(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
18 December 2008 (18.12.2008)

PCT

(10) International Publication Number
WO 2008/154448 A2

(51) International Patent Classification:
G06F 11/14 (2006.01)

(21) International Application Number:
PCT/US2008/066212

(22) International Filing Date: 6 June 2008 (06.06.2008)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/943,027 8 June 2007 (08.06.2007) US
12/135,024 6 June 2008 (06.06.2008) US

(71) Applicant *(for all designated States except US)*: **APPLE INC.** [US/US]; 1 Infinite Loop, Cupertino, California 95014 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: **CISLER, Pavel** [US/US]; 15571 Loma Vista Avenue, Los Gatos, California 95032 (US). **LYONS, David** [US/US]; 4497 Poinsettia Court, San Jose, California 95136 (US). **ULRICH, Robert** [US/US]; 2421 Apsis Avenue, San Jose, California 95124 (US).

(74) Agent: **GUSTAFSON, Brian, J.**; Fish & Richardson P.C., P.O. Box 1022, Minneapolis, Minnesota 55440 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, NO, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

(54) **Title:** APPLICATION-BASED BACKUP-RESTORE OF ELECTRONIC INFORMATION

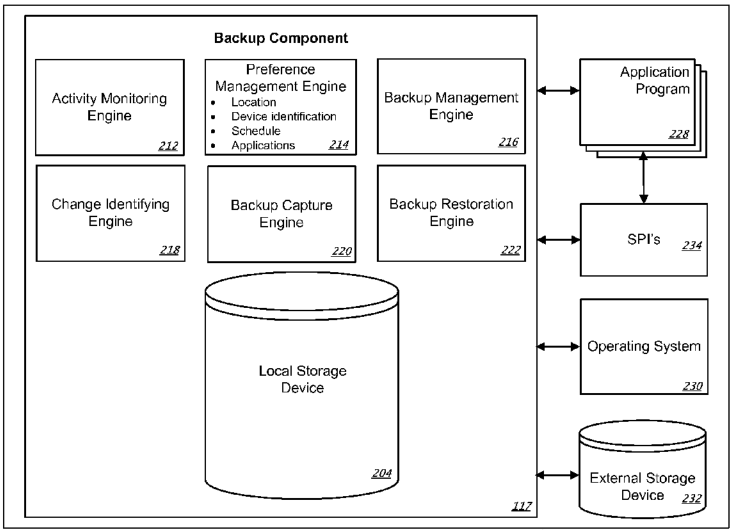


FIG. 2

(57) **Abstract:** Systems and methods are provides for storing and restoring digital data. In some implementations, a system is provided. The system includes a backup component, the backup component for capturing a state of a view of a user interface of an application as part of a backup operation and a first interface coupling the backup component and the application, the interface providing a communication link between the backup component and the application such that a backup user interface can display the captured state of the view. Other embodiments of this aspect include corresponding systems, apparatus, computer program products, and computer readable media.

APPLICATION-BASED BACKUP-RESTORE OF ELECTRONIC INFORMATION

TECHNICAL FIELD

[0001] The disclosed implementations relate generally to storing and restoring digital data.

BACKGROUND

[0002] Modern graphical user interfaces allow a large number of graphical objects or items to be displayed on a display screen at the same time. Operating systems, e.g., Apple Mac OS®, provide user interfaces in which a number of graphical user interface windows can be displayed, overlapped, resized, moved, configured, and reformatted according to the needs of a user or a particular application. Taskbars, menus, virtual buttons, and other user interface elements provide mechanisms for accessing and activating windows even when they are hidden behind other windows.

[0003] With the sophisticated tools available, users are encouraged not only to create and save a multitude of items in their computers, but to revise or otherwise improve on them over time. For example, a user can work with a certain file and thereafter save the file's current version on a storage device. The next day, however, the user could have had second thoughts about the revisions, or could have come up with new ideas, and therefore opens the file again.

[0004] The revision process is usually straightforward if the user wants to add more material to the file or make changes to what is there. But it is typically more difficult for a user who has changed his/her mind about changes that were previously made and wants the file returned to a previous version. Application programs for word processing typically let the user "undo" previous edits of a text, at least up to a predefined number of past revisions. The undo feature also usually is configured so that the previously made revisions must be undone in reverse chronological order; that is, the user must first undo the most recently made edit, then the second-most recent one, and so on. If the user saves and closes the document and thereafter opens the document again, it might not be possible to automatically undo any previous edits.

SUMMARY

[0005] Systems and methods are provides for storing and restoring digital data. In general, in one aspect, a system is provided. The system includes a backup component, the backup component for capturing a state of a view of a user interface of an application as part of a backup operation and a first interface coupling the backup component and the application, the interface providing a communication link between the backup component and the application such that a backup user interface can display the captured state of the view. Other embodiments of this aspect include corresponding methods, apparatus, computer program products, and computer readable media.

[0006] Implementations of the aspect can include one or more of the following features. The communication link provided by the first interface can include instructions describing how to present the backup user interface and the data to include in the backup user interface. The instructions describing how to present the backup user interface can include instructions to display the backup user interface and instructions to exit the backup user interface. Instructions describing the data to include in the backup user interface can include instructions for integrating backup data in a form of a user interface for the application.

[0007] The aspect can further include a second interface coupling the backup component to a system component, the second interface providing communication between the backup component and the system component. The communication provided by the second interface can include communication regarding a status of backup operations. The status of backup operations can include one or more of an initiation of a backup operation and a completion of a backup operation. The first interface can be a service provider interface. The application can be a mail application or an audio application.

[0008] In general, in one aspect, a method is provided. The method includes receiving from an application, while a current view of an application is displayed in a user interface, a first user input requesting that a history view associated with the current view of the application be displayed, retrieving information associated with the current view of the application, retrieving from a backup archive backup data associated with the history view, and providing the retrieved backup data and information to a backup component for presentation to the user. Other embodiments of this aspect include corresponding systems, apparatus, computer program products, and computer readable media.

[0009] Implementations of the aspect can include one or more of the following features. Retrieving information associated with the current view of the application can include retrieving format information. The format information can include information for generating a user interface layout for the application. The aspect can further include using the backup data and the format information to generate a plurality of visual representations corresponding to earlier versions of the current view of the application.

[0010] Particular embodiments of the subject matter described in this specification can be implemented to realize one or more of the following advantages. Service Provider Interfaces ("SPI's") can be used to provide efficient communication between a backup component and other system components including applications. Applications can take advantage of a backup interface to provide a backup interface for restoring data of the application.

[0011] The details of the various aspects of the subject matter described in this specification are set forth in the accompanying drawings and the description below. Other features, aspects, and advantages of the invention will become apparent from the description, the drawings, and the claims.

BRIEF DESCRIPTION FO THE DRAWINGS

[0012] FIG. 1 is a block diagram of an example computer system.

[0013] FIG. 2 is a block diagram of an example architecture for backup operations.

[0014] FIG. 3 is a screen shot depicting an example of a desktop user interface with multiple open applications.

[0015] FIG. 4 is a screen shot depicting an example of an initial state for a backup interface.

[0016] FIG. 5 is a screen shot depicting the backup interface of FIG. 4 where a snapshot has been selected.

[0017] FIG. 6 is a screen shot depicting the backup interface of FIG. 5 wherein a portion of a snapshot has been selected for restoration.

[0018] FIG. 7 is a screen shot showing the desktop user interface after some contents from an earlier version have been restored using the backup interface.

[0019] FIG. 8 is a screen shot depicting an example of a desktop user interface with multiple open applications.

[0020] FIG. 9 is a screen shot depicting an example of a state for a backup interface.

[0021] FIG. 10 is a screen shot depicting an example showing changed items in a backup interface.

[0022] FIG. 11 is a screen shot depicting an example backup interface.

[0023] FIG. 12 is a screen shot depicting the backup interface of Fig. 11 where a snapshot has been selected.

[0024] FIG. 13 is a screen shot depicting an example backup interface where contents of a snapshot have been selected.

[0025] FIG. 14 is a screen shot showing another example of the desktop user interface after some contents from an earlier version have been restored using the backup interface.

[0026] Like reference numbers and designations in the various drawings indicate like elements.

DETAILED DESCRIPTION

[0027] FIG. 1 is a block diagram of an example system 100. The system 100 can be used, for example, for capturing at least one earlier version of a user interface view and allowing a user to initiate a restoration based on the at least one earlier version. As used herein, a view refers to an item, element, or other content, capable of being stored and/or retrieved in an interface, that can be subjected to a backup operation by a backup component 117. For example, a user interface view can contain any number of icons, files, folders, application state information and/or machine state information, preferences, etc.

[0028] The system 100 includes a personal computer 102 communicatively coupled to a remote server 107 using a network interface 116 and a network 108 (e.g., local area network, wireless network, Internet, intranet, etc.). The computer 102 generally includes a processor 103, memory 105, one or more input devices 114 (e.g., keyboard, mouse, etc.) and one or more output devices 115 (e.g., a display device). A user interacts with the system 100 using the input and output devices 114, 115. The system 100 as shown includes various hardware elements. However, the system 100 can include hardware, software, and combinations of the two.

[0029] The computer 102 also includes a local storage device 106 and a graphics module 113 (e.g., a graphics card) for storing information and generating graphical objects, respectively. The local storage device 106 can be a computer-readable medium. The computer readable medium can be, for example, a machine-readable storage device, a machine-readable storage substrate, a memory device, a composition of matter effecting a machine-readable propagated signal, or a combination of one or more them.

[0030] While modifying a user interface view is described below with respect to a personal computer 102, it should be apparent that the disclosed implementations can be incorporated in, or integrated with, any electronic device that has a user interface, including without limitation, portable and desktop computers, servers, electronics, media players, game devices, mobile phones, e-mail devices, personal digital assistants (PDAs), embedded devices, televisions, other consumer electronic devices, etc.

[0031] One of ordinary skill in the art will recognize that the engines, methods, processes and the like that are described can themselves be an individual process or application, part of an operating system, a plug-in, an application or the like. In one implementation, the system and methods can be implemented as one or more plug-ins that are installed and run on the personal computer 102. The plug-ins are configured to interact with an operating system (e.g., MAC OS® X, WINDOWS XP, LINUX, etc.) and to perform various functions, e.g., as described below. A system and method for modifying a user interface view can also be implemented as one or more software applications running on the computer 102. Such a system and method can be characterized as a framework or model that can be implemented on various platforms and/or networks (e.g., client/server networks, wireless networks, stand-alone computers, portable electronic devices, mobile phones, etc.), and/or embedded or bundled with one or more software applications (e.g., e-mail, media player, browser, etc.).

[0032] The computer 102 includes the backup component 117 that allows for the storage of multiple versions of data including the computer's files or other items, for example within the local storage 106 or in an external storage repository. In one implementation, the backup component 117 also allows a user to select any of the stored versions and initiate a restoration of the selected version in the computer 102.

[0033] FIG. 2 is a block diagram of an example architecture 200 for backup operations. The architecture (e.g., a software architecture) 200 can be used to provide backup operations and restoration of backup data including application data (e.g., applications files, state, or other information) associated with one or more applications 228.

[0034] The backup component can interact with one or more applications 228 or the operation system 230 using one or more service provider interfaces ("SPI's") 234. The SPI's 234 facilitate interaction, for example, between the backup component 117 and a particular application 228. Particularly, the backup component 117 can utilize SPI techniques to enhance application programs with the backup functionality.

[0035] The SPI's can include tools for linking several applications 228 to the backup component 117 by using an SPI 234 to provide a similar menu or icon for each application 228 (e.g., an icon for initiating the backup component 117 from the user interface of the application 228). For example, the SPI 234 can be used by the backup component 117 when generating a backup version of a current state of the application 228 or when displaying a view of a previous state of the application and/or restoring application data. Examples of interactive communication provided to and from the backup component 117 using the SPI's 234 are described in greater detail below.

[0036] Although the SPI's 234 are shown in FIG. 2 as a separate component of the architecture 200, individual SPI's 234 can be part of applications 228, the operation system 230, or integrated in other components of system 100. Regardless of the location, the SPI's 234 can be made available by the system for use by, for example, one or more of the applications 228.

[0037] In one implementation, the backup component 117 provides backup operations and restoration capability for the system 100. Many different items or elements can be the subject of a backup operation. For example, data elements including folders, files, items, information portions, directories, images, system parameters, playlists, address books, e-mails, e-mail folders, a state of an application or state of the system, preferences (e.g., user or system preferences), and the like all can be candidates for inclusion in an archive of backup data during the backup operation. Other types of data can also be included in the backup data for the system 100. In this example, the backup component 117 includes a local storage device 204 and an external storage device 232. Backup data can be stored on either or both storage devices. Any number of local and/or external storage devices can be used by the backup component 117 for storing backup data representing one or more versions of data.

[0038] In one implementation, no local storage is provided. In one implementation, the backup component 117 runs as a background task on an operating system 230 that is not visible to the user. The backup component 117 can be capable of running across multiple user accounts.

[0039] The backup component 117 can interact with one or more of the applications 228 using the corresponding SPI 234. In one implementation, this can include providing backup operations for application data including application files, state or other items, and can further provide for user restoration of selected backup data. In one implementation, one copy of original data (e.g., folders, files, items, information portions, directories, images, system parameters, playlists, address books, e-mails, e-mail folders, application states, preferences, and the like) is stored in the system as backup data (e.g., an original version of the data). When one or more subsequent sets of backup data, or versions, of the data are generated through subsequent backup operations, the backup data can contain only the differences between a current data version and the prior backup data, thereby saving storage space. In some implementations, the storage devices 204 and 232 can be used to store the original backup data as well as links pointing to the original backup data. The links can be hard links which reference, or point to, physical data, or as another example can be symbolic links that reference another file on the storage device (e.g., by name or location). Different types of data can be scheduled to be stored on different devices or on different segments of a storage device during a backup operation. In one implementation, the backup component 117 stores the backup data of each backup operation in a format corresponding to a file system structure. As such, the backup data from each backup operation can be independently accessed and traversed as a file system hierarchy.

[0040] The backup component 117 includes an activity monitoring engine 212. In one implementation, the activity monitoring engine 212 monitors for changes within applications (e.g., application files or state) that are targeted for backup operations. A change can also include the addition of new files or other data structures, or deletion of existing ones.

[0041] In one implementation, the activity monitoring engine 212 is capable of discerning between a substantive change (e.g. the text within a document has been modified) and a non-substantive change (e.g. the play count within an iTunes playlist has been updated, or several changes cancel each other out) through the activity monitoring engine's 212 interaction with the applications 228. The activity monitoring engine 212 can, for example, create a list of modified elements to be used when a backup operation is eventually triggered. For example, in one implementation, only data associated with the modified elements can be included in the next backup operation.

[0042] In one implementation, the activity monitoring engine 212 can monitor the system for periods of inactivity. The activity monitoring engine 212 can then trigger a backup operation during a period of time in which the backup operation will not cause a system slowdown for an active user.

[0043] A preference management engine 214 specifies some operating parameters of the backup component 117. In one implementation, preference management engine 214 contains user-specified and/or system default application parameters for the backup component 117. These can include settings for the details of capturing and storing views of an application 228. For example, the preference management engine 214 can determine the frequency of a backup operations, the storage location for the backup data, the types of data elements (e.g., files or other items) that are to be included in the backup operations, and the events which can trigger a backup operation (periodic or event-driven, etc.).

[0044] In one implementation, the preference management engine 214 can detect that a new storage device has been added to the system and prompt the user whether the storage device should be included as a backup repository for storing backup data. Files and other items can be scheduled for a backup operation due to location (e.g. everything on the C: drive and within D:/photos), a correlation with specific applications (e.g. all pictures, music, e-mail in an inbox, an address book, and system settings), or a combination of strategies.

[0045] A backup management engine 216 coordinates the collection, storage, and retrieval of data (e.g., views) performed by the backup component 117. For example, the backup management engine 216 can trigger the activity monitoring engine 212 to monitor for activities that satisfy a requirement specified in the preference management engine 214.

[0046] A change identifying engine 218 locates specific elements (e.g., monitored files or other items within) to determine if the elements have changed. The change identifying engine 218 can be capable of discerning a substantive change from a non-substantive change, similar to the example described above for the activity monitoring engine 212. In one implementation, the change identifying engine 218 traverses a target set of elements (e.g., files or other items), comparing a previous version to the current version to determine whether or not a modification has occurred.

[0047] A backup capture engine 220 locates views that are to be included in the backup operation. The view (e.g., of an application) can include various elements (e.g., files, state, and other application information). The backup capture engine 220 can invoke the activity monitoring engine 212 and/or the change identifying engine 218, for example, to generate a capture list identifying data to include in a next backup operation. The backup capture engine 220 can then store data associated with the listed data in one or more targeted storage repositories. For example, the backup capture engine 220 can store the changed elements in a file system hierarchy that includes references (e.g., links) to unchanged elements. The backup capture engine 220 can track multiple version of each view included in the backup repository.

[0048] The backup component 117 includes a backup restoration engine 222 to restore previous versions of views (e.g., files or other items). In one implementation, the backup restoration engine 222 can provide a user interface (e.g., a graphical user interface) where a user can select a view or portions of a view (e.g., elements, items, etc.) to be restored.

[0049] The backup restoration engine 222 can initiate backup operations on one or more data elements or items as directed by a user, the system, or according to a specified or otherwise defined schedule. For example, the user can set up a scheduled backup operation to occur for any number of data files. The data files can be related to one another, such as when the files share similar attributes or a schema. For example, several files included on one particular webpage can have a similar schema and can require versions of each related file to remain consistent to display a successful webpage image.

[0050] The SPI's 234 can provide interaction between the backup component 117 and other aspects of the system 100. For example, one or more SPI's 234 can communicate various activities of the backup component 117 to the operating system 230. An SPI 234 can manage inquires regarding backup operations conducted by the backup component 117. For example, the location of backup data (e.g., a location on a particular storage device), the time when the last backup operation occurred, whether a particular element has been included in a prior backup operation, and whether a particular element is scheduled to be included in a next backup operation. Additionally, the system 100 can use the SPI 234 to instruct (e.g., sending a call) the backup component 117 to initiate a backup operation. The backup component 117 can provide a callback communicating that the backup operation has begun and/or that the backup operation is completed, interrupted, etc.

[0051] Application specific SPI's 234 allow the backup component 117 and one or more particular applications 228 to generate a backup interface that provides views of the particular applications. For example, a backup interface can be accessed through the applications 228 (e.g., though an icon or menu item in the application). The backup interface can provide a number of history views of the application in the form of a corresponding application user interface. While the backup interface has a general structure, the particular application can be used to provide details on constructing the specific user interface corresponding to the application (e.g., graphical elements, types of information presented, layout of information).

[0052] The SPI's 234 can facilitate the interaction between the backup component 117 and application 228, for example, by allowing the flow of communication and instructions to pass between the respective SPI and application. For example, an instruction from the application 228 to transition from a desktop user interface to a display of the backup user interface for the application (e.g., based on a user selected menu item or icon in the application); receiving, at the application 228, of backup data from the backup component 117 to present using in the form of the application user interface; and receiving, at the application 228, backup data for several versions of an application view with instructions to generate preview application user interfaces using the backup data. Other SPI instructions include instructions to retrieve different backup data, instructions to display different backup data, instructions to restore a particular portion of a view (e.g., file, elements, state of an application), and instructions to exit the backup interface.

[0053] FIG. 3 is a screen shot depicting an example of a user interface 300 (e.g., a desktop user interface) with multiple open applications (e.g., applications 228). The user interface 300 can be a user interface provided by an operating system (e.g., operating system 230). The user interface 300 includes a background, a menu bar 302, an application launch bar 304, and may include windows, icons, and other elements. Other configurations are possible. The user interface 300 can have multiple applications running, any or all of which can be presented in a separate window (e.g., a separate graphical representation in the user interface 300). While working with the applications, or at some other time, a user can initiate a backup interface.

[0054] As shown in FIG. 3 a user has launched an iTunes^{TM} application window 306 (available from Apple Inc., in Cupertino, California), which is currently presented in the foreground of the user interface 300, and a chat application 308 and a mail application 310 presented in the background of the user interface 300. In particular, the iTunes^{TM} application 306 has a user interface displaying contents of the user's library, which lists the user's songs. The iTunes^{TM} application 306 can be used for accessing, playing and organizing media, including digital music, images, and video files.

[0055] The backup component 117 can be activated, for example, to perform backup operations or to restore media content within the iTunes^{TM} application 306 or another application 228 using a backup interface. In one implementation, the user can activate the backup interface from the user interface using a user-selectable backup icon 312 in the application launch bar 304. As another example, the backup interface can be activated from the menu bar 302. The ability to activate the backup interface from the application 306 can be provided, for example, by an SPI.

[0056] The menu bar 302 can include a backup component options menu in addition to the activation selection. The backup component options menu can include selections allowing a user to specify data to include in a particular backup operation. For example, in iTunes^{TM}, the user can identify which content to include in backup operations (e.g., playlists, but not play counts, purchased songs, but not others, etc.).

[0057] In addition, the user can recover songs that were previously included, but are now missing from the iTunes^{TM} library. For example, the backup component can have performed at least one backup operation at an earlier time that included the missing songs. Consequently, the user can enter the backup interface to search for the missing songs or files in a backup snapshot of the iTunes^{TM} application. The user can then restore a view of the iTunes^{TM} application corresponding to that prior time (e.g., restoring a particular song or an entire application state). As used herein, a snapshot refers to backup data stored in a historical archive that corresponds to a particular view (e.g., of the application interface) taken at a particular point in time and can includes elements, items, or other information associated with the view as specified by the backup component 117. An SPI can facilitate the generation of snapshots corresponding to visual representations of the application interface.

[0058] In other implementations, the user can choose to enter the backup interface directly from an operating system menu. In some implementations, the backup component 117 can include several icons or menus available in other locations within the user interface 300 or running applications, thereby providing several options for activating or launching the backup interface.

[0059] FIG. 4 is a screen shot depicting an example of an initial state for a backup interface 400 after the backup interface has been initiated. The backup interface 400 includes snapshots 402 including a current snapshot 401, and various function buttons described below. As shown, the current snapshot 401 is displaying the most recent snapshot of the iTunes^{TM} application interface because the backup interface was entered using the iTunes^{TM} application interface. The current snapshot 401 can show the contents corresponding to the currently selected snapshot, or a portion thereof. In this particular example, a date is presented beneath the snapshot indicating when the snapshot was taken (e.g., when the corresponding backup operation was performed).

[0060] In some implementations, the user can select elements (e.g., items or other content) within the snapshots. For example, the user can select a particular snapshot from the snapshots 402, representing a previous view of the iTunes^{TM} application corresponding to a particular point in time, and next select one or more songs to restore using the backup interface 400. For example, the selected one or more songs can be songs that are missing from the current library.

[0061] In some implementations, the snapshots 402 can include a number of individual snapshots represented in a stack, each of which represents earlier versions or states of the iTunes^{TM} library that have been backed up in previous backup operations. Each snapshot provides a screenshot representation of the earlier version of the iTunes^{TM} library at a particular point in time. An SPI (e.g., SPI 229) can be used to generate the snapshots such that a representation of the application user interface is displayed. A specified number of snapshots 402 can be presented in the backup interface 400 at any given time. However, there can be many more snapshots than are presently shown.

[0062] A user can select a particular snapshot of the snapshots 402 by selecting a snapshot from the stack, causing that snapshot to become the current snapshot 401. Alternatively, the user can use one or more navigation buttons to move between snapshots 402. For example, arrow buttons 406a and 406b can be used to navigate snapshots 402 backward or forward in time, respectively. Alternatively, additional snapshots can be represented in the backup interface 400, for example, as a set of tick marks (not shown). The user can select a particular tick mark in order to bring a particular snapshot as the current snapshot 401.

[0063] The backup interface 400 can also include function controls. For example, the backup interface 400 can include a restore button 410 that, when selected, restores the view to the selected state represented by the selected snapshot. In some implementations, this also exits the backup interface. For example, a user can select one element in a snapshot and then select the restore button 410 to modify the current version of the element selected (e.g., restore the state of the view). For example, in iTunes^{TM}, the user can select a few songs to restore, and this can trigger the restore button to display a more precise message, such as "restore selection."

[0064] In one implementation, a "changed items only" button 412 filters the snapshots to show only those that differ from the current state. In one implementation, the "changed items only" button 412 does not refer to the incremental changes between snapshots 402, but rather when invoked acts to omit those snapshots whose states are identical to the current state of the iTunes^{TM} library from presentation in the snapshots 402. For example, if the most recent snapshot is identical to another snapshot that occurs earlier in time, selecting the changed items only button 412, in one implementation, causes the backup interface to cease displaying one of these versions, e.g., by removing the earlier snapshot from display. This can help the user to view only snapshots that contain changes to the current version.

[0065] A "done" button 418 can be selected to exit the backup interface 400 and return the user to the desktop user interface 300. In some implementations, the backup interface can automatically exit upon a particular snapshot being restored. In some implementations, the user can minimize the backup interface for purposes of navigating to other applications, such as an e-mail application or a web browser.

[0066] FIG. 5 is a screen shot depicting the backup interface 400 where an earlier snapshot has been selected from the snapshots 402. The iTunes^{TM} application interface corresponding to the selected snapshot (labeled March 3, 2007) is displayed as the current snapshot 501. This earlier version of the iTunes^{TM} application shows a file tree where the user has selected a source library 506, which can contain several songs and albums. The songs and albums can be sorted and arranged, e.g., by name, length, artist, or album title, to name a few examples. In some implementations, other sorting fields can be used, such as sorting by genre. In addition, the user can use the information in the source library 506 to create separate song lists known as playlists. Playlists can combine several artists and albums in one list and can be presented in a user-selected order. Playlists can also be burned to compact discs, transferred to an iPod^{TM} device (available from Apple Inc., in Cupertino, California), or played on the computer 102.

[0067] In this example, the source library 506 displayed in the current snapshot 501 includes songs that are not found in the current version of the library outside of the backup interface (e.g., the library shown in FIG. 3). In particular, the songs include twelve songs by the artist "Gorillaz." For example, the user could have lost these twelve songs from the source library (e.g., by deleting them or some system error) after the earlier version was backed up (i.e., sometime after March 3, 2007). The user may wish to restore these twelve missing songs and can do so by using the backup interface 400.

[0068] FIG. 6 is a screen shot depicting the backup interface 400 of FIG. 5 wherein a portion of the current snapshot 502 has been selected for restoration. For example, the user can select the twelve missing songs 602 within the current snapshot 502. In some implementations, upon selecting the missing songs 602, the restore button 410 can change, as shown, from "Restore All" to restore button 610 "Restore Selected". If the user selects the "Restore Selected" button, the selected songs will be restored to the library of the current iTunes^{TM} application (e.g., of FIG. 3). Upon selecting the restore button 610, the backup interface 400 can exit and return the user to the user interface 300 of FIG. 3 where the library of the current iTunes^{TM} application reflects the restoration of the missing songs. In some implementations, the user can choose to remain in the backup interface 400 to perform further restorations.

[0069] FIG. 7 is a screen shot showing the desktop user interface 300 after some contents from an earlier version have been restored using the backup interface 400. As shown, the iTunes^{TM} application user interface 700 includes a library 706 that now includes the restored twelve songs. In this example, actual data content represented by one snapshot was collected and restored in the iTunes^{TM} application. For example, the backup component 117 can retrieve the selected songs and restore them (e.g., copy them) to their original location in the file system.

[0070] Alternatively, the backup component can restore only the organization of the data. For example, in the iTunes^{TM} application, the user can have a number of different playlists, such as the "Party Shuffle" playlist 702. Each playlist can contain a different collection of songs. In general, songs which appear in more than one playlist do not correspond to multiple copies of the song because the playlist is just considered a particular organization of songs in the library 706. Therefore, if the user restores a playlist using the backup component, only the playlist organization can require restoration. Thus, the data corresponding to the underlying songs does not require restoration in such an example, unless they are also missing from the source library 706. These and other types of non-file form data (e.g., metadata or other application data) can be restored using the backup component.

[0071] FIG. 8 is a screen shot depicting an example of a desktop user interface 800 with multiple open applications. A user has launched an iPhoto^{TM} application 802, (available from Apple Inc., in Cupertino, California) which is currently presented in the foreground. Additionally, interfaces for a chat application 804, and a mail application 806 are presented in the background. In particular, the iPhoto^{TM} application 802 includes an interface that displays contents of the user's library 810, which lists the user's photo albums, each of which can include zero or more photographs or other media content. The iPhoto^{TM} application 802 can be used for accessing, storing, and organizing media, such as images, scanned photographs, and video content. While currently interacting with the iPhoto^{TM} application 802, or at some other time, a user can initiate a backup interface with respect to the iPhoto^{TM} application 802.

[0072] As shown in FIG. 8, the interface for the iPhoto^{TM} application 802 includes a source pane 808 for navigating the content stored in the library 810. For example, the user can select a particular photo album to view the contents in the photo album (e.g. photographs, images, and video content). The contents can be displayed, for example, as thumbnails in a display pane 803. In this example, the user has selected a "Library" item 810, and the iPhoto^{TM} application 802 has presented, in the interface, images in the library 810.

[0073] In some implementations, the user can select one or more photo albums to display the contents of each such album in the iPhoto^{TM} application 802. For example, the user can select a "Last 12 Months" photo album 812, and this can display all images collected in the selected album in the display pane 803. In some implementations, the contents can be displayed as thumbnail images, such as thumbnail image 814. In other implementations, the contents can be displayed in list form using the file name as the displayed image. The contents in each photo album can be different sets of images, for example, organized by event, theme, time, category, and the like.

[0074] In some implementations, a photograph count 816 can be presented to the user showing how many images are in the library 810 or in the currently selected photo album. In general, the user can create, delete, or transfer images in the photo albums and the photograph count 816 can adjust accordingly when a photograph is added or deleted, or when the user switches between photo albums. As shown, the photograph count 816 displays “1,973 photos,” indicating that there are currently 1,973 photographs in the library 810.

[0075] In some implementations, in addition to creating, deleting and transferring images in the photo album, the user can also modify the images by selecting various photograph options shown in a control panel 818. The control panel 818 can include options allowing the user to rotate, edit, e-mail, or generally move the images into other configurations. For example, the control panel 818 can present options to the user to select images from several photo albums to create a book or calendar entry using the selected images. As another example, the control panel 818 provides that the user can search for images using the application.

[0076] FIG. 9 is a screen shot depicting an example of a state for a backup interface 900 after the backup interface has been initiated. The backup interface 900 includes snapshots 902 including a current snapshot 904, and function buttons. The current snapshot 904 is displaying a snapshot of the iPhoto^{TM} application interface. The current snapshot 904 is displaying as encompassing a range from “May 9 – May 23, 2007”, where the range indicates the time in which the displayed contents of the current snapshot 904 were unchanged. Generally, each snapshot 902 provides a representation of the iPhoto^{TM} application interface including the various elements, items, and layout of the iPhoto^{TM} application 802 corresponding to the period of time covered by the snapshot.

[0077] The snapshots 902 can include a number of individual snapshots represented in a stack, each of which provide a visual representation of earlier versions or states of the iPhoto^{TM} library that have been backed up in previous backup operations. Thus, each snapshot 902 provides a screenshot representation of the earlier version of the iPhoto^{TM} library at a particular point in time. A specified number of snapshots 902 can be presented in the backup interface 900 at any given time. However, there can be many more snapshots than are presently shown.

[0078] A user can select a particular snapshot by selecting a snapshot from the stack, causing that snapshot to become the current snapshot 904. Alternatively, the user can use one or more navigation buttons to move between snapshots 902. For example, arrow buttons 906a and 906b can be used to navigate snapshots 902 backward or forward in time, respectively. Alternatively, additional snapshots can be represented in the backup interface 900, for example, as a set of tick marks (not shown). The user can select a particular tick mark in order to bring a particular snapshot as the current snapshot 904.

[0079] The backup interface 900 can also include function controls. For example, the backup interface 900 a restore button 910 that, when selected, restores the view to the selected state represented by the selected snapshot. In some implementations, this also exits the backup interface. A user can select an element in a snapshot and then select the restore button 910 to modify the current version of the selected element (e.g., restore the state of the view). For example, in the iPhoto^{TM}, the user can select a few photos to restore, and this can trigger the restore button to display a more precise message, such as "restore selection."

[0080] In some implementations, the user can select a "show changed items only" button 912 to modify the content to be presented in the interface 902. For example, when the button "show changed items only" 912 has been selected, the timeline can adjust to display only items with reference to the most recent snapshot 904. An example of selecting the "show changed items only" button 912 is described with reference to FIG. 10.

[0081] FIG. 10 is a screen shot depicting an example showing changed items in a backup interface. In particular, the user has checked the "show changed items only" button 912 for purposes of identifying only the changed items with reference to the most recent snapshot (i.e., snapshot 904 of FIG. 9). When the "show changed items only" button 912 is selected, all images that are present in the most recent snapshot are deleted from the prior snapshots. For example, the "January 9 – January 23, 2005" snapshot 920 shows four photographs that are not present in the snapshot 904 of FIG. 9. Thus, when the button 912 is selected, the content of each snapshot is presented relative to the content of the most recent snapshot. In another implementation, the button 912 can filter out snapshots 902 that are identical to the current snapshot, and leave the remaining snapshots intact.

[0082] FIG. 11 is a screen shot depicting an example backup interface 900 after the backup interface has been initiated. The iPhoto^{TM} application interface is represented in a current snapshot 1104. A number of snapshots 902 are represented in a stack. When a snapshot 902 is selected as the current snapshot 1104, an associated date is presented indicating when the snapshot was taken (e.g., when the backup operation occurred). In some implementations, a date range can be displayed indicating that the content in the currently displayed snapshot is the same throughout the time period, as shown, for example, in snapshot 1104. A date range can be used to compress several unchanged snapshots to a single snapshot. The restore 910 function button can operate substantially as described earlier with reference to FIG. 4. Each snapshot 902 can be a representation of the iPhoto™ application showing the state of a library 1114 for each snapshot 902. The user can navigate between snapshots 902 with the current snapshot 1104 corresponding to the selected snapshot 902.

[0083] FIG. 12 is a screen shot depicting the backup interface 900 of FIG. 11 where a new current snapshot 1202 has been selected from the snapshots 902. The iPhoto™ application interface corresponding to the snapshot 1202 (corresponding to a displayed date range of January 9 – January 23, 2005) is displayed. As shown, the number of photographs 816 in the snapshot 1202 is 1,912, whereas the most recent snapshot (snapshot 904 of FIG. 9) contains 1,973 pictures.

[0084] The user can search the contents of the snapshot 1202, e.g., for a photo album that is not present in the current application interface (e.g., as shown in FIG. 8) such as the "Euro 2004" album 1204 shown in snapshot 1202. Accordingly, the user can navigate to snapshot 1202 to access the content of the album 1204, for example, to restore the contents of that album.

[0085] FIG. 13 is a screen shot depicting the backup interface 900 where the snapshot 1202 has been selected from the snapshots 902. However, the user has selected the album 1204 "Euro 2004" within the snapshot 902. Consequently, the display pane 1302 of the snapshot 1202 now displays the contents of the "Euro 2004" album 1204. The user can select a restore button 910 to restore the content of the "Euro 2004" album. As another example, the user can choose to restore only a portion of the album 1204 shown in snapshot 1202 by selecting particular images in the display pane 1302. For example, to only restore image 1314, the user can select the image 1304 and then select the restore button 910.

[0086] Selecting the restore button 910 causes the album 1204, or a selected portion thereof (e.g., image 1314), to be restored in the current view of the iPhoto™ application and is thereafter accessible to the user. For example, the entire album 1204 can be restored to the library in the iPhoto™ application allowing the user to modify the contents in the album 1204. In some implementations, the caption of the restore button 910 can change depending on the user's selection, for example, to "Restore 1 image" when a user selects one image.

[0087] FIG. 14 is a screen shot showing another example of the desktop user interface 800 after some contents from a snapshot representing a previous view of the application have been restored using the backup interface. Particularly, the user here selected the Euro 2004 folder 1304 for restoration from the snapshot 1202. In FIG. 14, the iPhoto^{TM} library 1114 now includes the restored album “Euro 2004” 1304. Both the album and the corresponding image content within the album have been restored to the iPhoto^{TM} application. In some implementations, the user can choose to restore only particular image content. For example, images can be selected from a backup snapshot and restored in a new album (e.g., a “restored” album). Alternatively, the selection can be stored in an existing album.

[0088] The above examples involve backup versions of music libraries or photo albums. Many different types of applications, items, system settings, preferences, elements or processes can be backed up and made accessible through a backup interface.

[0089] In another implementation, the backup interface can be used to present backup information associated with a messaging application. For example, the user can enter the backup interface to restore one or more deleted friends from a “buddy list”, deleted conversations, preferences, and the like.

[0090] In the above description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding. It will be apparent, however, to one skilled in the art that implementations can be practiced without these specific details. In other instances, structures and devices are shown in block diagram form in order to avoid obscuring the disclosure.

[0091] In particular, one skilled in the art will recognize that other architectures and graphics environments can be used, and that the examples can be implemented using graphics tools and products other than those described above. In particular, the client/server approach is merely one example of an architecture for providing the functionality described herein; one skilled in the art will recognize that other, non-client/server approaches can also be used. Some portions of the detailed description are presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

[0092] It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the discussion, it is appreciated that throughout the description, discussions utilizing terms such as “processing” or “computing” or “calculating” or “determining” or “displaying” or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system’s registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

[0093] An apparatus for performing the operations herein can be specially constructed for the required purposes, or it can comprise a general-purpose computer selectively activated or reconfigured by a computer program stored in the computer. Such a computer program can be stored in a computer readable storage medium, such as, but is not limited to, any type of disk including floppy disks, optical disks, CD-ROMs, and magnetic-optical disks, read-only memories (ROMs), random access memories (RAMs), EPROMs, EEPROMs, magnetic or optical cards, or any type of media suitable for storing data including electronic instructions, and each coupled to a computer system bus.

[0094] The algorithms and modules presented herein are not inherently related to any particular computer or other apparatus. Various general-purpose systems can be used with programs in accordance with the teachings herein, or it can prove convenient to construct more specialized apparatuses to perform the method steps. The required structure for a variety of these systems will appear from the description. In addition, the present examples are not described with reference to any particular programming language. It will be appreciated that a variety of programming languages can be used to implement the teachings as described herein. Furthermore, as will be apparent to one of ordinary skill in the relevant art, the modules, features, attributes, methodologies, and other aspects can be implemented as software, hardware, firmware or any combination of the three. Of course, wherever a component is implemented as software, the component can be implemented as a standalone program, as part of a larger program, as a plurality of separate programs, as a statically or dynamically linked library, as a kernel loadable module, as a device driver, and/or in every and any other way known now or in the future to those of skill in the art of computer programming. Additionally, the present description is in no way limited to implementation in any specific operating system or environment.

[0095] The subject matter described in this specification can be implemented as one or more computer program products, i.e., one or more modules of computer program instructions encoded on a computer readable medium for execution by, or to control the operation of, data processing apparatus. The instructions can be organized into modules (or engines) in different numbers and combinations from the exemplary modules described. The computer readable medium can be a machine-readable storage device, a machine-readable storage substrate, a memory device, a composition of matter effecting a machine-readable propagated signal, or a combination of one or more them. The term "data processing apparatus" encompasses all apparatus, devices, and machines for processing data, including by way of example a programmable processor, a computer, or multiple processors or computers. The apparatus can include, in addition to hardware, code that creates an execution environment for the computer program in question, e.g., code that constitutes processor firmware, a protocol stack, a database management system, an operating system, or a combination of one or more of them. A propagated signal is an artificially generated signal, e.g., a machine-generated electrical, optical, or electromagnetic signal, that is generated to encode information for transmission to suitable receiver apparatus.

[0096] While this specification contains many specifics, these should not be construed as limitations on the scope of what may be claimed, but rather as descriptions of features specific to particular implementations of the subject matter. Certain features that are described in this specification in the context of separate embodiments can also be implemented in combination in a single embodiment. Conversely, various features that are described in the context of a single embodiment can also be implemented in multiple embodiments separately or in any suitable subcombination. Moreover, although features may be described above as acting in certain combinations and even initially claimed as such, one or more features from a claimed combination can in some cases be excised from the combination, and the claimed combination may be directed to a subcombination or variation of a subcombination.

[0097] Similarly, while operations are depicted in the drawings in a particular order, this should not be understood as requiring that such operations be performed in the particular order shown or in sequential order, or that all illustrated operations be performed, to achieve desirable results. In certain circumstances, multitasking and parallel processing may be advantageous. Moreover, the separation of various system components in the embodiments described above should not be understood as requiring such separation in all embodiments, and it should be understood that the described program components and systems can generally be integrated together in a single software product or packaged into multiple software products.

[0098] The subject matter of this specification has been described in terms of particular embodiments, but other embodiments can be implemented and are within the scope of the following claims. For example, the actions recited in the claims can be performed in a different order and still achieve desirable results. As one example, the processes depicted in the accompanying figures do not necessarily require the particular order shown, or sequential order, to achieve desirable results. In certain implementations, multitasking and parallel processing may be advantageous. Other variations are within the scope of the following claims.

What is claimed is:

CLAIMS

1. A system comprising:

a backup component, the backup component for capturing a state of a view of a user interface of an application as part of a backup operation; and

a first interface coupling the backup component and the application, the interface providing a communication link between the backup component and the application such that a backup user interface can display the captured state of the view.

2. The system of claim 1, the communication link provided by the first interface includes instructions describing how to present the backup user interface and the data to include in the backup user interface.

3. The system of claim 2, where the instructions describing how to present the backup user interface includes instructions to display the backup user interface and instructions to exit the backup user interface.

4. The system of claim 2, where instructions describing the data to include in the backup user interface include instructions for integrating backup data in a form of a user interface for the application.

5. The system of claim 1, further comprising:

a second interface coupling the backup component to a system component, the second interface providing communication between the backup component and the system component.

6. The system of claim 5, where the communication provided by the second interface includes communication regarding a status of backup operations.

7. The system of claim 6, where the status of backup operations includes one or more of an initiation of a backup operation and a completion of a backup operation.

8. The system of claim 1, where the first interface is a service provider interface.

9. The system of claim 1, where the application is a mail application.

10. The system of claim 1, where the application is an audio application.

11. A method comprising:

receiving from an application, while a current view of an application is displayed in a user interface, a first user input requesting that a history view associated with the current view of the application be displayed;

retrieving information associated with the current view of the application;

retrieving from a backup archive backup data associated with the history view; and

providing the retrieved backup data and information to a backup component for presentation to the user.

12. The method of claim 11, where retrieving information associated with the current view of the application includes retrieving format information.

13. The method of claim 12, where the format information includes information for generating a user interface layout for the application.

14. The method of claim 13, further comprising:

using the backup data and the format information to generate a plurality of visual representations corresponding to earlier versions of the current view of the application.

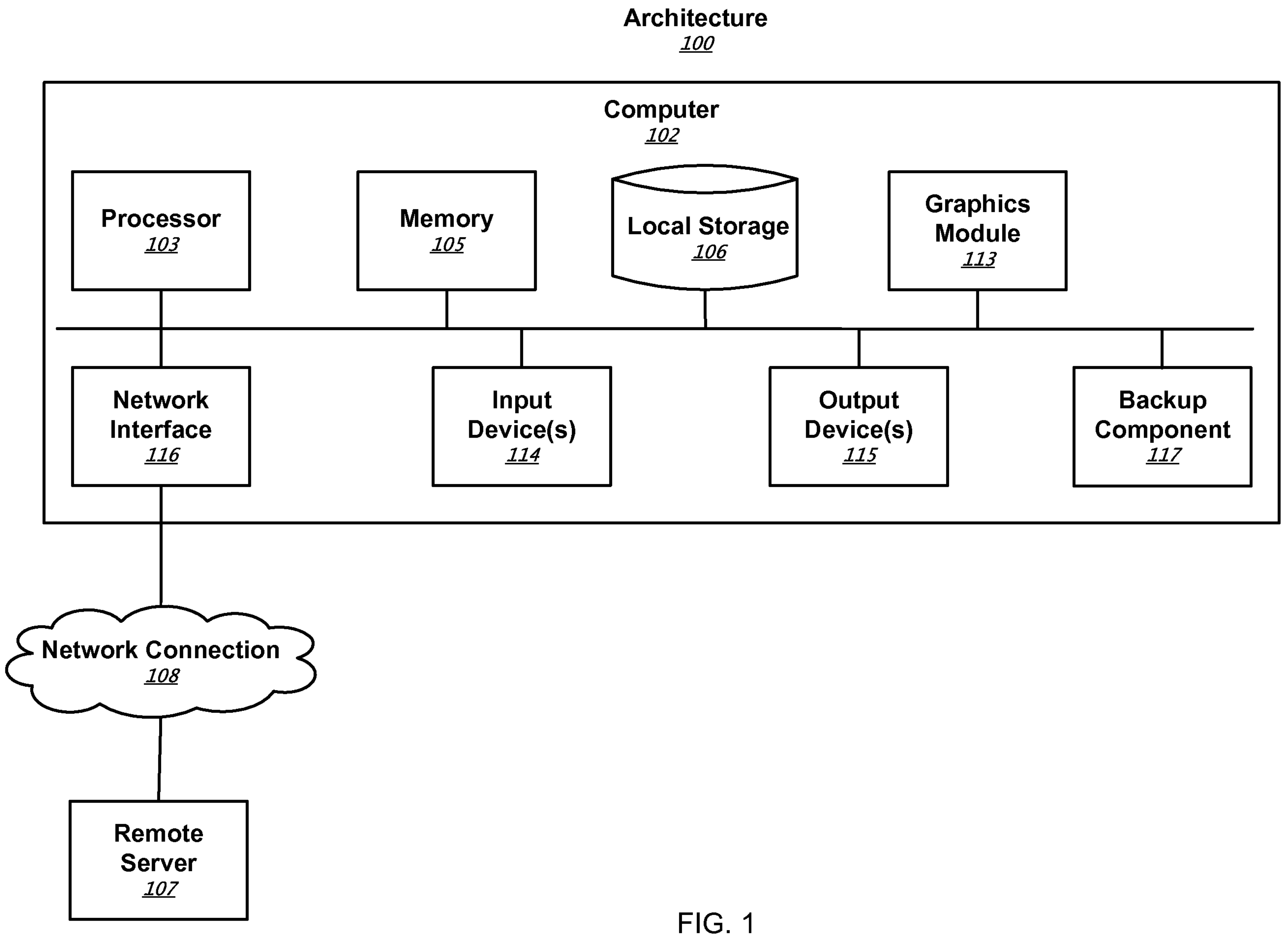

Architecture
100
Computer
102
Processor
103
Memory
105
Local Storage
106
Graphics
Module
113
Network
Interface
116
Input
Device(s)
114
Output
Device(s)
115
Backup
Component
117
Network Connection
108
Remote
Server
107

FIG. 1

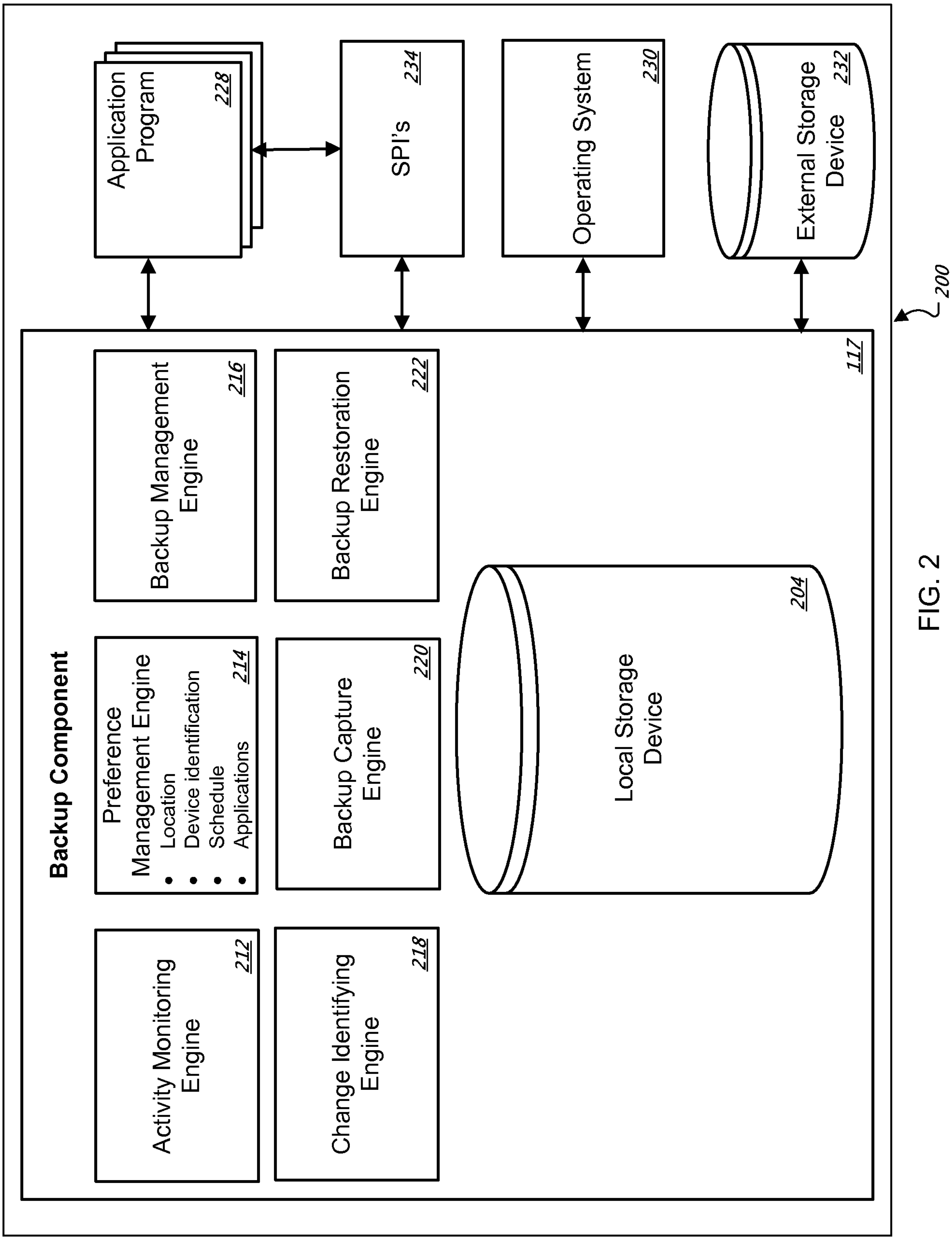
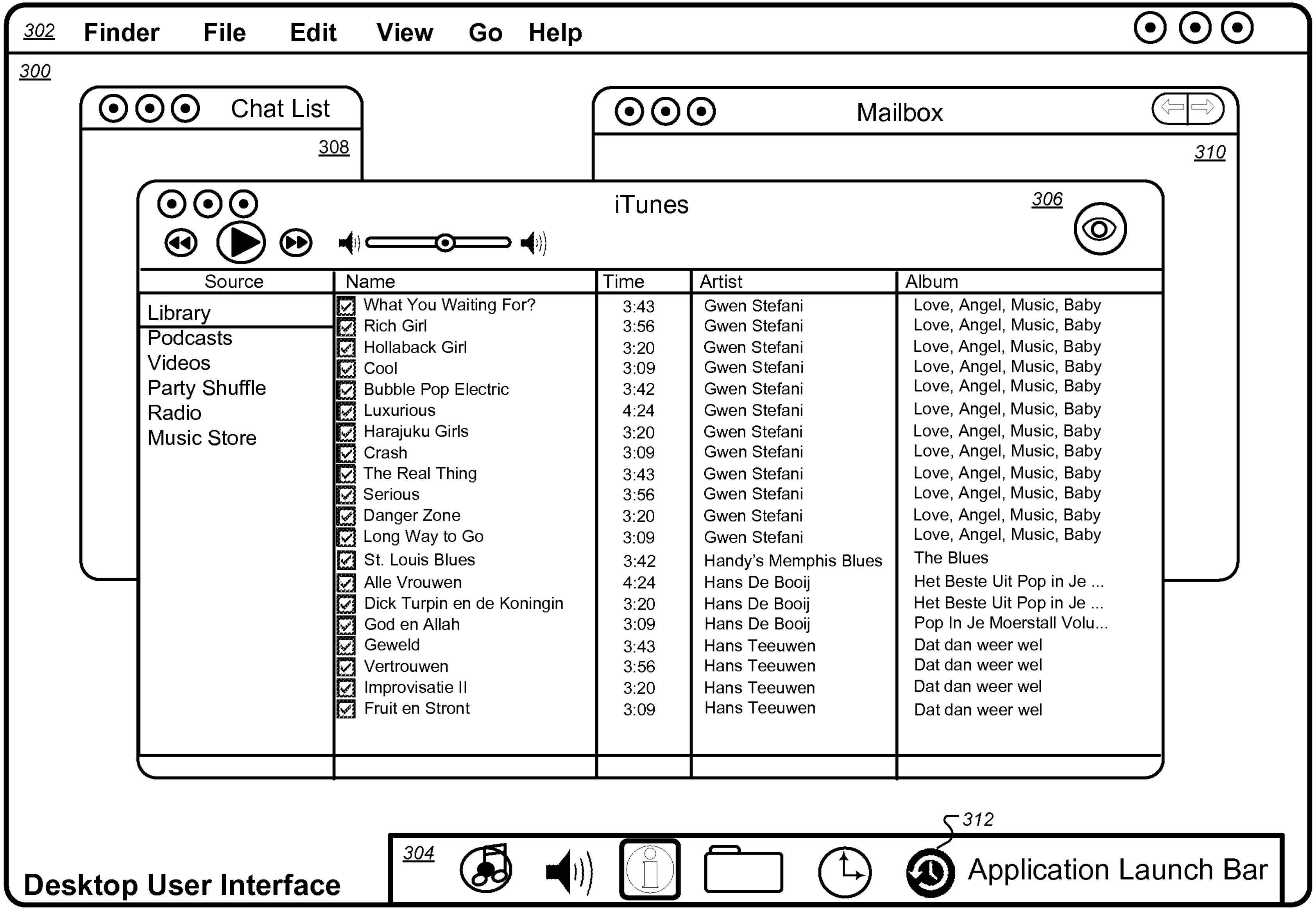
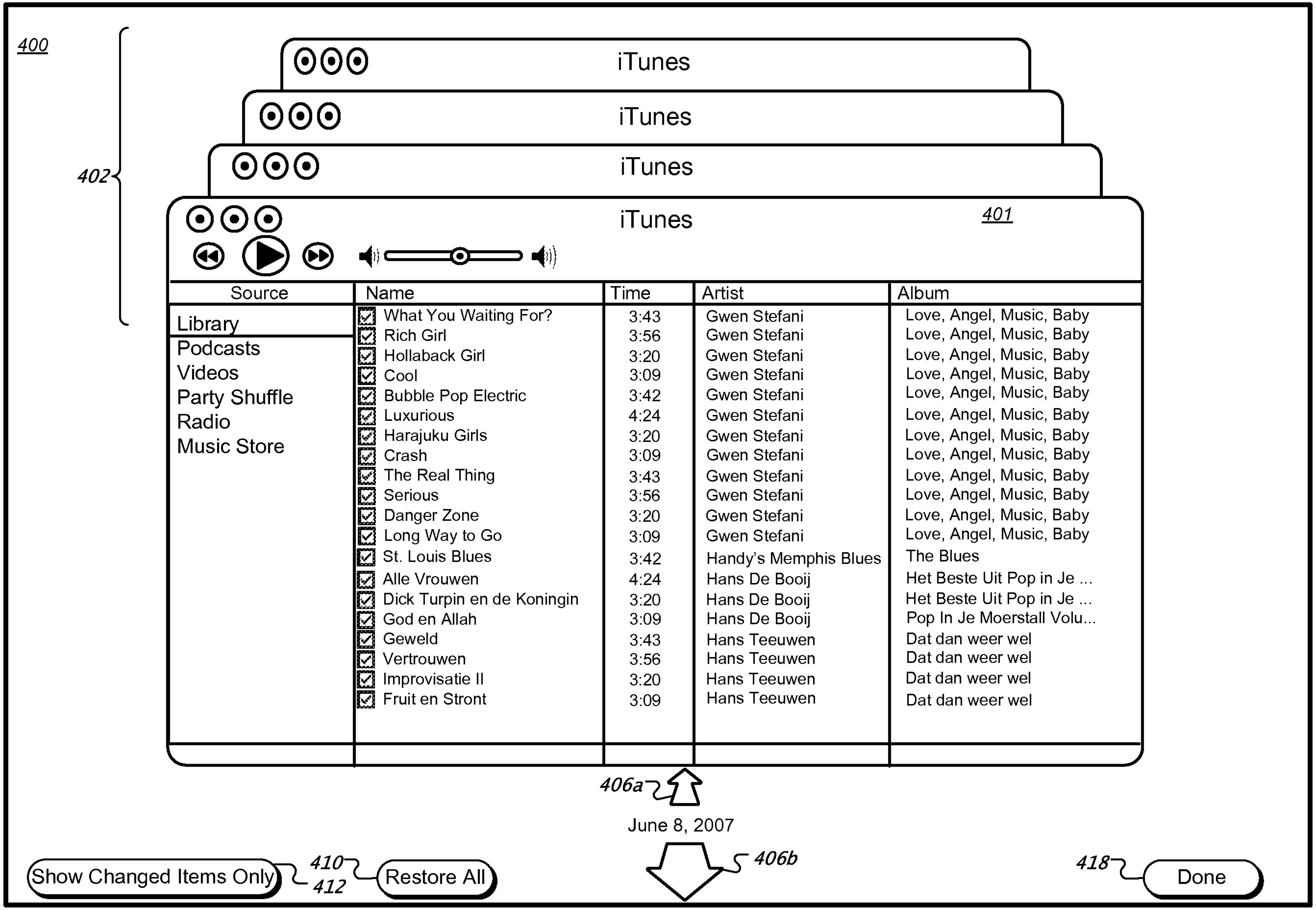

Backup Component
Activity Monitoring Engine
212
Preference Management Engine
• Location
• Device identification
• Schedule
• Applications
214
Backup Management Engine
216
Change Identifying Engine
218
Backup Capture Engine
220
Backup Restoration Engine
222
Local Storage Device
204
117
Application Program
228
SPI's
234
Operating System
230
External Storage Device
232
200

FIG. 2



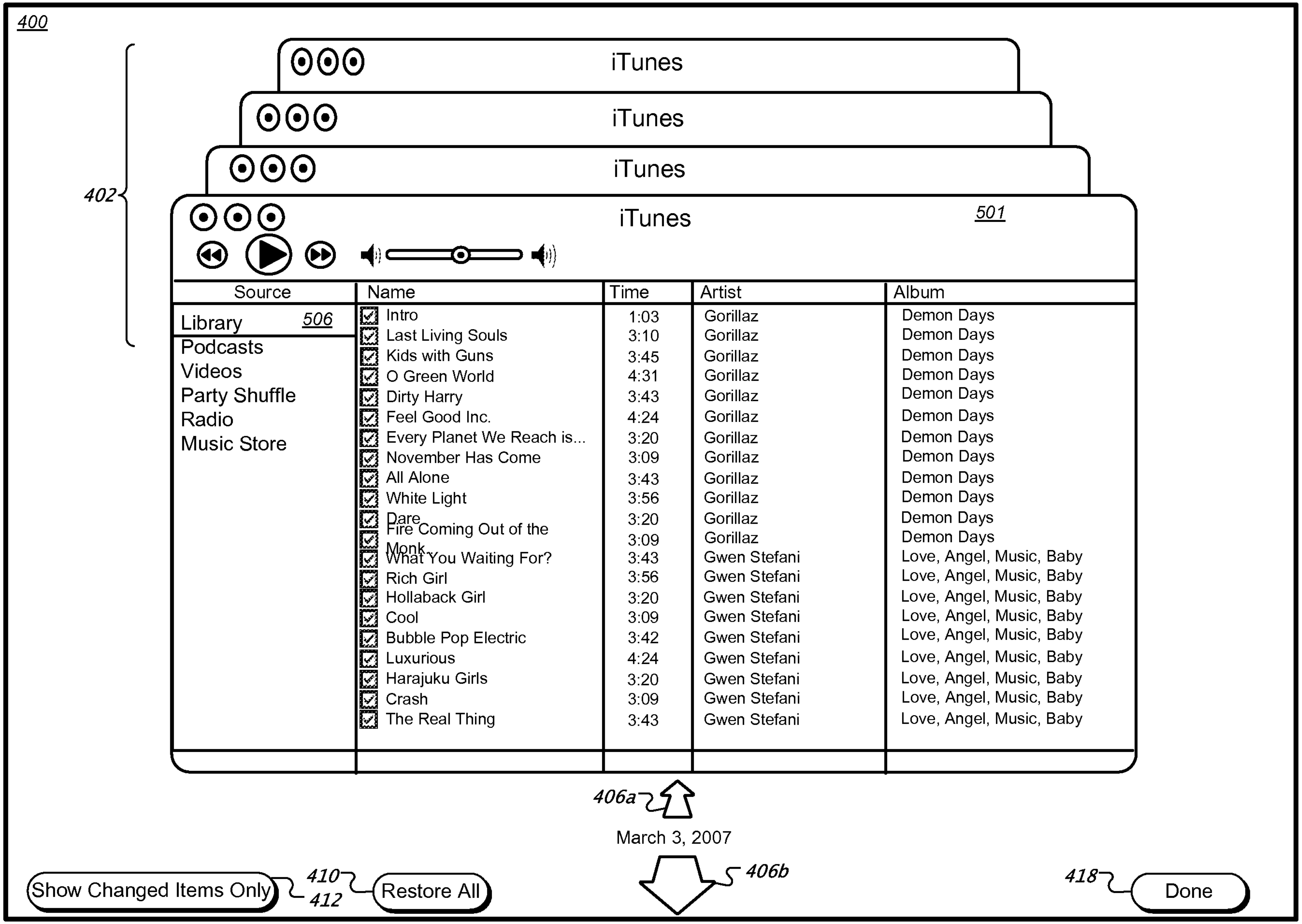
Name	Time	Artist	Album
☑ What You Waiting For?	3:43	Gwen Stefani	Love, Angel, Music, Baby
☑ Rich Girl	3:56	Gwen Stefani	Love, Angel, Music, Baby
☑ Hollaback Girl	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Cool	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ Bubble Pop Electric	3:42	Gwen Stefani	Love, Angel, Music, Baby
☑ Luxurious	4:24	Gwen Stefani	Love, Angel, Music, Baby
☑ Harajuku Girls	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Crash	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ The Real Thing	3:43	Gwen Stefani	Love, Angel, Music, Baby
☑ Serious	3:56	Gwen Stefani	Love, Angel, Music, Baby
☑ Danger Zone	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Long Way to Go	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ St. Louis Blues	3:42	Handy's Memphis Blues	The Blues
☑ Alle Vrouwen	4:24	Hans De Booij	Het Beste Uit Pop in Je ...
☑ Dick Turpin en de Koningin	3:20	Hans De Booij	Het Beste Uit Pop in Je ...
☑ God en Allah	3:09	Hans De Booij	Pop In Je Moerstall Volu...
☑ Geweld	3:43	Hans Teeuwen	Dat dan weer wel
☑ Vertrouwen	3:56	Hans Teeuwen	Dat dan weer wel
☑ Improvisatie II	3:20	Hans Teeuwen	Dat dan weer wel
☑ Fruit en Stront	3:09	Hans Teeuwen	Dat dan weer wel

FIG. 3



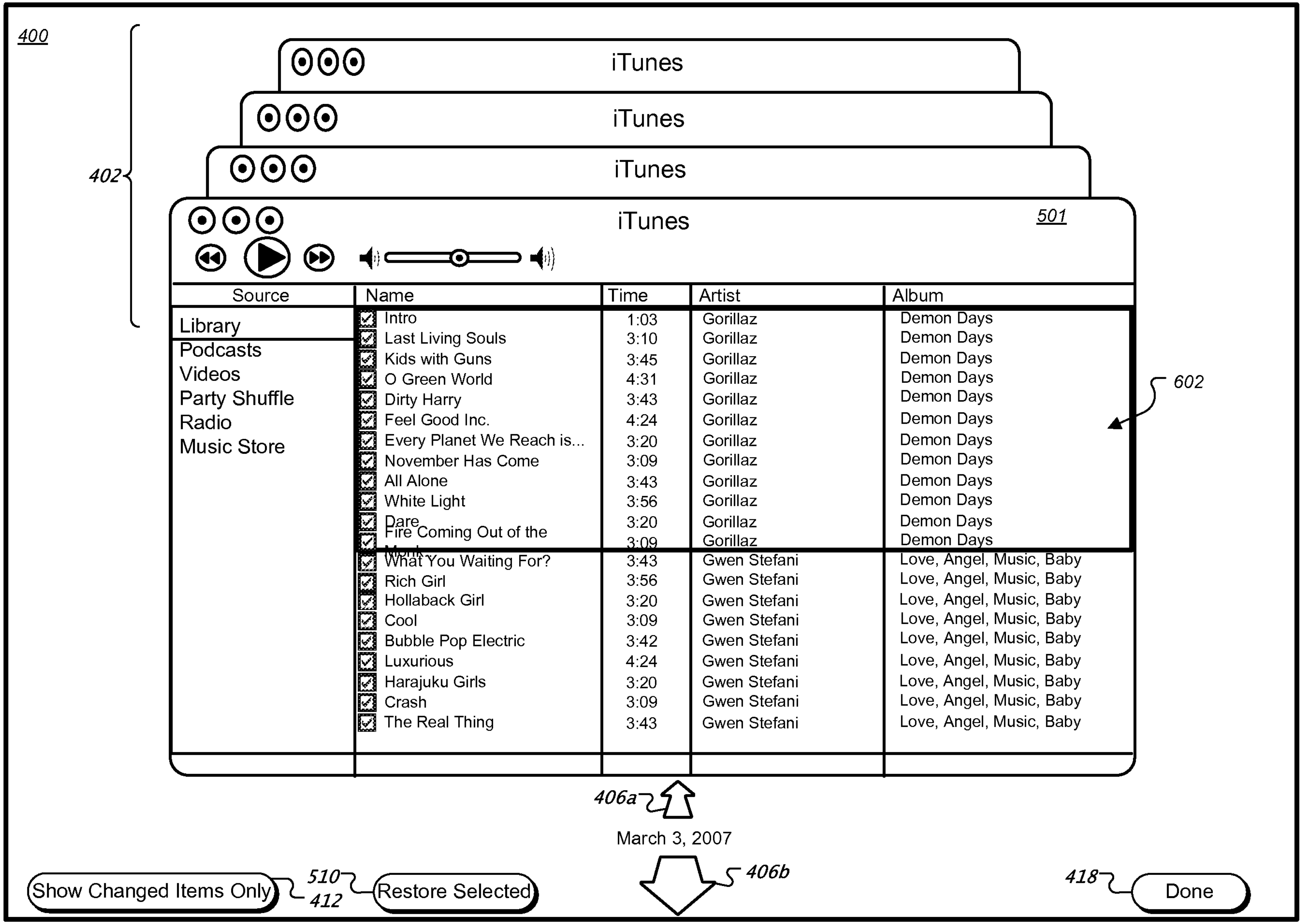
Name	Time	Artist	Album
☑ What You Waiting For?	3:43	Gwen Stefani	Love, Angel, Music, Baby
☑ Rich Girl	3:56	Gwen Stefani	Love, Angel, Music, Baby
☑ Hollaback Girl	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Cool	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ Bubble Pop Electric	3:42	Gwen Stefani	Love, Angel, Music, Baby
☑ Luxurious	4:24	Gwen Stefani	Love, Angel, Music, Baby
☑ Harajuku Girls	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Crash	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ The Real Thing	3:43	Gwen Stefani	Love, Angel, Music, Baby
☑ Serious	3:56	Gwen Stefani	Love, Angel, Music, Baby
☑ Danger Zone	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Long Way to Go	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ St. Louis Blues	3:42	Handy's Memphis Blues	The Blues
☑ Alle Vrouwen	4:24	Hans De Booij	Het Beste Uit Pop in Je ...
☑ Dick Turpin en de Koningin	3:20	Hans De Booij	Het Beste Uit Pop in Je ...
☑ God en Allah	3:09	Hans De Booij	Pop In Je Moerstall Volu...
☑ Geweld	3:43	Hans Teeuwen	Dat dan weer wel
☑ Vertrouwen	3:56	Hans Teeuwen	Dat dan weer wel
☑ Improvisatie II	3:20	Hans Teeuwen	Dat dan weer wel
☑ Fruit en Stront	3:09	Hans Teeuwen	Dat dan weer wel

FIG. 4



Name	Time	Artist	Album
☑ Intro	1:03	Gorillaz	Demon Days
☑ Last Living Souls	3:10	Gorillaz	Demon Days
☑ Kids with Guns	3:45	Gorillaz	Demon Days
☑ O Green World	4:31	Gorillaz	Demon Days
☑ Dirty Harry	3:43	Gorillaz	Demon Days
☑ Feel Good Inc.	4:24	Gorillaz	Demon Days
☑ Every Planet We Reach is...	3:20	Gorillaz	Demon Days
☑ November Has Come	3:09	Gorillaz	Demon Days
☑ All Alone	3:43	Gorillaz	Demon Days
☑ White Light	3:56	Gorillaz	Demon Days
☑ Dare	3:20	Gorillaz	Demon Days
☑ Fire Coming Out of the Monk	3:09	Gorillaz	Demon Days
☑ What You Waiting For?	3:43	Gwen Stefani	Love, Angel, Music, Baby
☑ Rich Girl	3:56	Gwen Stefani	Love, Angel, Music, Baby
☑ Hollaback Girl	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Cool	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ Bubble Pop Electric	3:42	Gwen Stefani	Love, Angel, Music, Baby
☑ Luxurious	4:24	Gwen Stefani	Love, Angel, Music, Baby
☑ Harajuku Girls	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Crash	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ The Real Thing	3:43	Gwen Stefani	Love, Angel, Music, Baby

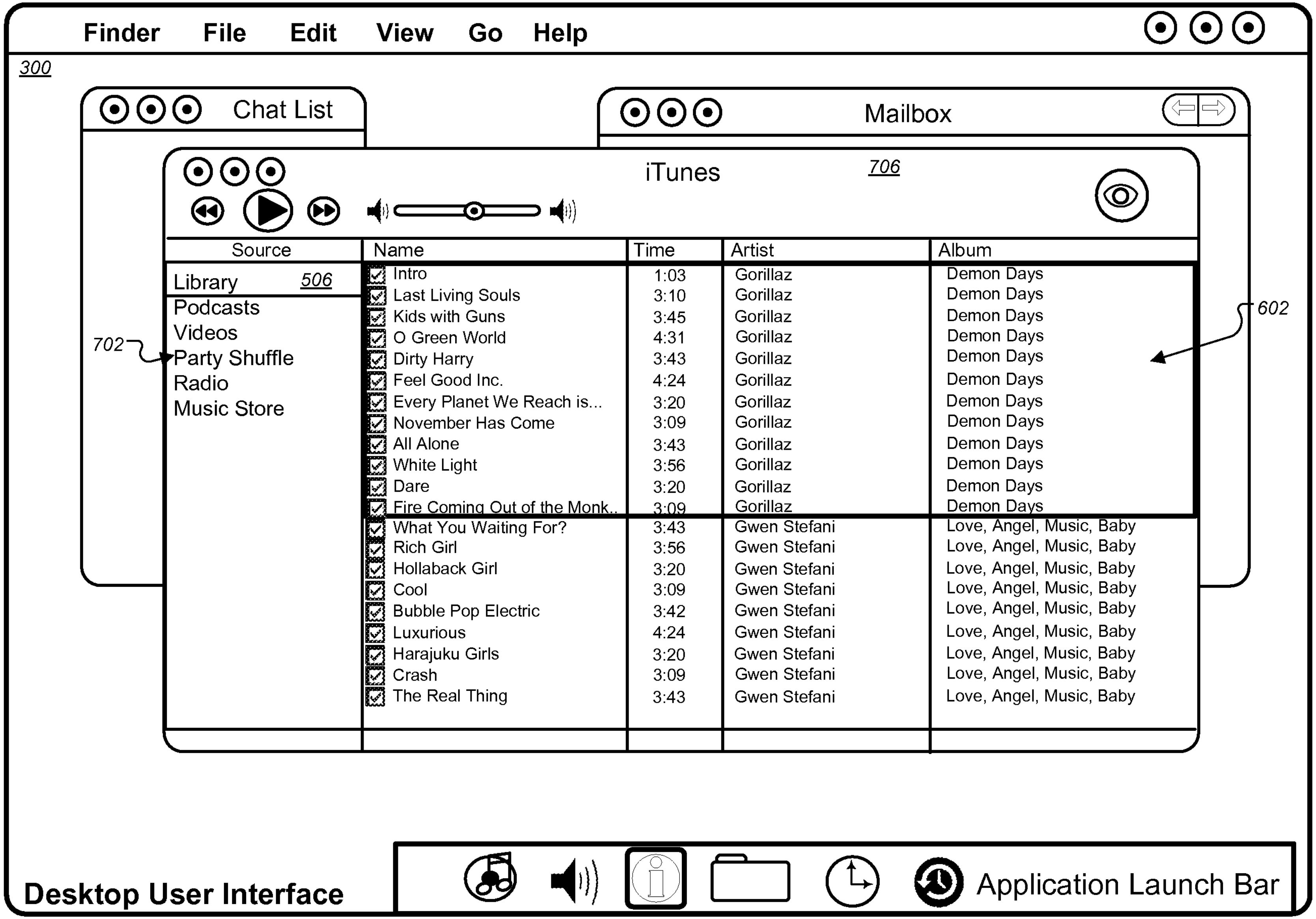
FIG. 5



Source		Name	Time	Artist	Album
Library	☑	Intro	1:03	Gorillaz	Demon Days
Podcasts	☑	Last Living Souls	3:10	Gorillaz	Demon Days
Videos	☑	Kids with Guns	3:45	Gorillaz	Demon Days
Party Shuffle	☑	O Green World	4:31	Gorillaz	Demon Days
Radio	☑	Dirty Harry	3:43	Gorillaz	Demon Days
Music Store	☑	Feel Good Inc.	4:24	Gorillaz	Demon Days
	☑	Every Planet We Reach is...	3:20	Gorillaz	Demon Days
	☑	November Has Come	3:09	Gorillaz	Demon Days
	☑	All Alone	3:43	Gorillaz	Demon Days
	☑	White Light	3:56	Gorillaz	Demon Days
	☑	Dare	3:20	Gorillaz	Demon Days
	☑	Fire Coming Out of the Monk	3:09	Gorillaz	Demon Days
	☑	What You Waiting For?	3:43	Gwen Stefani	Love, Angel, Music, Baby
	☑	Rich Girl	3:56	Gwen Stefani	Love, Angel, Music, Baby
	☑	Hollaback Girl	3:20	Gwen Stefani	Love, Angel, Music, Baby
	☑	Cool	3:09	Gwen Stefani	Love, Angel, Music, Baby
	☑	Bubble Pop Electric	3:42	Gwen Stefani	Love, Angel, Music, Baby
	☑	Luxurious	4:24	Gwen Stefani	Love, Angel, Music, Baby
	☑	Harajuku Girls	3:20	Gwen Stefani	Love, Angel, Music, Baby
	☑	Crash	3:09	Gwen Stefani	Love, Angel, Music, Baby
	☑	The Real Thing	3:43	Gwen Stefani	Love, Angel, Music, Baby

March 3, 2007

FIG. 6



Name	Time	Artist	Album
☑ Intro	1:03	Gorillaz	Demon Days
☑ Last Living Souls	3:10	Gorillaz	Demon Days
☑ Kids with Guns	3:45	Gorillaz	Demon Days
☑ O Green World	4:31	Gorillaz	Demon Days
☑ Dirty Harry	3:43	Gorillaz	Demon Days
☑ Feel Good Inc.	4:24	Gorillaz	Demon Days
☑ Every Planet We Reach is...	3:20	Gorillaz	Demon Days
☑ November Has Come	3:09	Gorillaz	Demon Days
☑ All Alone	3:43	Gorillaz	Demon Days
☑ White Light	3:56	Gorillaz	Demon Days
☑ Dare	3:20	Gorillaz	Demon Days
☑ Fire Coming Out of the Monk..	3:09	Gorillaz	Demon Days
☑ What You Waiting For?	3:43	Gwen Stefani	Love, Angel, Music, Baby
☑ Rich Girl	3:56	Gwen Stefani	Love, Angel, Music, Baby
☑ Hollaback Girl	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Cool	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ Bubble Pop Electric	3:42	Gwen Stefani	Love, Angel, Music, Baby
☑ Luxurious	4:24	Gwen Stefani	Love, Angel, Music, Baby
☑ Harajuku Girls	3:20	Gwen Stefani	Love, Angel, Music, Baby
☑ Crash	3:09	Gwen Stefani	Love, Angel, Music, Baby
☑ The Real Thing	3:43	Gwen Stefani	Love, Angel, Music, Baby

FIG. 7

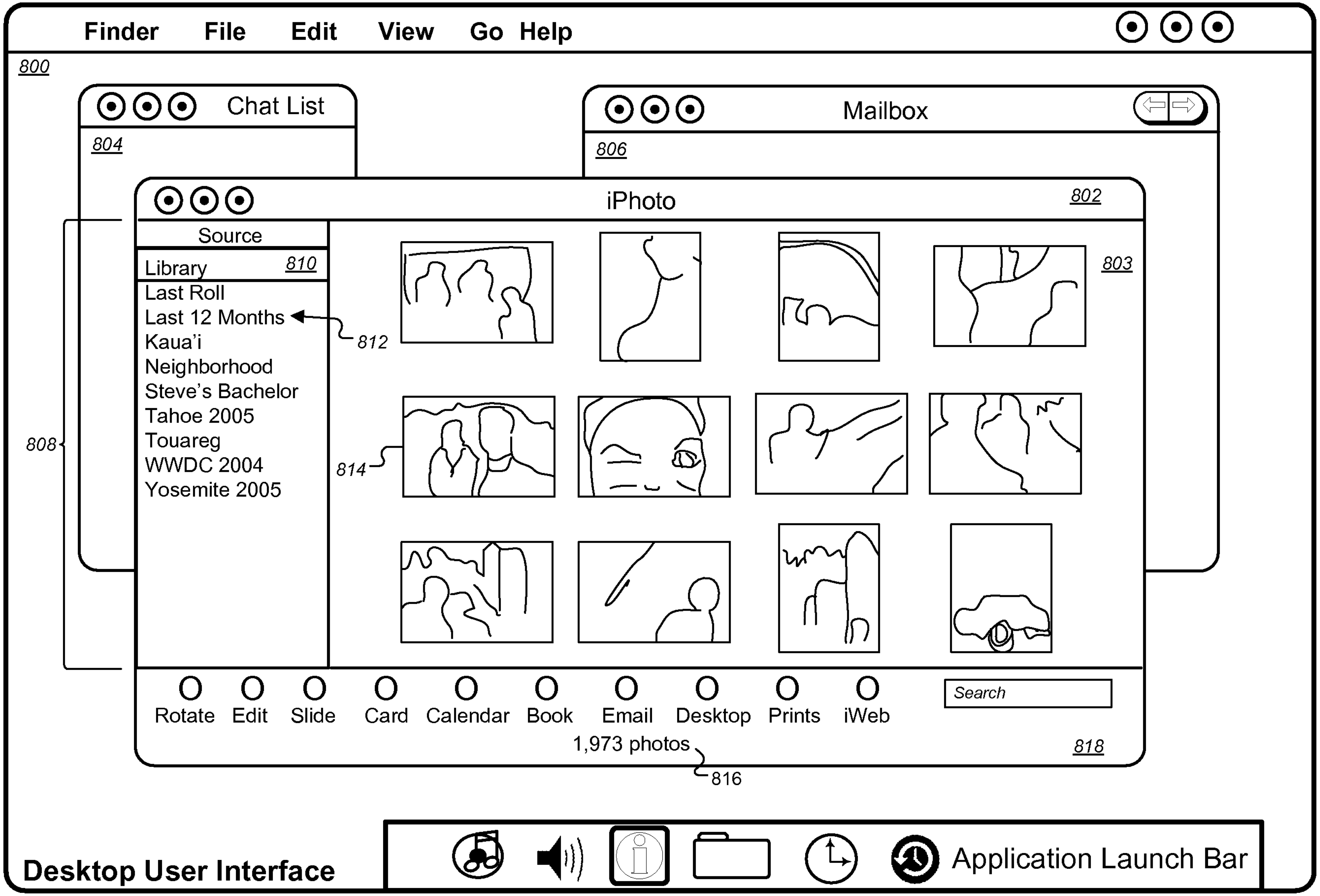

Finder File Edit View Go Help
800
Chat List
804
Mailbox
806
iPhoto
802
Source
Library
810
Last Roll
Last 12 Months
812
Kaua'i
Neighborhood
Steve's Bachelor
Tahoe 2005
Touareg
WWDC 2004
Yosemite 2005
808
814
803
Rotate
Edit
Slide
Card
Calendar
Book
Email
Desktop
Prints
iWeb
Search
1,973 photos
816
818
Desktop User Interface
Application Launch Bar

FIG. 8

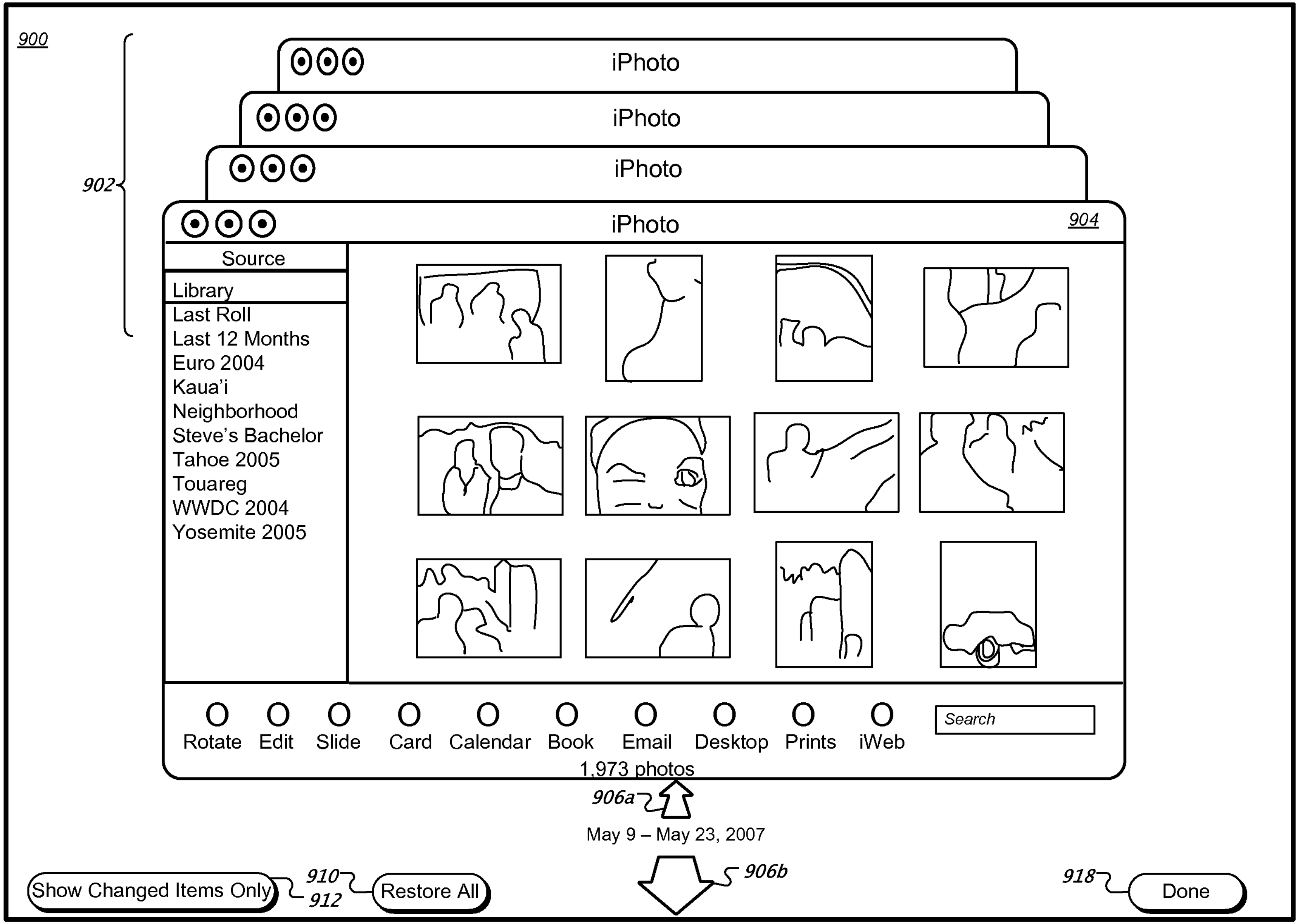

900
902
iPhoto
iPhoto
iPhoto
iPhoto
904
Source
Library
Last Roll
Last 12 Months
Euro 2004
Kaua'i
Neighborhood
Steve's Bachelor
Tahoe 2005
Touareg
WWDC 2004
Yosemite 2005
Rotate
Edit
Slide
Card
Calendar
Book
Email
Desktop
Prints
iWeb
Search
1,973 photos
906a
May 9 – May 23, 2007
906b
Show Changed Items Only
910
912
Restore All
918
Done

FIG. 9

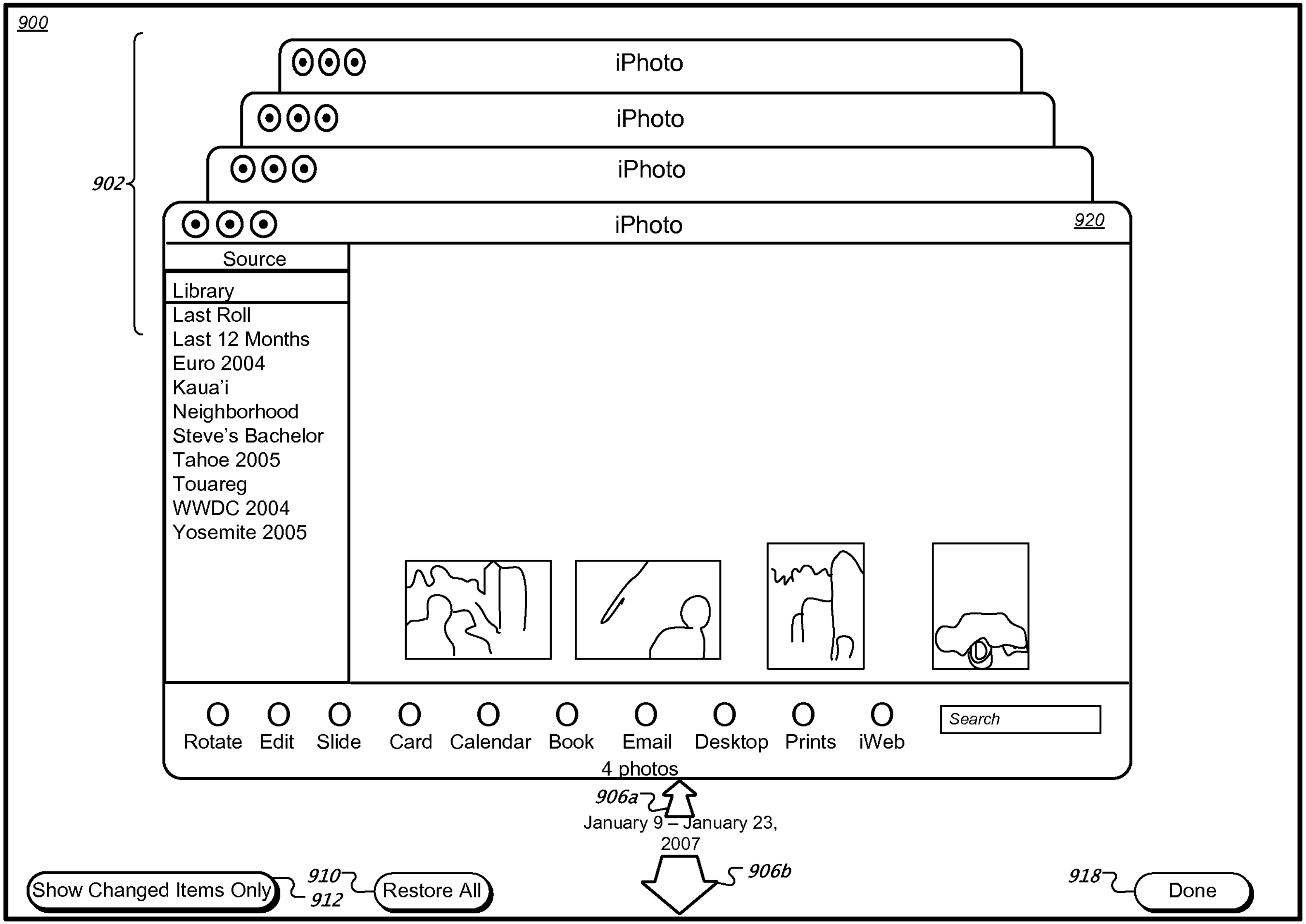

900
902
iPhoto
iPhoto
iPhoto
iPhoto
920
Source
Library
Last Roll
Last 12 Months
Euro 2004
Kaua'i
Neighborhood
Steve's Bachelor
Tahoe 2005
Touareg
WWDC 2004
Yosemite 2005
Rotate
Edit
Slide
Card
Calendar
Book
Email
Desktop
Prints
iWeb
Search
4 photos
906a
January 9 – January 23, 2007
906b
Show Changed Items Only
910
912
Restore All
918
Done

FIG. 10

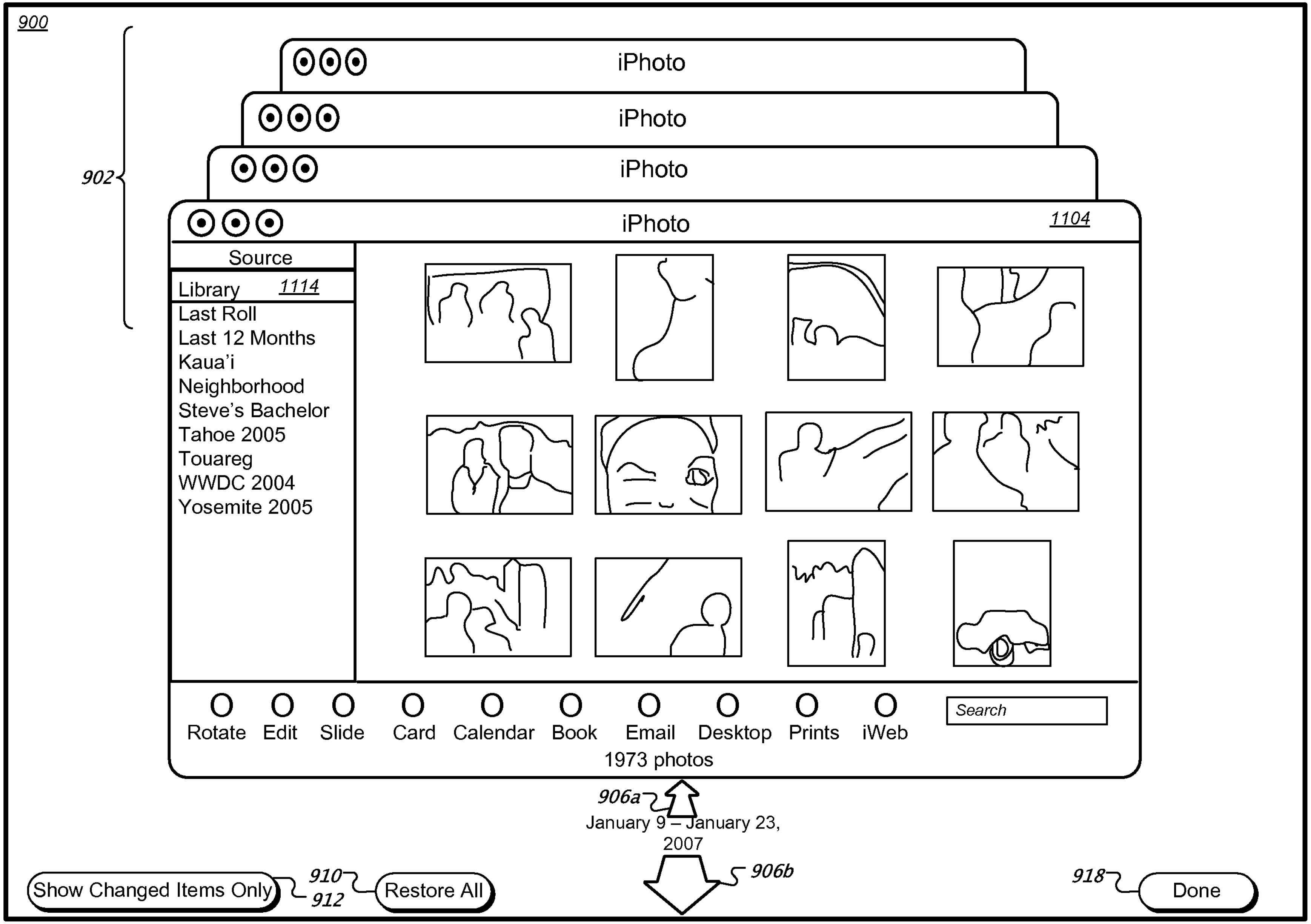

900
902
iPhoto
iPhoto
iPhoto
iPhoto
1104
Source
Library
1114
Last Roll
Last 12 Months
Kaua'i
Neighborhood
Steve's Bachelor
Tahoe 2005
Touareg
WWDC 2004
Yosemite 2005
Rotate
Edit
Slide
Card
Calendar
Book
Email
Desktop
Prints
iWeb
Search
1973 photos
906a
January 9 – January 23, 2007
906b
Show Changed Items Only
910
912
Restore All
918
Done

FIG. 11

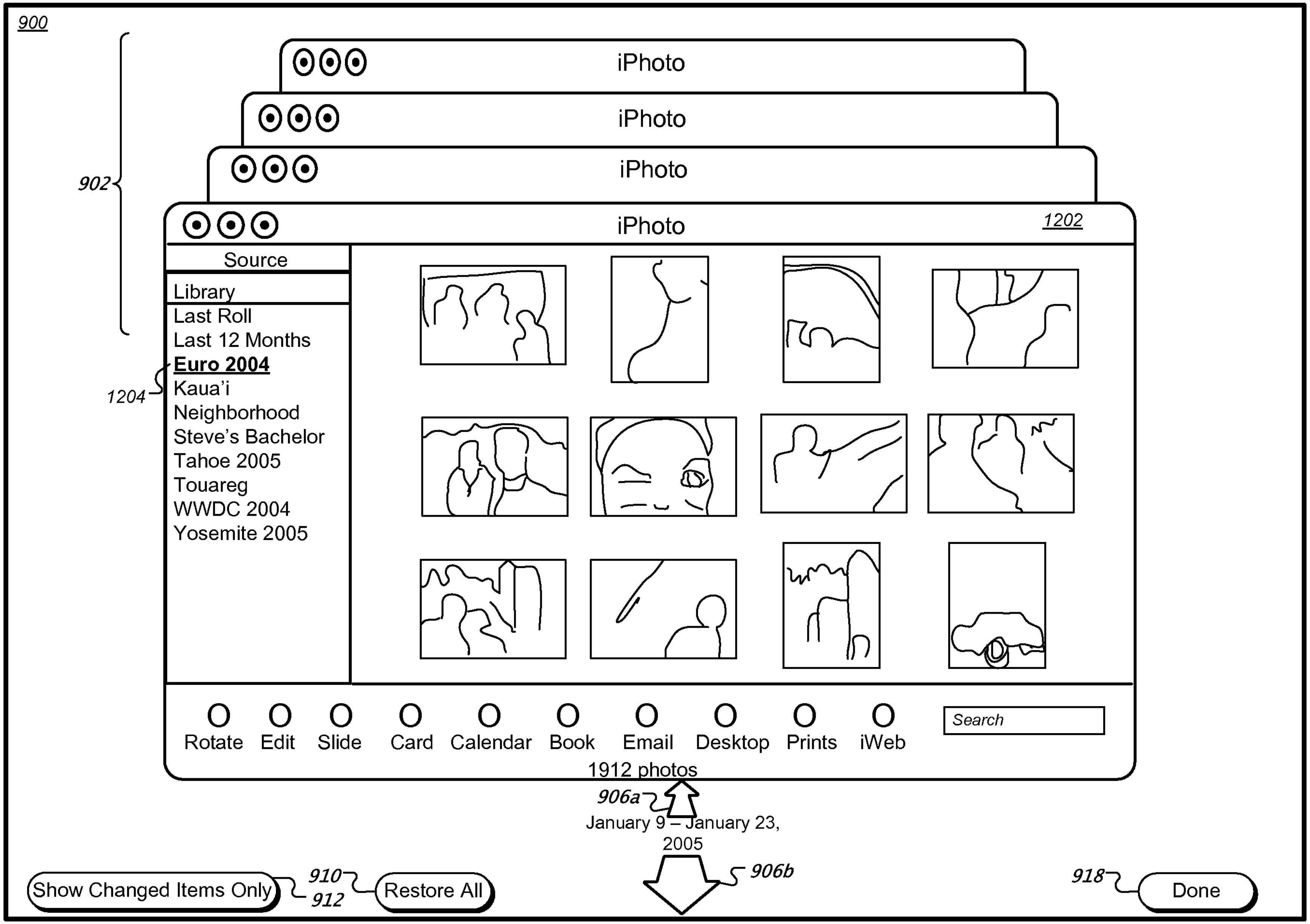

900
902
iPhoto
iPhoto
iPhoto
iPhoto
1202
Source
Library
Last Roll
Last 12 Months
Euro 2004
Kaua'i
Neighborhood
Steve's Bachelor
Tahoe 2005
Touareg
WWDC 2004
Yosemite 2005
1204
Rotate
Edit
Slide
Card
Calendar
Book
Email
Desktop
Prints
iWeb
Search
1912 photos
906a
January 9 – January 23, 2005
906b
Show Changed Items Only
910
912
Restore All
918
Done

FIG. 12

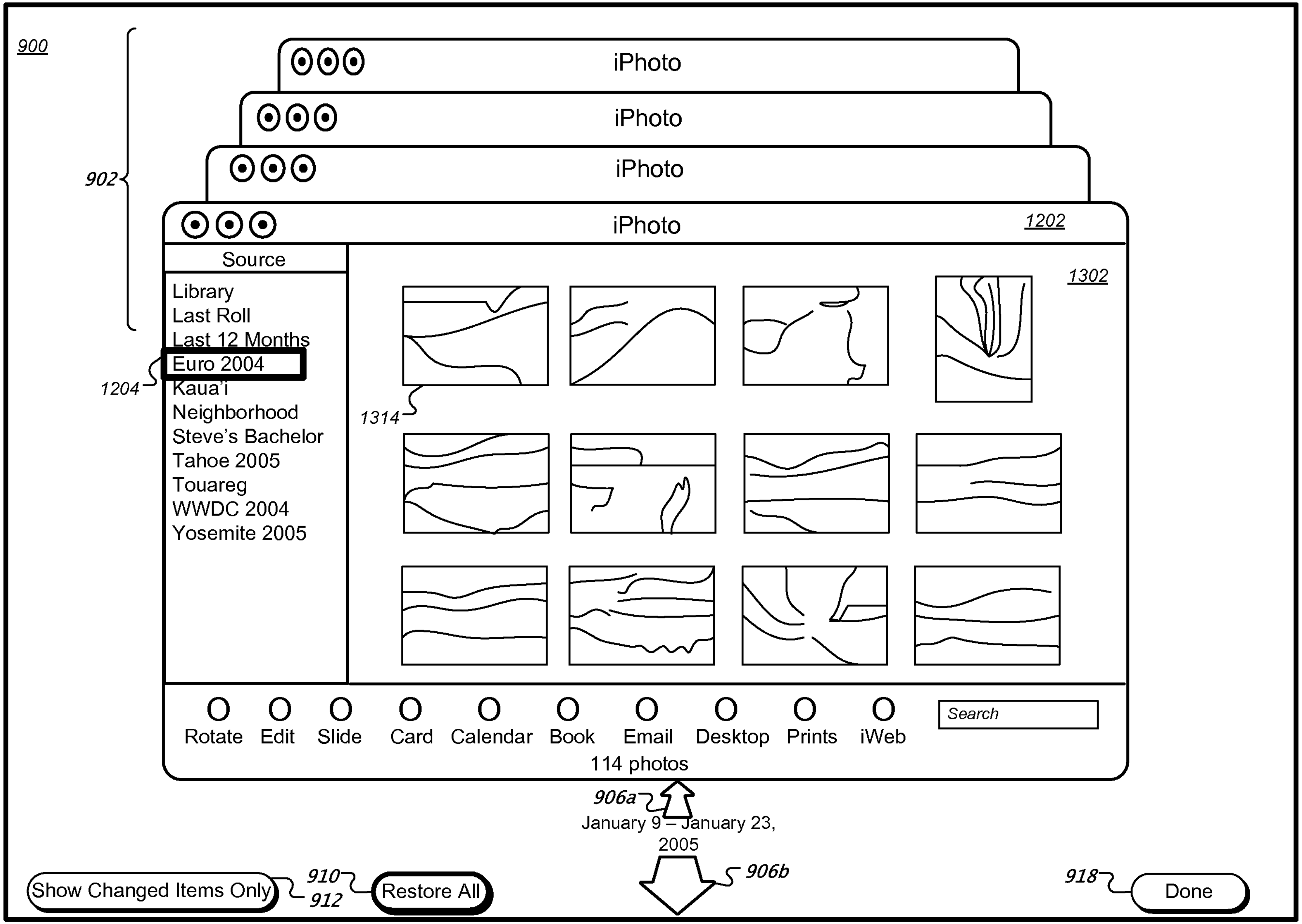

900
902
iPhoto
iPhoto
iPhoto
iPhoto
1202
Source
Library
Last Roll
Last 12 Months
Euro 2004
Kaua'i
Neighborhood
Steve's Bachelor
Tahoe 2005
Touareg
WWDC 2004
Yosemite 2005
1204
1302
1314
Rotate
Edit
Slide
Card
Calendar
Book
Email
Desktop
Prints
iWeb
Search
114 photos
906a
January 9 – January 23, 2005
906b
Show Changed Items Only
910
912
Restore All
918
Done

FIG. 13

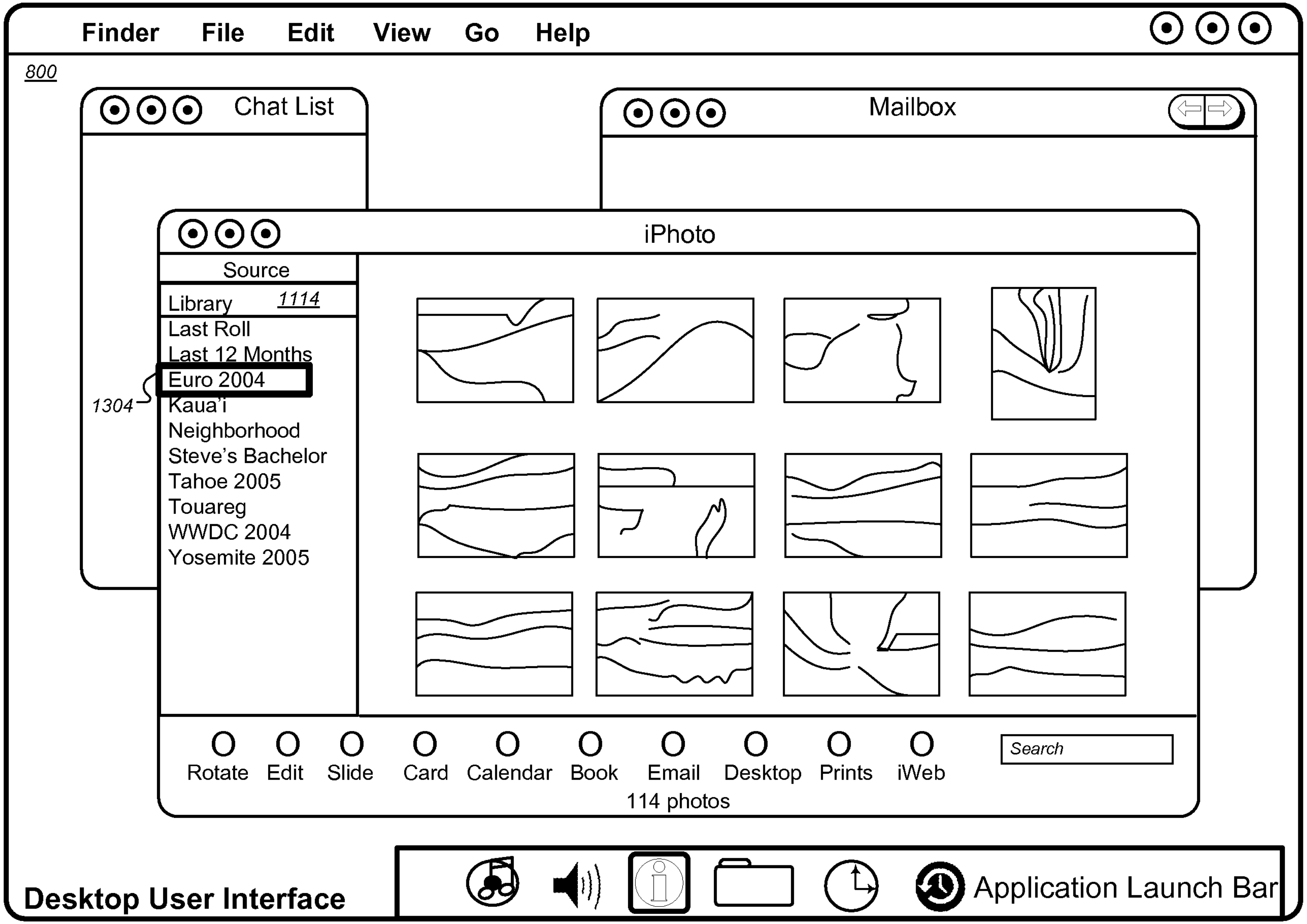

Finder
File
Edit
View
Go
Help
800
Chat List
Mailbox
iPhoto
Source
Library
1114
Last Roll
Last 12 Months
Euro 2004
1304
Kaua'i
Neighborhood
Steve's Bachelor
Tahoe 2005
Touareg
WWDC 2004
Yosemite 2005
Rotate
Edit
Slide
Card
Calendar
Book
Email
Desktop
Prints
iWeb
Search
114 photos
Desktop User Interface
Application Launch Bar

FIG. 14