known or conventional details are not described in order to avoid obscuring the description of the present invention.

[0027] Many of the methods of the present invention may be performed with a digital processing system, such as a conventional, general purpose computer system. Special purpose computers which are designed or programmed to perform only one function may also be used.

[0028] FIG. 1 shows one example of a typical computer system which may be used with the present invention. Note that while FIG. 1 illustrates various components of a computer system, it is not intended to represent any particular architecture or manner of interconnecting the components as such details are not germane to the present invention. It will also be appreciated that network computers and other data processing systems which have fewer components or perhaps more components may also be used with the present invention. The computer system of FIG. 1 may, for example, be an Apple Macintosh computer.

[0029] As shown in FIG. 1, the computer system 101, which is a form of a data processing system, includes a bus 102 which is coupled to a microprocessor 103 and a ROM 107 and volatile RAM 105 and a non-volatile memory 106. The microprocessor 103, which may be, for example, a G3 or G4 microprocessor from Motorola, Inc. or IBM is coupled to cache memory 104 as shown in the example of FIG. 1. The bus 102 interconnects these various components together and also interconnects these components 103, 107, 105, and 106 to a display controller and display device 108 and to peripheral devices such as input/output (110) devices which may be mice, keyboards, modems, network interfaces, printers, scanners, video cameras and other devices which are well known in the art. Typically, the input/output devices 110 are coupled to the system through input/output controllers 109. The volatile RAM 105 is typically implemented as dynamic RAM

(DRAM) which requires power continually in order to refresh or maintain the data in the memory. The non-volatile memory **106** is typically a magnetic hard drive or a magnetic optical drive or an optical drive or a DVD RAM or other type of memory systems which maintain data even after power is removed from the system. Typically, the non-volatile memory will also be a random access memory although this is not required. While FIG. 1 shows that the non-volatile memory is a local device coupled directly to the rest of the components in the data processing system, it will be appreciated that the present invention may utilize a non-volatile memory which is remote from the system, such as a network storage device which is coupled to the data processing system through a network interface such as a modem or Ethernet interface. The bus **102** may include one or more buses connected to each other through various bridges, controllers and/or adapters as is well known in the art. In one embodiment the I/O controller **109** includes a USB (Universal Serial Bus) adapter for controlling USB peripherals, and/or an IEEE-1394 bus adapter for controlling IEEE-1394 peripherals.

[0030] It will be apparent from this description that aspects of the present invention may be embodied, at least in part, in software. That is, the techniques may be carried out in a computer system or other data processing system in response to its processor, such as a microprocessor, executing sequences of instructions contained in a memory, such as ROM **107**, volatile RAM **105**, non-volatile memory **106**, cache **104** or a remote storage device. In various embodiments, hardwired circuitry may be used in combination with software instructions to implement the present invention. Thus, the techniques are not limited to any specific combination of hardware circuitry and software nor to any particular source for the instructions executed by the data processing system. In addition, throughout this description, various functions and operations are described as being performed by or caused by software code to simplify description. However, those skilled in the art will recognize what is meant by such expressions is that the functions result from execution of the code by a processor, such as the microprocessor **103**.

[0031] A machine readable media can be used to store software and data which when executed by a data processing system causes the system to perform various methods of the present invention. This executable software and data may be stored in various places including for example ROM **107**, volatile RAM **105**, non-volatile memory **106** and/or cache **104** as shown in FIG. 1. Portions of this software and/or data may be stored in any one of these storage devices.

[0032] Thus, a machine readable media includes any mechanism that provides (i.e., stores and/or transmits) information in a form accessible by a machine (e.g., a computer, network device, personal digital assistant, manufacturing tool, any device with a set of one or more processors, etc.). For example, a machine readable media includes recordable/non-recordable media (e.g., read only memory (ROM); random access memory (RAM); magnetic disk storage media; optical storage media; flash memory devices; etc.), as well as electrical, optical, acoustical or other forms of propagated signals (e.g., carrier waves, infrared signals, digital signals, etc.); etc.

[0033] At least one embodiment of the present invention seeks to display a window with reduced distractions so that a user can focus on more important windows.

[0034] FIGS. 7-11 illustrate example scenarios of displaying a window according to one embodiment of the present invention. Traditional window **303** is shown in FIG. 7. Win-

dow **303** contains control buttons **311**, **313** and **315** for closing, minimizing and maximizing the window. Window **303** also has title bar **310**, which may be used to relocate the window on screen **301**. Consider a scenario where the battery power of the system is lower than a threshold. After the system detects such a system status change, window **321** may be displayed near the center of screen **301**, as shown in FIG. 8. Window **321** is translucent so that regular window **303** is still visible under window **321**. Once window **321** is displayed on the screen, a timer is started to control the closing of the window. When the timer expires, window **321** is automatically closed without any user input. Thus, window **321** displays the message of low battery power to the user without forcing the user to provide inputs to dismiss the message window. Since window **321** is translucent and transient, the portion of window **303** that is under window **321** is still visible. Thus, the user can continue working with window **303** (or other window) without having to provide additional input to get message window **321** out of the way.

[0035] In one embodiment of the present invention, translucent window **321** is always displayed at the top level of the window displaying hierarchy so that the translucent window is always visible when displayed. This eliminates the need for the user to change the window displaying hierarchy to bring up the translucent window when another traditional window is brought up to the top of the window displaying hierarchy (e.g., by creating a new window or accidentally changing the hierarchy). In another embodiment of the present invention, the user can change the position of the translucent window in the hierarchy so that if the user desires the translucent window may be sent to a background position.

[0036] In one embodiment of the present invention, the image of window **321** is faded out when the timer expires, which is illustrated by the images of windows **321**, **331**, and **341** in FIGS. 8, 9 and 10. After the image of window **321** is faded out, window **321** is destroyed.

[0037] In another embodiment of the present invention, a translucent message window starts to fade out when a status change is detected. For example, a message window is displayed when the system detects the ringing signal on a phone line. When the system detects that the ringing signal is no longer present on the phone line, the image of the message window is faded out; and the message window is destroyed. Similarly, a translucent progress window for showing the progress of copying a file can be faded out and destroyed after the copy process ends. In one example, message window **361** as shown in FIG. 11 is displayed when a new message arrives. When the user starts to open an application to view the new message, message window **361** is closed automatically so that the user does not have to provide input to dismiss the message window or wait for the message window to fade out.

[0038] In one embodiment of the present invention, the image of window **321** gradually sets in when the window is first displayed. In another embodiment of the present invention, window **321** in FIG. 8 is automatically relocated or moved (e.g., in an animation fashion) to a different location so that the image of window **321** does not obscure the display of any particular portion of windows **303** for an extended period of time. For example, window **321** may be automatically moved across the screen horizontally from the left hand side of screen **301** to the right hand side of screen **301** (or near the center of screen **321** in a circular motion).

[0039] In one embodiment of the present invention, the system detects (or manages) all the translucent windows so

that when a second translucent window is displayed before the first translucent window is closed, the first translucent window is repositioned so that the second translucent window can be easily seen on the screen without interference with each other. For example, after battery low window 321 is displayed as in FIG. 8, the system may detect a new message arrived for the user. Thus, translucent window 361 is displayed as in FIG. 11 to inform the user about the new message. At the same time, window 351 is automatically moved to a position as seen in FIG. 11 so that both translucent windows 351 and 361 can be easily seen on the screen. Alternatively, the first translucent window (e.g., window 351) is hidden so that only the second window (e.g., window 361) is displayed. The timer of the first window is stopped while being hidden until the second window is automatically closed. In a further embodiment of the present invention, the window system tracks the translucent windows to automatically schedule the sequence and the screen positions of the display of the translucent windows according to the importance of the windows, the time to close (or estimated time to close), and/or other conditions.

[0040] FIGS. 8-11 show an example of translucent windows that are initiated by a system without any input associated with user input devices (e.g., a keyboard, mouse, track ball, touch pad, touch screen, joy stick, button, or other criteria). In one embodiment of the present invention, these translucent windows do not consume any user input; and no user input can be provided to these windows to close these windows, which close automatically when certain conditions are met (e.g., the expiration of a timer, the change in system status, and others). In one alternative embodiment of the present invention, these windows accept predetermined inputs (e.g., special function keys, such as the escape key "ESC" for closing) so that a user has the option to directly control the display of these translucent windows.

[0041] A user may initiate a translucent window through an input associated with a user input device. For example, a user may use a special function key to adjust volume (or contrast, or brightness). In response to the special function key, a translucent window is displayed to show the current volume level (or contrast, or brightness). In one embodiment of the present invention, the window system (or an application program) automatically determines a location for displaying the translucent volume window (e.g., independent from the location of a cursor on the screen). When the volume window receives an input from the function key for adjust volume, the timer for the translucent volume window is restarted. After the user stops adjusting the volume for a predetermined amount of time, the timer expires; and the volume control window is faded out and closed automatically. In one embodiment of the present invention, the volume window is not translucent. In one embodiment of the present invention, the translucent window initiated by an input associated with a user input device does not close in response to any input from a user input device (e.g., the window does not have a button for closing the window); the window closes only automatically. When the window does not close in response to any input from a user input device, the window may still respond to system inputs, such as a request from the operating system to close the window (e.g., when the user starts to shut down a computer system). In one embodiment of the present invention, a message window initiated by a user only displays a message to the user without consuming any input from user input devices.

[0042] In one embodiment of the present invention, when a translucent window accepts user input, the translucent window consumes only predetermined inputs for user input devices; other inputs are forwarded to normal windows as if the translucent window does not exist. For example, if a cursor related event (e.g., a click) is not accepted by the translucent window, the input is considered for the window that is just under the translucent window so that the user can interact with the window under the translucent window as if the translucent window does not exist. If the translucent window does not consume a keyboard input, the keyboard input is forwarded to the window that has the keyboard focus (which is typically indicated by a highlighted title bar). Thus, the presence of the translucent window has minimum distractions for the user working on regular windows.

[0043] FIGS. 7-11 illustrate one embodiment of the present invention with translucent windows. It is apparent to one skilled in the art from this description that some methods of the present invention can be implemented for windows that are not translucent.

[0044] FIG. 12 shows a flow diagram of a method to display a window according to one embodiment of the present invention. Operation 401 displays a user interface window (e.g., a translucent window which when displayed on top of a portion of a second window allows the user to see the portion of the second window through the translucent window); and operation 403 automatically closes the user interface window (e.g., fade out an image of the window and destroy the window) without user input (e.g., after a timer expired, or after a determination that a system status is changed or a condition is met, or after receiving input that is not associated with any user input device).

[0045] FIG. 13 shows a flow diagram of a method to close a window according to one embodiment of the present invention. Operation 411 displays a first window in response to an input (e.g., an input from a user input device, or an input that is not associated with any user input device, such as an input triggered by a system event, a change in system status, ringing signals on a phone line, or inputs initiated by the operating system). Operation 413 starts a timer. Operation 415 closes the first window when the timer expires (e.g., fade out an image of the first window and destroy the first window).

[0046] FIG. 14 shows a detailed flow diagram of a method to control a translucent window according to one embodiment of the present invention. After operation 421 receives an input (e.g., a user input from a user input device, such as a keyboard, a mouse, a track ball, a touch pad, a touch screen, a joy sticker, a button, or others) from a digital processing system, operation 423 displays a first translucent window on a display device (e.g., a LCD display, a CRT monitor, a touch screen, or others) of the digital processing system (e.g., on top of a portion of a second window), where the first window does not close in response to any input from a user input device of the digital processing system. Operation 425 starts a timer. When operation 427 determines that an input (e.g., a user input or a system input) for the first window is received, operation 431 restarts the timer; and operation 433 processes the input (alternatively, the timer may be stopped and restarted after the input is processed). When operation 429 determines that a second translucent window is displayed, operation 435 repositions (or hides) the first translucent window. When one of a number of translucent windows is closed, the remaining translucent window(s) may be repositioned (or displayed if hidden). Operation 437 closes the first translucent window

when the timer expires (e.g., by fading out an image of the first window and destroying the first window).

[0047] FIG. 15 shows a method to display a translucent window according to one embodiment of the present invention. Operation 441 combines the image of a translucent window and the portion of the image of window under the translucent window to generate a combined image for the translucent window and the window under the translucent window. Operation 443 displays the combined image on the screen for the translucent window and the window under the translucent window. If operation 445 determines that the translucent window is updated or operation 447 determines that the window under the translucent window is updated, operation 441 is performed to update the corresponding portion of the screen image. In a buffered window system, the images of the translucent window and the window under the translucent window are generated separately; and the window system combines the images of the windows to display the translucent window and the window under it. In a non-buffered window system, the translucent window may generate the image of the translucent window on top of the other window using the image of the window under it. For example, the translucent window obtains the image of the window under it after the window under it draws on the frame buffer; then, the translucent window generates a combined image to update the corresponding portion of the screen.

[0048] FIGS. 16-21 show example screen images of windows displayed according to one embodiment of the present invention. When a user starts to adjust the volume level (e.g., pressing on a function key for increasing or decreasing volume, or selecting an item from a system control menu with a cursor control device, such as a mouse or a touch pad), translucent volume window 511 appears on screen 501. Since window 511 is translucent, the portion of window 503 under window 511 is still visible. In one embodiment of the present invention, when window 511 is initially loaded, the background of volume window 511 has a high degree of transparency; and the content of window 511 has a low degree of transparency (or no transparency). Therefore, the user can easily see the content of window 511 when the user is supposed to focus on window 511. As the user provides input to adjust the volume level, window 511 remains in a state with a high degree of transparency for the background and a low degree of transparency for the content. For example, when the user decreases the volume level (e.g., pressing a function key, or an array key), the volume level is decreased as indicated by window 521 in FIG. 17. When the user further decreases the volume level to mute the speakers, window 531 changes an icon to indicate that the speakers are muted, as shown in FIG. 18. When the user starts to adjust the brightness of the monitor, translucent brightness window 541 appears, as shown in FIG. 19, while the volume window is hidden (or destroyed, or converted into the brightness window by redrawing the icon and the level bar). If the user stops providing input for the brightness window for an amount of time (e.g., a predetermined amount of time, a randomly selected amount of time, a time period determined according to a system condition or other criteria, a time period calculated on the fly, or a time period specified by a user) window 541 starts to fade away and be destroyed, as shown in FIGS. 20 and 21. In one embodiment of the present invention, when a translucent window starts to fade away, the degree of transparency of the content in the translucent window is increased to allow the user to see better the window under the translucent window,

as illustrated by window 551 in FIG. 20. Thus, the degree of transparency of the window can be adjusted during the life cycle of the window to lead the focus point of the user. Further, a user may specify the degree of transparency of the window (e.g., as a preference parameter). The image of the window may fade out smoothly in an animation; or the image of the window may fade out in a number of discrete steps. The degree of translucency, the speed for fading out, the discrete levels of translucency for fading out, the time to expire, and/or other parameters for controlling the display of the window may be set by the user or adjusted by the system (or application software programs) automatically according to system conditions or other criteria. For example, the system (or application programs) may adjust the time to expire according to the number of translucent windows displayed concurrently on the screen; or the system (or an application program) may adjust the initial degree of translucency according to the color pattern at the location where the translucent window is displayed.

[0049] In the foregoing specification, the invention has been described with reference to specific exemplary embodiments thereof. It will be evident that various modifications may be made thereto without departing from the broader spirit and scope of the invention as set forth in the following claims. The specification and drawings are, accordingly, to be regarded in an illustrative sense rather than a restrictive sense.

1. A method of hierarchal displaying comprising:
 - displaying a plurality of windows on a graphical user interface in a hierarchy;
 - displaying a translucent region of display with the plurality of windows so that the translucent region of display is at the top of the hierarchy;
 - starting a timer; and
 - fading out the translucent region of display as the timer counts to a preset time.
2. The method of claim 1 additionally comprising destroying the translucent region of display when the timer reaches a preset time.
3. The method of claim 1 additionally comprising destroying the translucent region of display when the user interacts.
4. The method of claim 1 wherein an initial degree of translucency of the translucent region of display is dependent on a location of the translucent region of display.
5. A machine readable media containing instructions that, when executed by a data processing system, will cause said system to perform operations comprising:
 - displaying a plurality of windows on a graphical user interface in a hierarchy; displaying a translucent region of display with the plurality of windows so that the translucent region of display is at the top of the hierarchy;
 - starting a timer; and
 - fading out the translucent region of display as the timer counts to a preset time.
6. The machine readable media of claim 5 additionally comprising destroying the translucent region of display when the timer reaches a preset time.
7. The machine readable media of claim 5 additionally comprising destroying the translucent region of display when the user interacts.
8. The machine readable media of claim 5 wherein an initial degree of translucency of the translucent region of display is dependent on a location of the translucent region of display.

9. A method to inform a computer user of an incoming call comprising:

- displaying a graphical user interface;
- detecting an incoming call signal; and
- displaying a translucent message to inform the computer user of the incoming call signal.

10. The method of claim **9** additionally comprising fading out the translucent message.

11. The method of claim **9** additionally comprising fading out the translucent message when the incoming call signal ceases.

12. The method of claim **9** additionally comprising destroying the translucent message when the incoming call signal ceases.

13. A machine readable media containing instructions that, when executed by a data processing system, will cause said system to perform operations comprising: displaying a graphical user interface;

- detecting an incoming call signal; and
- displaying a translucent message to inform the computer user of the incoming call signal.

14. The machine readable media of claim **13** additionally comprising fading out the translucent message.

15. The machine readable media of claim **13** additionally comprising fading out the translucent message when the incoming call signal ceases.

16. The machine readable media of claim **13** additionally comprising destroying the translucent message when the incoming call signal ceases.

17. A method to show the progress of an operation, comprising:

- processing the operation;
- displaying a graphical user interface; and
- displaying a translucent progress indicator which indicates to a user the progress of the processing.

18. The method of claim **17** wherein the operation is transferring a file.

19. The method of claim **17** wherein the operation is installing software.

20. The method of claim **17** additionally comprising fading out the translucent progress indicator when the process is completed.

21-54. (canceled)

* * * * *