SYSTEM AND METHOD FOR USER MODIFICATION OF METADATA IN A SHELL BROWSER

TECHNICAL FIELD

The present invention relates generally to the field of computer software. More particularly, the invention relates to a system and method for providing an improved user experience within a shell or file system browser so that users can more readily identify an item based on the metadata associated with that item.

BACKGROUND OF THE INVENTION

The need to readily identify items stored in a computing environment such as a personal computer (PC) is dramatically increasing as more individuals utilize computers in their daily routines and as the type of stored information varies between pictures, music, documents, etc. Documents and media are typically stored on computers in a hierarchical fashion and are organized with files of information or media stored within folders. File system browsers enable users to navigate through the file system and locate and open files and folders. For example, Microsoft Corporation's WINDOWS® EXPLORER™ is an operating system utility which enables users to browse the file system.

Many users find it difficult to correctly identify a file based on the information currently available in conventional file system browsers. Of course the contents of a file can be verified by opening it with an application program, but this method of browsing files is extremely inefficient. The ability to view metadata about a file within a file system browser can greatly assist a user in identifying a particular file without having to open it. In Microsoft Corporation's WINDOWS® 9X operating systems, for example, a user can view object metadata by accessing the property sheet for a particular object. A property sheet presents the user with a list of the attributes or settings of an object in the form of a tabbed, index-card-like selection of property pages, each of which features standard dialog-style controls for customizing parameters. However, using the property sheet to locate an item can be slow and cumbersome, and some users find it difficult to locate the relevant metadata in a property sheet. Similarly, the use of infotips to locate an item can be slow and cumbersome because a user

must hover the mouse over each file in order to view the limited metadata displayed in an infotip.

Conventional file system browsers do not allow users to enter and edit metadata relating to files and folders, which would significantly enhance a user's ability to later locate a file. To date, the ability of users to enter and edit metadata has been limited to special purpose software programs. For example, media players for electronic music files present users with the ability to edit metadata associated with music albums and artists. Another example of such programs includes application programs for electronic picture files. However, the utility of media players and other such programs is limited to the particular type of file supported by the program, as opposed to a general purpose file system browser which supports multiple file types.

Microsoft Corporation's WINDOWS® XP operating system includes an image browser for use in the My Pictures folder. The My Pictures folder is endowed with special features which enable users to view pictures as photos, not just as document icons. My Picture's image browsing features include the ability to view thumbnail-size and large versions of photos, rotate photos that are sideways, and create a slide show. A user can also view a photo's details, such as its dimensions, the date and time it was taken, and the name of the camera that took it. The preview control area in the My Picture's folder contains an enlarged preview image of a user-selected image, iterator buttons to assist a user in iterating through a series of pictures and controls for rotating pictures in a clockwise or counterclockwise direction. While the image browsing features in WINDOWS® XP have advanced the state of the art by alleviating the need to invoke an application program to view and manipulate pictures, users still cannot enter and edit metadata associated with the pictures.

Accordingly, there is a need for an improved user experience within a shell or file system browser which enables users to readily locate an item based on the metadata associated with that item. There is also a need for a system and method which allow users to enter and edit metadata associated with items of various types within a shell browser without the need to invoke an application program.

SUMMARY OF THE INVENTION

The present invention meets the above needs and overcomes one or more deficiencies in the prior art by providing a system and method for user modification of metadata in a shell browser. In one aspect of the present invention, a shell browser is provided which includes a window and an edit control. The window displays a group of items and also displays metadata values associated with one or more of the displayed items. The edit control permits user modification of at least a portion of the metadata values displayed in the window.

Another aspect of the present invention is a graphical user interface embodied on a computer-readable medium and executable on a computer. The graphical user interface includes a first screen area which displays a set of items in a shell browser and a second screen area which displays metadata associated with one or more of the displayed items. The graphical user interface also presents the user with means within the shell browser for modifying the displayed metadata.

In a further aspect of the present invention, computer-implemented methods are provided for enabling a user to modify metadata within a shell browser. One such method includes displaying a plurality of items, receiving a first input from the user representing a selection of at least one displayed item, displaying metadata associated with the selected item(s) and providing an edit control for user modification of the displayed metadata. Another such method includes displaying a welcome pane and metadata associated with the welcome pane and providing an edit control for user modification of the displayed metadata.

Still another aspect of the present invention is a data structure containing metadata associated with one or more items displayed in a shell browser. The data structure, which is stored on one or more computer-readable media, includes a field containing user modifiable metadata associated with the one or more displayed items, and the user modifiable metadata contained in the data structure is also displayed in the shell browser.

BRIEF DESCRIPTION OF THE DRAWING

The present invention is described in detail below with reference to the attached drawing figures, wherein:

FIG. 1 is a block diagram of a computing-system environment suitable for use in implementing the present invention;

FIG. 2A is a block diagram of an exemplary graphical user interface for a shell browser having an edit control in accordance with an embodiment of the present invention;

FIG. 2B is a block diagram of an exemplary graphical user interface for a shell browser having one or more edit controls in accordance with an embodiment of the present invention;

FIG. 3 is a schematic diagram of a welcome pane in a shell browser;

FIG. 4 is a schematic diagram of a selected pane in a shell browser;

FIG. 5 is a schematic diagram of the selected pane of FIG. 4 including a context menu enabling a user to modify metadata in a shell browser in accordance with an embodiment of the present invention;

FIG. 6 is a flow diagram illustrating a method for enabling a user to modify metadata displayed in a welcome pane within a shell browser in accordance with an embodiment of the present invention;

FIG. 7 is a flow diagram illustrating a method for enabling a user to modify metadata displayed in a selected pane within a shell browser in accordance with an embodiment of the present invention; and

FIG. 8 is a block diagram of a data structure containing user modifiable metadata associated with an item displayed in a shell browser.

DETAILED DESCRIPTION OF THE INVENTION

The present invention relates to a system and method for providing an improved user experience within a shell browser so that users can more readily identify an item based on the metadata associated with that item. An exemplary operating environment for the present invention is described below.

Referring to the drawings in general and initially to FIG. 1 in particular, wherein like reference numerals identify like components in the various figures, an exemplary operating environment for implementing the present invention is shown and designated generally as operating environment 100. The computing system environment 100 is only one example of a suitable

computing environment and is not intended to suggest any limitation as to the scope of use or functionality of the invention. Neither should the computing environment 100 be interpreted as having any dependency or requirement relating to any one or combination of components illustrated in the exemplary operating environment 100.

The invention may be described in the general context of computer-executable instructions, such as program modules, being executed by a computer. Generally, program modules include routines, programs, objects, components, data structures, etc., that perform particular tasks or implement particular abstract data types. Moreover, those skilled in the art will appreciate that the invention may be practiced with a variety of computer-system configurations, including hand-held devices, multiprocessor systems, microprocessor-based or programmable-consumer electronics, minicomputers, mainframe computers, and the like. The invention may also be practiced in distributed-computing environments where tasks are performed by remote-processing devices that are linked through a communications network. In a distributed-computing environment, program modules may be located in both local and remote computer-storage media including memory storage devices.

With reference to FIG. 1, an exemplary system 100 for implementing the invention includes a general purpose computing device in the form of a computer 110 including a processing unit 120, a system memory 130, and a system bus 121 that couples various system components including the system memory 130 to the processing unit 120.

Computer 110 typically includes a variety of computer-readable media. By way of example, and not limitation, computer-readable media may comprise computer-storage media and communication media. Examples of computer-storage media include, but are not limited to, Random Access Memory (RAM); Read-Only Memory (ROM); Electronically Erasable Programmable Read-Only Memory (EEPROM); flash memory or other memory technology; CD-ROM, digital versatile discs (DVD) or other optical or holographic disc storage; magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices; or any other medium that can be used to store desired information and be accessed by computer 110. The system memory 130 includes

computer-storage media in the form of volatile and/or nonvolatile memory such as ROM 131 and RAM 132. A Basic Input/Output System 133 (BIOS), containing the basic routines that help to transfer information between elements within computer 110 (such as during start-up) is typically stored in ROM 131. RAM 132 typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by processing unit 120. By way of example, and not limitation, FIG. 1 illustrates operating system 134, application programs 135, other program modules 136, and program data 137.

The computer 110 may also include other removable/nonremovable, volatile/nonvolatile computer-storage media. By way of example only, FIG. 1 illustrates a hard disk drive 141 that reads from or writes to nonremovable, nonvolatile magnetic media, a magnetic disk drive 151 that reads from or writes to a removable, nonvolatile magnetic disk 152, and an optical disc drive 155 that reads from or writes to a removable, nonvolatile optical disc 156 such as a CD-ROM or other optical media. Other removable/nonremovable, volatile/nonvolatile computer-storage media that can be used in the exemplary operating environment include, but are not limited to, magnetic tape cassettes, flash memory units, digital versatile disks, digital video tape, solid state RAM, solid state ROM, and the like. The hard disk drive 141 is typically connected to the system bus 121 through a nonremovable memory interface such as interface 140. Magnetic disk drive 151 and optical disc drive 155 are typically connected to the system bus 121 by a removable memory interface, such as interface 150.

The drives and their associated computer-storage media discussed above and illustrated in FIG. 1 provide storage of computer-readable instructions, data structures, program modules and other data for computer 110. For example, hard disk drive 141 is illustrated as storing operating system 144, application programs 145, other program modules 146, and program data 147. Note that these components can either be the same as or different from operating system 134, application programs 135, other program modules 136, and program data 137. Typically, the operating system, application programs and the like that are stored in RAM are portions of the corresponding systems, programs, or data read from hard disk drive 141, the portions varying in size and scope depending on the

functions desired. Operating system 144, application programs 145, other program modules 146, and program data 147 are given different numbers here to illustrate that, at a minimum, they can be different copies. A user may enter commands and information into the computer 110 through input devices such as a keyboard 162; pointing device 161, commonly referred to as a mouse, trackball or touch pad; a wireless-input-reception component 163; or a wireless source such as a remote control. Other input devices (not shown) may include a microphone, joystick, game pad, satellite dish, scanner, or the like. These and other input devices are often connected to the processing unit 120 through a user-input interface 160 that is coupled to the system bus 121 but may be connected by other interface and bus structures, such as a parallel port, game port, IEEE 1394 port, or a universal serial bus (USB), or infrared (IR) bus.

A display device 191 is also connected to the system bus 121 via an interface, such as a video interface 190. Display device 191 can be any device to display the output of computer 110 not limited to a monitor, an LCD screen, a Thin Film Transistor (TFT) screen, a flat-panel display, a conventional television, or screen projector. In addition to the display device 191, computers may also include other peripheral output devices such as speakers 197 and printer 196, which may be connected through an output peripheral interface 195.

The computer 110 in the present invention will operate in a networked environment using logical connections to one or more remote computers, such as a remote computer 180. The remote computer 180 may be a personal computer, and typically includes many or all of the elements described above relative to the computer 110, although only a memory storage device 181 has been illustrated in FIG. 1. The logical connections depicted in FIG. 1 include a local-area network (LAN) 171 and a wide-area network (WAN) 173 but may also include other networks, such as connections to a metropolitan-area network (MAN), intranet, or the Internet.

When used in a LAN networking environment, the computer 110 is connected to the LAN 171 through a network interface or adapter 170. When used in a WAN networking environment, the computer 110 typically includes a modem 172 or other means for establishing communications over the WAN 173, such as the Internet. The modem 172, which may be internal or external, may be

connected to the system bus 121 via the network interface 170, or other appropriate mechanism. Modem 172 could be a cable modem, DSL modem, or other broadband device. In a networked environment, program modules depicted relative to the computer 110, or portions thereof, may be stored in the remote memory storage device. By way of example, and not limitation, FIG. 1 illustrates remote application programs 185 as residing on memory device 181. It will be appreciated that the network connections shown are exemplary and other means of establishing a communications link between the computers may be used.

Although many other internal components of the computer 110 are not shown, those of ordinary skill in the art will appreciate that such components and the interconnections are well-known. For example, including various expansion cards such as television-tuner cards and network-interface cards within a computer 110 is conventional. Accordingly, additional details concerning the internal construction of the computer 110 need not be disclosed in connection with the present invention.

When the computer 110 is turned on or reset, the BIOS 133, which is stored in ROM 131, instructs the processing unit 120 to load the operating system, or necessary portion thereof, from the hard disk drive 141 into the RAM 132. Once the copied portion of the operating system, designated as operating system 144, is loaded into RAM 132, the processing unit 120 executes the operating-system code and causes the visual elements associated with the user interface of the operating system 134 to be displayed on the display device 191. Typically, when an application program 145 is opened by a user, the program code and relevant data are read from the hard disk drive 141 and the necessary portions are copied into RAM 132, the copied portion represented herein by reference numeral 135.

As previously mentioned, the present invention may be described in the general context of computer-useable instructions. Computer-useable instructions include functions, procedures, schemas, routines, code segments, and modules useable by one or more computers or other devices. The computer-useable instructions form an interface to allow a computer to react according to a source of input. The instructions cooperate with other code segments to initiate a

variety of tasks in response to data received in conjunction with the source of the received data.

Turning to FIG. 2A, a window 200 represents a screen-size display area for a graphical user interface of a shell browser. The window 200 contains a preview pane area 202 and a view area 204. The preview pane 202 may include a preview control 206, a user interface (UI) or edit control 208, and a task control 210. Typically, the preview control 206 will provide the user with an image or other visual display of the item being previewed (e.g., a selected file). The preview control 206 may also present the user with controls such as iterator buttons which allow the user to shift the focus from one item to the next by clicking a mouse button. Metadata corresponding to one or more items and/or metadata corresponding to the item container may be displayed in a variety of locations within the window 200. For example, the edit control and metadata may be co-located within edit control area 208 so that the edit control area not only includes a display of key properties of the previewed item but also presents the user with the option of making edits to the metadata. The task control 210 contains tasks relevant to the namespace and/or the selection.

For purposes of the present invention, the terms "metadata" and "user modifiable metadata" exclude the shell item name. The term "shell item name" refers to the property which is used for purposes of sorting and displaying the item within the shell browser. As mentioned above, one unique aspect of the present invention is the ability of a user to edit metadata within a shell browser.

Those skilled in the art will appreciate that the present invention contemplates the presence of optional features within the window 200. For example, the preview control 206 and the task control 210 are not essential features for purposes of the present invention. Moreover, other non-essential features which are not shown in FIG. 2A, such as a toolbar which includes iterator buttons or a show/hide button so the user can open/close the preview pane, are also within the scope of the present invention. Nevertheless, these and other optional features may assist the user in readily locating a particular item in the shell browser.

The view area 204 provides a listview of one or more items 212, such as file system files or folders. The term "listview" refers to an enumeration

or list of items within a container. The terms "item" and "shell item" are used interchangeably herein to refer to files, folders and other such containers, and other non-file objects which can be represented in a listview. Examples of non-file objects may include, but would not be limited to, contacts, favorites and email messages. The terms "shell browser" and "file system browser" are used interchangeably herein to refer to a browser which allows a user to navigate through various namespaces including files and other non-file items.

Those skilled in the art will appreciate that the present invention contemplates many possible designs and layouts for the window 200. For example, the preview pane 202 is shown above the view area 204 in FIG. 2A. However, other layouts, such as placing the preview pane 202 and the view area 204 side-by-side, are clearly within the scope of the present invention. The location of the edit control 208 is also independent of the location of the displayed metadata and independent of the location of any other controls. There are also many possible view types for the items depicted in listview area 204, such as details, slide show, filmstrip, thumbnail, tiles, icons, etc.

FIG. 2B is similar to FIG. 2A, except that the view area 204 is replaced by a view area 214 which displays the items 212 in details mode. As is typical for shell items displayed in details mode, the items 212 are aligned in a column at the left-hand side of view area 214, and one or more column headings 216 form the top row of a set of columns containing metadata 218 relating to the corresponding item located in the same row. Importantly, the present invention contemplates the ability of a user to explicitly change a metadata value to another value through instantiation of one or more edit controls 208 anywhere within the window 200. For example, an edit control may be provided within the preview pane 202 and/or within the view area 214. For example, an edit control which is not initially visible to a user may be provided within the view area 214. Such a control can be instantiated, for example, when the user hovers over the metadata 218 and then clicks on it to enter an editing mode.

Referring next to FIG. 3, a schematic illustration is provided of a welcome pane 300 in a shell browser. A welcome pane is sometimes referred to as a "null select" pane because it represents a namespace or container as opposed to a selection. If the user has not yet made a selection, a preview pane 302

displays metadata 304 and key tasks relating to the folder or shell library. If desired, the tasks may be separated into premiered tasks 306 and other relevant tasks 308. The welcome pane 300 also includes a view area 310, in which multiple files or other items 312 may be viewed. The welcome pane metadata 304 may include information such as properties of the container (e.g., MyPictures), in which case the metadata display may be static. Alternatively, the welcome pane metadata 304 may include information such as a sampling of metadata from each of the items within the container, in which case the metadata display may change frequently. For example, the metadata display may be limited to properties of one item at a time by cycling from one item to the next every 30 seconds.

FIG. 4 is a schematic illustration of a selected pane 400 in a shell browser. As opposed to a welcome pane, a selected pane represents a selection by the user. If the user selects a container or folder, the selected pane need not be identical to the welcome pane for that container or folder. In FIG. 4, the selected pane 400 includes a preview pane 402 which contains a preview control 404, a metadata display 406 and a tasks display 408. Like the welcome pane 300 (in FIG. 3), the selected pane 400 also includes a view area 410, in which multiple files or other items 412 may be viewed. In FIG. 4, however, the user has selected one of the files. Consequently, the preview control 404 displays a preview image of the selected file, the metadata display 406 shows properties of the selected file, and the tasks display 408 provides a menu of relevant tasks for operating on the selected file.

FIG. 5 is a schematic representation of the selected pane of FIG. 4 but which also includes a context menu 500 to enable a user to modify metadata in a shell browser in accordance with an embodiment of the present invention. The context menu 500 in FIG. 5 presents the user with several options for changing the selected metadata. The generic text shown in the menu 500 is of course merely one example of the type of options which may be presented to a user for editing the displayed metadata. A context menu can be provided in any window, including a welcome pane, to improve the user experience. As those skilled in the art will appreciate, any number and variety of context menus could be supported by the present invention. For purposes of the present invention, one

means for enabling user modifications to displayed metadata within a shell browser is to provide a context menu such as editable metadata context menu 500. A user may summon the context menu, for example, by clicking on the corresponding text or object in the preview pane.

Those skilled in the art will appreciate that the present invention contemplates means other than context menus for enabling user modifications to displayed metadata within a shell browser. Another such means for is for the user to click on the metadata to enter an editing mode. By contrast, a user could enter an editing mode by hovering over the relevant text or object in the preview pane. Numerous alternative means are available and within the scope of the present invention.

FIG. 6 is a flow diagram illustrating a method 600 for enabling a user to modify metadata displayed in a welcome pane within a shell browser in accordance with an embodiment of the present invention. The method 600 includes displaying a welcome pane and metadata associated with the welcome pane at 602. Then, at 604, the method provides a control for user modification of the displayed metadata. When the user manipulates the control to modify the displayed metadata at 606, the method then associates the modified metadata with the welcome pane at 608 so that the modified metadata will be displayed the next time the welcome pane is displayed.

FIG. 7 is a flow diagram illustrating a method 700 for enabling a user to modify metadata displayed in a selected pane within a shell browser in accordance with an embodiment of the present invention. At 702, the method 700 first displays a number of items, such as items in a welcome pane or items in a selected container. When the user selects one or more of the items at 704, the method displays metadata associated with the selected item(s) at 706. At 708, the method provides a control for user modification of the displayed metadata. When the user manipulates the control to modify the displayed metadata at 710, the method then associates the modified metadata with the selected item(s) at 712 so that the modified metadata will be displayed the next time the selected item(s) is/are displayed.

In the event a user selects multiple items at 704, the displayed metadata may include intersecting properties of the selected items, a union of

properties, or perhaps a new property relevant to the selected items. Alternatively, the displayed metadata may include a rotating sample of metadata from each of the selected items (e.g., cycling from one selected item's metadata to the next selected item's metadata every 30 seconds). It is possible for the display of metadata which would result from a selection of all of the items to be identical to the display of metadata which would result from a null select.

FIG. 8 is a block diagram of a data structure 800 containing user modifiable metadata associated with an item displayed in a shell browser. The data structure 800 includes a title field 802 which indicates the name of the item. In the case of non-file items, the title field 802 may contain the name of whatever property is used to alphabetize that item in a listview. The data structure 800 includes a user editable properties field 804 containing one or more properties associated with the displayed item, wherein the user editable properties are displayed in the shell browser with the displayed item. The data structure 800 may optionally include a read-only properties field 806 which contains any read-only properties associated with the displayed item and worthy of display in the shell browser. Given the size constraints of the metadata display in the shell browser, the number of properties in fields 804 and 806 may be limited. Consequently, the data structure 800 may optionally include an all properties field 808, which contains a link or pointer to a location (e.g., a property page) which contains all of the properties or metadata associated with the displayed item. Of course, the all properties field 808 would not be necessary in the event that fields 804 and 806 contain all of the properties associated with the displayed item. The data structure 800 is stored on one or more computer-readable media, such as in a file system or shell, to provide rich storage views, and thus an improved user experience, within the shell browser.

The present invention enables a number of scenarios which were not possible with conventional shell browsers. As a first example, a student can manage her projects using the preview pane. When she obtains new documents as part of a project she is working on, she can select those documents in her document library and enter the name of the document author and the name of the project into keyword fields using the edit control. Now the new documents will show up in her favorite view: "Documents Grouped by Keyword and Listed by

Author." A second example of a new scenario enabled by the present invention involves an employee looking for materials for an upcoming ad campaign. As he browses through his employer's stock collection of photos using the shell browser, he selects a couple of pictures and, from the preview pane, adds a new keyword "Summer 2003 Campaign." Having updated the metadata for a multiple selection, the employee then pivots by keyword and can view all of the "Summer 2003 Campaign" files grouped together. Many other scenarios which take advantage of the present invention would be apparent to those skilled in the art.

Alternative embodiments and implementations of the present invention will become apparent to those skilled in the art to which it pertains upon review of the specification, including the drawing figures. Accordingly, the scope of the present invention is defined by the appended claims rather than the foregoing description.

CLAIMS

What is claimed is:

1. A shell browser comprising: a window which displays a group of items, wherein the window also displays metadata values associated with one or more of the displayed items; and an edit control which permits user modification of at least a portion of the metadata values displayed in the window.

2. The shell browser of claim 1, wherein the group of items includes one or more files.

3. The shell browser of claim 1, wherein the group of items includes one or more non-file objects.

4. The shell browser of claim 1, wherein the group of items includes one or more folders.

5. The shell browser of claim 1, wherein the group of items includes a folder and the contents of the folder.

6. The shell browser of claim 1, wherein the shell browser supports a plurality of item types.

7. The shell browser of claim 6, wherein the plurality of item types includes music files, picture files and/or document files.

8. The shell browser of claim 1, wherein the metadata values are displayed in a preview pane area in the window.

9. The shell browser of claim 8, wherein the group of items are displayed in a listview area in the window.

10. The shell browser of claim 1, wherein the group of items are displayed in response to an input from the user.

11. The shell browser of claim 1, wherein the displayed metadata values are associated with only one of the displayed items in the event that the user has selected only that item.

12. The shell browser of claim 1, wherein the displayed metadata values are associated with multiple displayed items in the event that the user has selected those items.

13. The shell browser of claim 1, wherein the displayed metadata values are associated with the group of items collectively in the event that the user has not selected any of the items.

14. The shell browser of claim 1, wherein the displayed metadata values are associated with the group of items collectively in the event that the user has selected all of the items.

15. The shell browser of claim 1, wherein user modification of metadata values includes adding, editing and/or deleting metadata values.

16. The shell browser of claim 1, wherein all of the displayed metadata values are user modifiable values.

17. The shell browser of claim 1, wherein a first portion of the displayed metadata values are read-only values and a second portion of the displayed metadata values are user modifiable values.

18. The shell browser of claim 1, wherein the edit control is displayed in the window in proximity to the displayed metadata values.

19. The shell browser of claim 1, wherein the edit control is displayed whenever the user hovers a mouse cursor over a displayed metadata value which is user modifiable.

20. The shell browser of claim 1, wherein the edit control includes one or more context menus.

21. The shell browser of claim 1, wherein the edit control is configured to permit user modification of a displayed metadata value in the event that the user clicks on the metadata value and provided that the metadata value is user modifiable.

22. An operating system stored on one or more computer-readable media, the operating system including the shell browser of claim 1.

23. A computer comprising a central processing unit, a memory and an operating system including the shell browser of claim 1.

24. A graphical user interface embodied on one or more computer-readable media and executable on a computer, said graphical user interface comprising: a first screen area which displays a set of items in response to an input from a user in a shell browser; a second screen area which displays metadata associated with one or more of the displayed items; and means within the shell browser for enabling user modifications to the displayed metadata.

25. The graphical user interface of claim 24, wherein the set of items includes one or more files.

26. The graphical user interface of claim 24, wherein the set of items includes one or more non-file objects.

27. The graphical user interface shell browser of claim 24, wherein the set of items includes one or more folders.

28. The graphical user interface of claim 24, wherein the set of items includes a folder and the contents of the folder.

29. The graphical user interface of claim 24, wherein the shell browser supports a plurality of item types.

30. The graphical user interface of claim 29, wherein the plurality of item types includes music files, picture files and/or document files.

31. The graphical user interface of claim 24, wherein the displayed metadata is associated with only one of the displayed items in the event that the user has selected only that item.

32. The graphical user interface of claim 24, wherein the displayed metadata is associated with multiple displayed items in the event that the user has selected those items.

33. The graphical user interface of claim 24, wherein the displayed metadata values are associated with the set of items collectively.

34. The graphical user interface of claim 24, wherein user modifications of metadata include adding new metadata and/or editing displayed metadata.

35. The graphical user interface of claim 24, wherein the means for enabling user modifications includes one or more context menus.

36. The graphical user interface of claim 24, wherein the means for enabling user modifications is configured to permit user modification of metadata values when the user clicks on displayed metadata.

37. The graphical user interface of claim 24, wherein the means for enabling user modifications is located within the first screen area.

38. The graphical user interface of claim 24, wherein the means for enabling user modifications is located within the second screen area.

39. An operating system stored on one or more computer-readable media, the operating system including the graphical user interface of claim 24.

40. A computer comprising a central processing unit, a memory, and an operating system including the graphical user interface of claim 24.

41. A computer-implemented method for enabling a user to modify metadata within a shell browser, the method comprising: displaying a plurality of items; receiving a first input from the user representing a selection of at least one displayed item; displaying metadata associated with the at least one selected item; and providing a control for user modification of the displayed metadata.

42. The computer-implemented method of claim 41, further comprising: receiving a second input from the user representing a modification to the displayed metadata; and associating the modified metadata with the at least one selected item so that the modified metadata will be displayed the next time the selected item is displayed.

43. The computer-implemented method of claim 41, wherein the first input represents a selection of two or more displayed items and the displayed metadata is associated with all of the selected items.

44. The computer-implemented method of claim 41, wherein the displayed items are shell items such as files, folders and/or non-file objects.

45. One or more computer-readable media having computer-executable instructions for performing the method of claim 41.

46. One or more computer-readable media having computer-executable instructions for performing the method of claim 42.

47. A computer-implemented method for enabling a user to modify metadata within a shell browser, the method comprising: displaying a welcome pane and metadata associated with the welcome pane; and providing a control for user modification of the displayed metadata.

48. The computer-implemented method of claim 47, further comprising: receiving an input from the user representing a modification to the displayed metadata; and associating the modified metadata with the welcome pane so that the modified metadata will be displayed the next time the welcome pane is displayed.

49. The computer-implemented method of claim 47, wherein the displayed items are shell items such as files, folders and/or non-file objects.

50. One or more computer-readable media having computer-executable instructions for performing the method of claim 47.

51. One or more computer-readable media having computer-executable instructions for performing the method of claim 48.

52. One or more computer-readable media having stored thereon a data structure containing metadata associated with one or more items displayed in a shell browser, said data structure comprising: a first field containing user modifiable metadata associated with the one or more displayed items, wherein the user modifiable metadata contained in the first field of the data structure is also displayed in the shell browser.

53. The one or more computer-readable media of claim 52, wherein the user modifiable metadata contained in the first field of the data structure includes previous user modifications.

54. The one or more computer-readable media of claim 52, wherein the displayed items are items such as files, folders and/or non-file objects.

55. The one or more computer-readable media of claim 52, wherein the data structure further comprises a second field containing read-only metadata associated with the one or more displayed items, wherein the read-only metadata is also displayed in the shell browser.

56. The one or more computer-readable media of claim 55, wherein the data structure further comprises a third field containing a link to all metadata associated with the one or more displayed items, wherein the link is also displayed in the shell browser.