

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

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
31 March 2011 (31.03.2011)

(10) International Publication Number
WO 2011/037723 A2

(51) International Patent Classification:
G06F 3/048 (2006.01) *G06F 3/041* (2006.01)
G06F 9/44 (2006.01)

CA, CH, CL, 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, NL, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(21) International Application Number:
PCT/US2010/047060

(22) International Filing Date:
28 August 2010 (28.08.2010)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/245,237 23 September 2009 (23.09.2009) US
12/700,611 4 February 2010 (04.02.2010) US

(71) Applicant (for all designated States except US): MICROSOFT CORPORATION [US/US]; One Microsoft Way, Redmond, Washington 98052-6399 (US).

(72) Inventors: HARRIS, Jonathan; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US). ALLEN, Andrew; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (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,

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

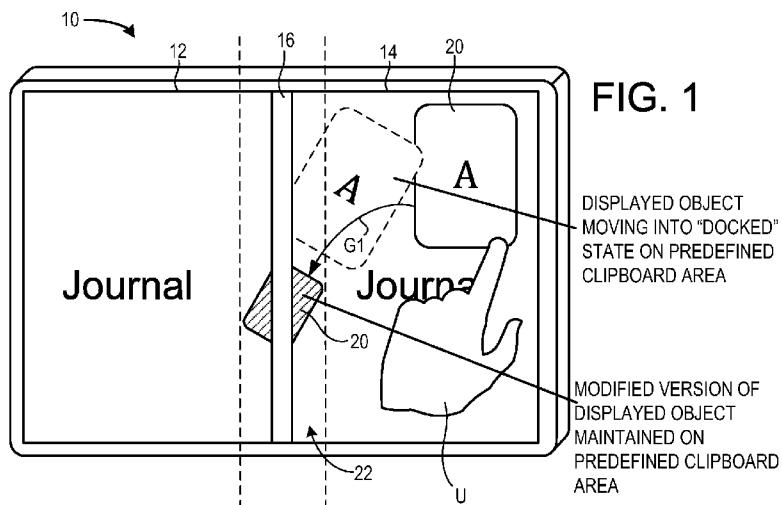
Declarations under Rule 4.17:

- as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))

Published:

- without international search report and to be republished upon receipt of that report (Rule 48.2(g))

(54) Title: COMPUTING SYSTEM WITH VISUAL CLIPBOARD



(57) Abstract: A visual clipboard system and method are provided, in which drag-and-release operations are employed to move copies of a displayed object onto and off of a predefined clipboard area. When a copy of a displayed object is placed on the predefined clipboard area, the displayed object is visually modified to indicate that the visually-modified copy is docked on the predefined clipboard area, and thus eligible for use in a copy operation. When a copy is removed from the predefined clipboard area and placed in a desired target location, the original appearance of the displayed object is restored.

WO 2011/037723 A2

COMPUTING SYSTEM WITH VISUAL CLIPBOARD

BACKGROUND

[0001] Various methods exist in computers and other electronic devices for copying, pasting and moving items. For example, the use of a simple clipboard mechanism for copying and pasting text and other items has been used in personal computers for quite some time. Although various improvements have been devised, current clipboard solutions are often limited in terms of flexibility and ease of use. In addition, many of the existing solutions do not take appropriate advantage of the enhanced interactivity that is offered through touch displays and other contemporary interface mechanisms.

SUMMARY

[0002] Accordingly, the present description provides a visual clipboard system and method in which a displayed object is placed onto a visual clipboard in response to a drag-and-release operation applied to the displayed object. Upon termination of the operation in a predefined clipboard area, a visually-modified version of the displayed object is generated and positioned in the predefined clipboard area. The visually-modified version of the displayed object is maintained in the predefined clipboard area until it is removed in response to another drag-and-release operation. In particular, in response to such an operation that terminates outside of the predefined clipboard area, the original appearance of the displayed object is restored and the displayed object is disposed in a target location.

[0003] This Summary is provided to introduce a selection of concepts in a simplified form that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject matter. Furthermore, the claimed subject matter is not limited to implementations that solve any or all disadvantages noted in any part of this disclosure.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] Fig. 1 is a depiction of an exemplary electronic device employing the visual clipboard functionality of the present description. The figure also depicts an exemplary placing of a selected object on the visual clipboard, so as to place a copy of the object in a docked state on the clipboard.

[0005] Fig. 2 depicts, with respect to the visual clipboard and selected object of Fig. 1, an exemplary removal (e.g., copy or move) of the copy of the selected object from the visual clipboard.

[0006] Fig. 3 depicts the exemplary device and visual clipboard of Figs. 1 and 2, as used with multiple objects that may be selectively placed in and removed from the visual clipboard in copy operations.

[0007] Fig. 4 is an exemplary computing device that may be used in connection

5 with or to provide the visual clipboard functionality of the present description.

[0008] Fig. 5 depicts an exemplary visual clipboard method.

DETAILED DESCRIPTION

[0009] The present disclosure relates to a visual clipboard for enabling copy, paste and/or move operations in an electronic device. All of these operations involve copying of 10 an item, though they may sometimes be referred to as “move” or “paste” operations without specific reference to the copying step that enables the ultimate moving or pasting of the item in a new location.

[0010] Figs. 1-3 depict an exemplary electronic device 10 with which the visual

clipboard may be employed. Device 10 includes two display screens 12 and 14, which

15 may be physically and/or virtually separated by an intervening separator 16, also referred

to herein as a divider. In some embodiments, separator 16 is implemented to include a

folding hinge which enables the two screens to be folded against one another, thereby

providing the device with a note-book type form factor. Additional examples of such

devices may be found in U.S. Serial Number 12/410,303 entitled “Dual Screen Portable

20 Touch Sensitive Computing System” and Serial Number 12/410,311 entitled “Bimodal

Touch Sensitive Digital Notebook”, both filed on March 24, 2009, the disclosures of

which are incorporated herein by this reference, in their entireties and for all purposes.

Various examples of the clipboard will be described in the context of device 10, though it

will be readily appreciated that the described clipboard functionality may be used with a

25 variety of other device types and form factors.

[0011] Referring to Fig. 1, the figure illustrates the placing of an object 20 on or

into a visual clipboard 22, which in the present example is a predefined margin area or

region defined near and on opposing sides of separator 16. Object 20 may be of virtually

any type or format, or visual representation thereof, including a photograph or other

30 image, file, icon, text, audio clip, video clip, grouping of items, etc.

[0012] Object 20 is placed on the visual clipboard 22 in response to a user input.

In the present example, display screens 12 and 14 are touch-interactive. Accordingly, one

method of placing an object onto clipboard 22 includes moving the object via a “drag-and-

release” touchscreen gesture G1 applied by user U. In particular, the object may be

dragged with a user's finger from its original location and released at a location on or within the area of visual clipboard 22. Initiating the operation may optionally including tapping, holding or performing some other action to place the object into an "editable" state allowing it to be moved or otherwise manipulated/edited. In any event, upon bringing the object into the region of the clipboard, the user would release the object, thereby causing the object to be placed on the clipboard. Alternatively, a menu item, button, mouse gesture or other actuator or interactive mechanism could be employed to place a selected object onto the visual clipboard.

[0013] In some cases, it may be desirable to provide a visual indication that a given object is being held on the visual clipboard. In Fig. 1, for example, object 20 is rotated and shown in a reduced size to indicate the object is "tucked" or "docked" (i.e., placed) onto the clipboard. Additionally, or alternatively, an item's status as being on the clipboard may be indicated via color, a specially-assigned icon, dimming, transparent representation, or any other appropriate indication.

[0014] In addition, an indication may be provided to the user that a given object is eligible or ready to be placed onto the clipboard. For example, upon reaching the region of the clipboard, but prior to release by the user's finger, the object may rotate, change color, shake/vibrate, and/or some other indication may be provided to indicate that a release of the object will result in its placement onto the clipboard. Furthermore, the availability of the clipboard may depend on the object being held in the clipboard region for a set amount of time (e.g., 2 seconds) before the clipboard is activated to allow the object to be docked.

[0015] The region comprising the visual clipboard may be defined and configured in various ways. The region may be placed adjacent on both opposing sides of a dual-screen separator, as in the example of Figs. 1-3. In this arrangement, the icon or image representing a docked item may be partly visible on both of display screens 12 and 14. Alternatively, the clipboard region could be placed in other border regions of a screen, or on any other viewable portion of a display screen.

[0016] Various possibilities exist for indicating the presence of the clipboard functionality to the user. In one example, the clipboard region is defined in a region, but the region is not visually indicated on the display (or at least not until the region is used for clipboard functions). For example, the visual clipboard 22 of Figs. 1-3 is defined by the region between the vertical dashed lines to either side of separator 16, however the region is not explicitly shown or otherwise visually delineated to the user. The presence

of the region is known to the user, however (e.g., through tutorials or other communications about product features), and its existence and general shape/configuration can be discerned when objects are brought into the region, such as by dragging or other moving operations. Alternatively, the region's boundaries may be explicitly indicated to the user. In one example, the explicit indication is not provided until an object is brought near or into the clipboard region. In other examples, boundaries of the region are visually delineated to the user whether or not objects are near or on the clipboard.

[0017] Fig. 2 illustrates the removal of object 20 from visual clipboard 22. As previously indicated, the docking functionality may be configured so that the docked object appears partly on both sides of separator 16 and is therefore accessible from either screen 12 or 14. In the exemplary undocking operation of Fig. 2, docked object 20 is accessed via another drag-and-release touchscreen gesture G2 applied by user U. In this example, the gesture is applied to the portion of visual clipboard 22 located on screen 12 to the left of separator 16. Accordingly, it will be appreciated that a copy/move operation may be performed in which the source location is one of the display screens, with the destination location being on the other display screen. In any case, an additional gesture or operation may be employed to retrieve the object from the clipboard. For example, the user may need to rest his/her finger on the docked object for a predetermined time interval before being able to drag/remove the object from the clipboard. During movement to the desired destination, object 20 may remain in its diminished, rotated state, for example until it is released at its desired resting location. After placement at the desired location, the object may remain in an editable state, for example until the user "taps" the object or otherwise causes the object to transition to a non-editable state.

[0018] Also, as will be appreciated from the example of Fig. 2, undocking the object may also include restoring the object to its original appearance. In the specific example of Fig. 2, object 20 is returned to its original size and orientation after the drag-and-release operation has caused it to be placed in the desired target location.

[0019] During docking or undocking operations, differentiation may be employed to provide a user with the ability to selectively choose different clipboard operations. For example, in a bi-modal interactive system, one mode may be employed to provide a copy operation, while another mode provides a move operation. Referring particularly to the undocking operation of Fig. 2, accessing the docked item with a stylus or pen device could produce a copy of the object for positioning in a new location, while access with the user's finger could result in a move operation (i.e., leaving no copy behind in the clipboard).

[0020] Referring now to Fig. 3, visual clipboard 22 may be used to provide copy/paste and/or move functionality for multiple objects 20 (i.e., object 20a, 20b, etc.). When multiple objects are placed on the clipboard, various methods may be employed to position the objects on the clipboard, relative to one another and to other portions of the display screen or screens. For example, placement routines may be employed to take into account some or all of the following criteria/factors:

- User intent: i.e., routine(s) may be employed to attempt to retain the object docked in the location where it was placed by the user
- Spatial distribution: i.e., routines may be employed to achieve uniform or other spatial distributions of multiple objects placed on the clipboard
- Placement/positioning may be predicated on when objects were docked, object type, filename, object date, etc.
- Accessibility: i.e., sizing and positioning objects to facilitate access via finger touch or other means

[0021] In addition to the benefits and advantages recited above, the described visual clipboard may provide the following benefits and advantages, depending upon the particulars of a given implementation:

- The ability to copy and paste and move multiple objects via touch gestures: In particular, multiple object functionality may be facilitated by objects retaining aspects of their appearance when placed in the “docked” state on the visual clipboard. Accordingly, when multiple objects are docked on the clipboard, a user may more easily select a desired object for withdrawal.
- The ability to drag objects into and out of a holding place on screen
- The ability to transfer objects from one screen to another via an on-screen holding location
- The ability to move objects to the spine region of a dual-screen display, and hold the objects there for later use: For example, a notebook form factor device might employ such a spine region. Docked items could be retained near the spine region while the user navigated through a plurality of pages in order to find a desired target page on which to place one of the docked items. Upon finding the desired target page and causing it to be displayed on one of the screens, the desired docked item could be withdrawn and placed in a desired location on the newly-displayed screen. In this example, the ability to move

one or more items to new, “distant” source locations is enhanced by use of the described visual clipboard.

[0022] In addition, the docked objects typically are maintained on the predefined clipboard area during navigation or other modification to content displayed on the one or

5 more adjoining display screens. In particular, user navigation commands may be employed in the depicted notebook form factor device to step through a sequence of content pages. This flipping of content pages may occur in order for the user to select a desired target location (e.g., a page) to place a copy of an object that has been placed on the clipboard.

10 **[0023]** From the above, it should be understood that the present disclosure also encompasses a visual clipboard method. In some cases, as shown in the example of Fig. 4, such a method may be employed in connection with the execution of interface instructions

46 stored on a data-holding subsystem 44 of a computing device 40. In such a case, a processor or other logic subsystem 42 may be employed in connection with executing the

15 instructions to perform copy/move operations of items displayed on display subsystem 48. It should be appreciated that computing device 40 is a generalized example of a device which may employ the visual clipboard functionality described herein, and in specific implementations the device may include various features of the examples discussed with reference to Figs. 1-3.

20 **[0024]** When included in the present examples, a logic subsystem (e.g., logic subsystem 42) may include one or more physical devices configured to execute one or more instructions. For example, the logic subsystem may be configured to execute one or more instructions that are part of one or more programs, routines, objects, components, data structures, or other logical constructs. Such instructions may be implemented to

25 perform a task, implement a data type, transform the state of one or more devices, or otherwise arrive at a desired result. The logic subsystem may include one or more processors that are configured to execute software instructions. Additionally or alternatively, the logic subsystem may include one or more hardware or firmware logic machines configured to execute hardware or firmware instructions. The logic subsystem

30 may optionally include individual components that are distributed throughout two or more devices, which may be remotely located in some embodiments.

[0025] When included in the present examples, a data-holding subsystem (e.g., data-holding subsystem 44) may include one or more physical devices configured to hold data and/or instructions executable by the logic subsystem to implement the herein

described methods and processes. When such methods and processes are implemented, the state of data-holding subsystem may be transformed (e.g., to hold different data). The data-holding subsystem may include removable media and/or built-in devices. The data-holding subsystem may include optical memory devices, semiconductor memory devices, and/or magnetic memory devices, among others. The data-holding subsystem may include devices with one or more of the following characteristics: volatile, nonvolatile, dynamic, static, read/write, read-only, random access, sequential access, location addressable, file addressable, and content addressable. In some embodiments, a logic subsystem and data-holding subsystem may be integrated into one or more common devices, such as an application specific integrated circuit or a system on a chip. The data-holding subsystem may also be in the form of computer-readable removable media, which may be used to store and/or transfer data and/or instructions executable to implement the herein described methods and processes.

[0026] When included, a display subsystem (e.g., display subsystem 48) may be used to present a visual representation of data held by a data-holding subsystem. As the herein described methods and processes change the data held by the data-holding subsystem, and thus transform the state of the data-holding subsystem, the state of the display subsystem may likewise be transformed to visually represent changes in the underlying data. The display subsystem may include one or more display devices utilizing virtually any type of technology. Such display devices may be combined with a logic subsystem (e.g., logic subsystem 42) and/or a data-holding subsystem (e.g., data-holding subsystem 44) in a shared enclosure, or such display devices may be peripheral display devices.

[0027] Fig. 5 depicts an exemplary method 60 that may be employed to provide visual clipboard functionality. In the context of an object that is displayed on a display screen, the method may include first determining whether the object is editable and therefore able to be copied, moved, etc., as shown at 62. As previously discussed, an object may be made editable through various actions. One example includes the user tapping or otherwise selecting the displayed object. Then, as shown at 64, a user input may be received in the form of a drag-and-release operation, such as a touchscreen gesture. If the drag-and-release operation terminates within the clipboard area (e.g., a predefined margin area of a display screen), then a visually-modified version of the display object is generated and positioned within the clipboard area, as shown at 66. As discussed above, the modification may include rotation, size variation, dimming,

transparent appearance, special coloration, etc. In general, however, it will often be desirable that the modified version be similar in many respects to the original (e.g., similar shape and other details). This may be particularly useful in implementations where multiple items are “docked” on the clipboard area. Retaining aspects of the original 5 appearance of the object can facilitate the user selecting the correct desired object when multiple objects are present.

[0028] Continuing with exemplary method 60, step 68 illustrates an example of how the copy/move operation is completed by removing a docked item from the clipboard area. In particular, in response to a drag-and-release operation applied to a docked item 10 (i.e., a modified version of a displayed object that is positioned on the clipboard) and that terminates outside the clipboard area, the docked item is restored to its original unmodified appearance and placed in a target location. Typically, the target location is determined by where the undocking gesture terminates (e.g., where the user lifts/releases his/her finger to “let go” of the item being manipulated).

[0029] It is to be understood that the configurations and/or approaches described herein are exemplary in nature, and that these specific embodiments or examples are not to be considered in a limiting sense, because numerous variations are possible. The specific routines or methods described herein may represent one or more of any number of processing strategies. As such, various acts illustrated may be performed in the sequence 20 illustrated, in other sequences, in parallel, or in some cases omitted. Likewise, the order of the above-described processes may be changed.

[0030] The subject matter of the present disclosure includes all novel and nonobvious combinations and subcombinations of the various processes, systems and configurations, and other features, functions, acts, and/or properties disclosed herein, as 25 well as any and all equivalents thereof.

CLAIMS:

1. A visual clipboard method, comprising,

in response to a first drag-and-release operation that is applied to a displayed object and that terminates in a predefined clipboard area of a display screen, creating a visually-modified version of the displayed object and positioning the visually-modified version in the predefined clipboard area;

in response to user navigation commands, stepping through a sequence of displayable content pages and causing display of a selected content page on the display screen;

maintaining the visually-modified version of the displayed object in the predefined clipboard area in a visible state during said stepping through the sequence of displayable content pages; and

in response to a second drag-and-release operation that is applied to the visually-modified version of the displayed object and that terminates on the selected content page outside of the predefined clipboard area, restoring an original appearance of the displayed object and positioning said restoration in a location on the selected content page.

2. The method of claim 1, in which the predefined clipboard area is a predefined margin area defined on two opposing sides of a divider interposed between two display screens of a computing device having a notebook form factor.

3. The method of claim 2, where the first drag-and-release operation is a touchscreen gesture applied to a first one of the two display screens and the second drag-and-release operation is a touchscreen gesture applied to a second one of the two display screens.

4. The method of claim 1, further comprising, for each of a plurality of different displayed objects, positioning and maintaining a visually-modified version of the displayed object in the predefined clipboard area to selectively enable copy operations for the plurality of different displayed objects.

5. The method of claim 1, where said creating step and said positioning step include placing a smaller version of the displayed object in the predefined clipboard area.

6. The method of claim 1, where said creating step and said positioning step include placing a rotated version of the displayed object in the predefined clipboard area.

7. The method of claim 1, where said creating step and said positioning step include placing a dimmer version of the displayed object in the predefined clipboard area.

8. The method of claim 1, where the predefined clipboard area is visibly indicated on the display screen.

9. A computing device with visual clipboard functionality, comprising:

a display subsystem;

a logic subsystem; and

a data-holding subsystem operatively coupled with display subsystem and the logic subsystem; the data-holding subsystem containing interface instructions executable by the logic subsystem to:

in response to a first drag-and-release operation that is applied to a displayed object and that terminates in a predefined clipboard area of a display screen of the display subsystem, create a visually-modified version of the displayed object and position the visually-modified version in the predefined clipboard area;

maintain the visually-modified version of the displayed object in the predefined clipboard area in a visible state; and

in response to a second drag-and-release operation that is applied to the visually-modified version of the displayed object and that terminates outside of the predefined clipboard area, restore an original appearance of the displayed object and position said restoration in a location determined by where the second drag-and-release operation is terminated.

10. The computing device of claim 9, where the computing device has a notebook form factor, in which the display subsystem includes two display screens that are foldably coupled with one another.

11. The computing device of claim 10, where the predefined clipboard area is a predefined margin area that is defined on opposing sides of a divider that separates the two display screens.

12. The computing device of claim 11, where the interface instructions are configured to enable a copy operation in which the first drag-and-release operation is applied to one of the two display screens, with the second drag-and-release operation being applied to the other of the two display screens.

13. The computing device of claim 9, where the interface instructions are configured to create a smaller version of the displayed object and position said smaller version in the predefined clipboard area.

14. The computing device of claim 9, where the interface instructions are configured to create a rotated version of the displayed object and position said rotated version in the predefined clipboard area.

15. The computing device of claim 9, where the interface instructions are configured to create a dimmer version of the displayed object and position said dimmer version in the predefined clipboard area.

1/3

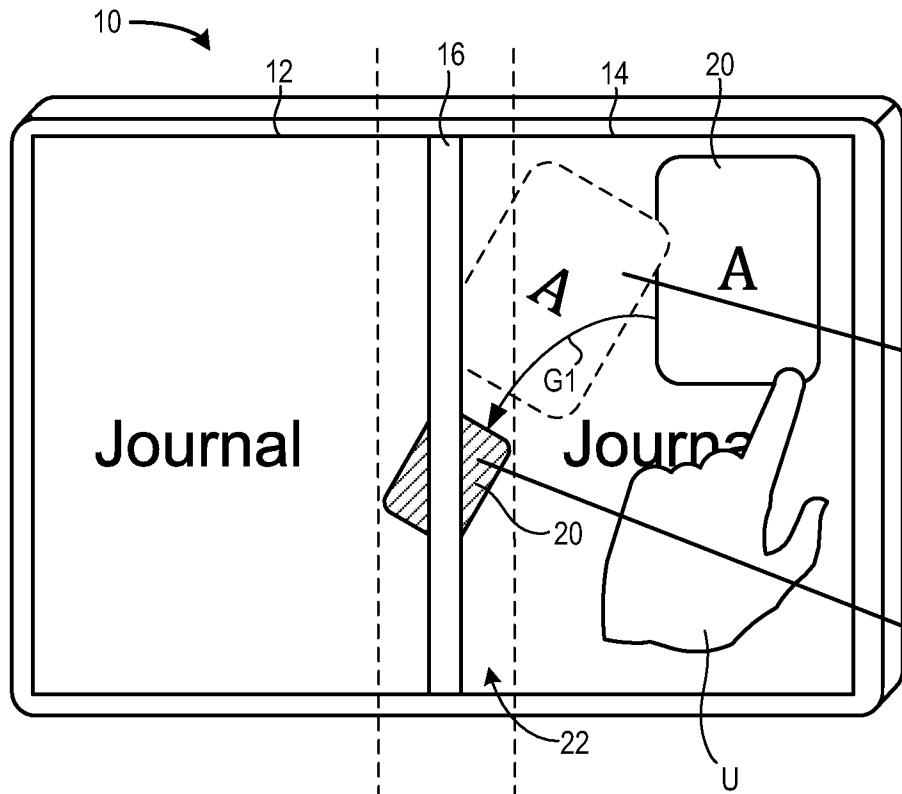


FIG. 1

DISPLAYED OBJECT
MOVING INTO "DOCKED"
STATE ON PREDEFINED
CLIPBOARD AREA

MODIFIED VERSION OF
DISPLAYED OBJECT
MAINTAINED ON
PREDEFINED CLIPBOARD
AREA

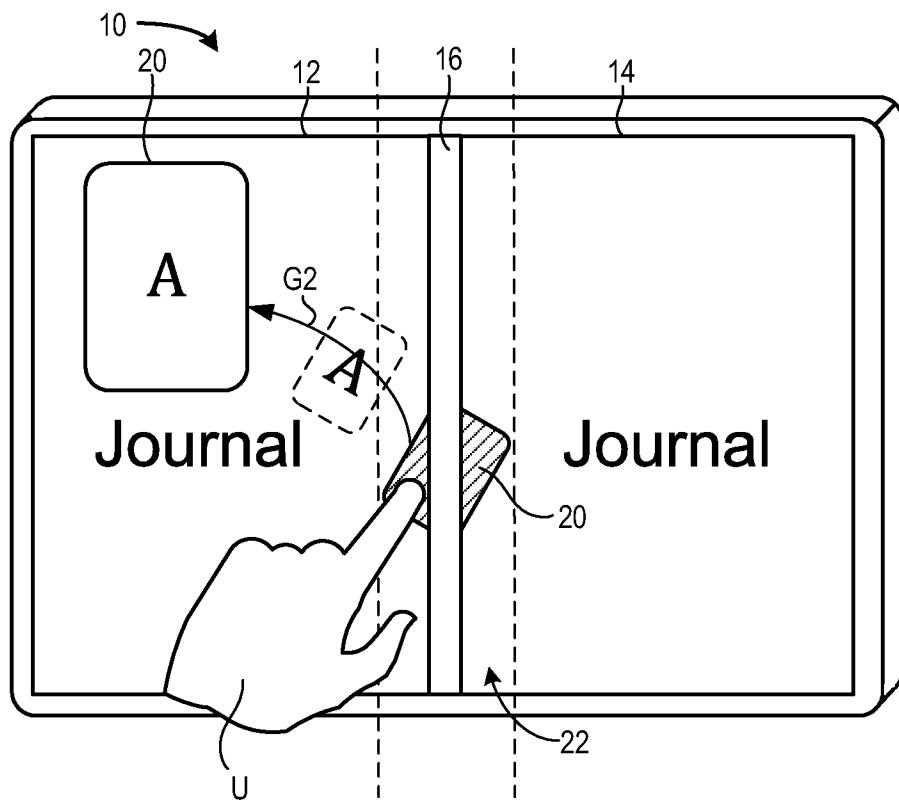
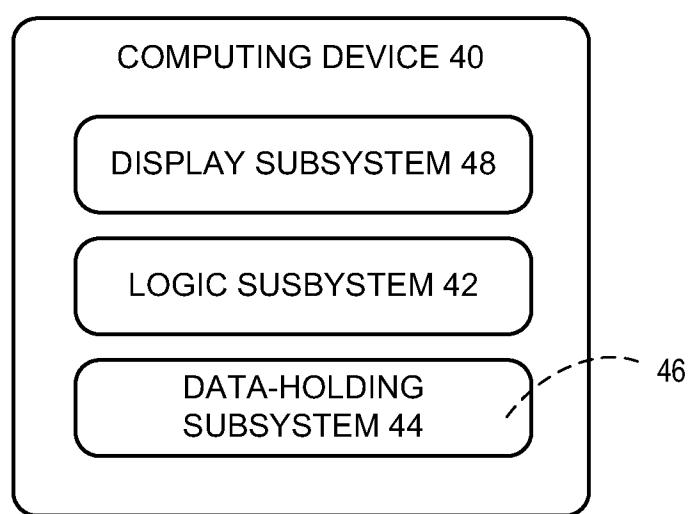
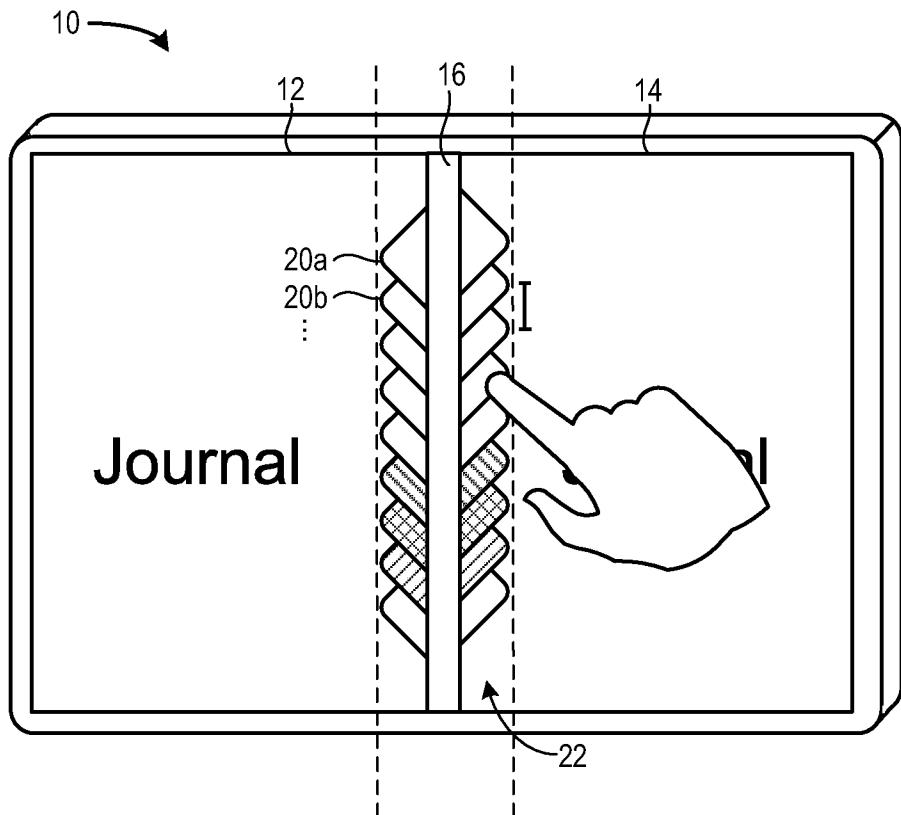


FIG. 2

2/3



3/3

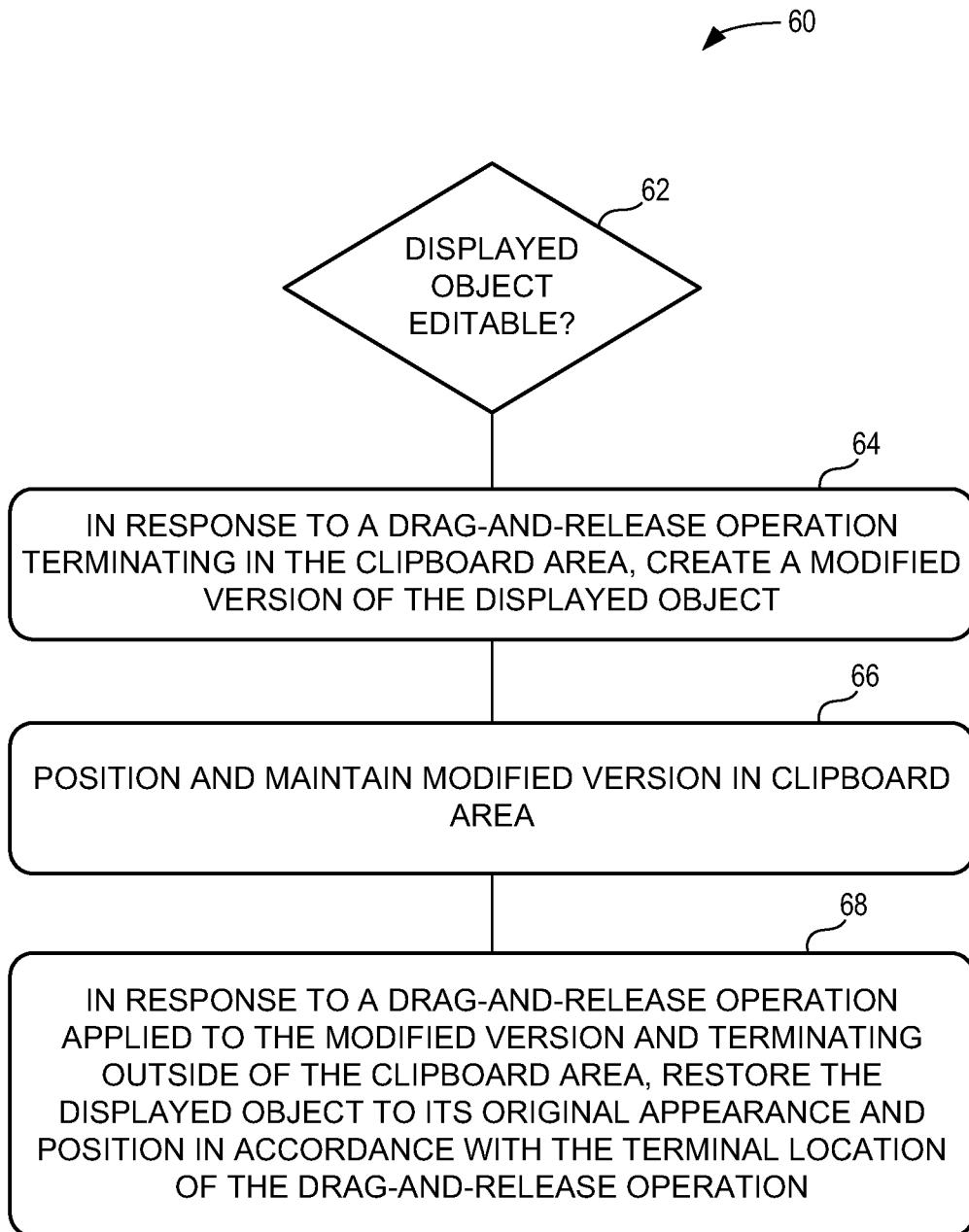


FIG. 5