Techniques for ink for interaction are described. According to various embodiments, ink and touch input may be combined to provide diverse input scenarios. According to various embodiments, ink can be used to reconfigure a document. According to various embodiments, ink can be employed to interact with a map in various ways.
Fig. 1

Client Device 102
- Operating System 104
- Applications 106
- Map Module 108
- Communication Module 110

Input Mechanisms 114
- Digitizer 116
- Touch Input Devices 118

Ink Module 120
- Transient Layer API 122
- Permanent Layer API 124
- Ink & Touch API 126

Ink
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated.
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed in our glasses. Our chairs, being his patents, embraced and
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed in our glasses. Our chairs, being his patents, embraced and
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed in our glasses. Our chairs, being his patents, embraced and

Fig. 5
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed in our glasses. Our chairs, being his patents, embraced and
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed in our glasses. Our chairs, being his patents, embraced and
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. Nice first draft!

His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed in our glasses. Our chairs, being his patents, embraced and

Fig. 8
900 Detect a pen in proximity to an input surface

902 Ascertain a current ink mode

904 Cause a visual affordance identifying the current ink mode to be displayed

906 Process ink content applied to the input surface according to the current ink mode

Fig. 9
1000 Receive ink content applied to a document via input from a pen to an input surface while in a transient ink mode

1002 Detect that the pen is removed from proximity to the input surface

1004 Initiate a timer

1008 Reset the timer

1006 Pen is detected at the input surface before the timer expires?

Yes

No

1010 Remove the ink content from the document and propagate the ink content to a transient layer for the document

Fig. 10
1100 Receive transient ink content to a document from multiple different users

1102 Propagate transient ink content from each user to a different respective transient ink layer for the document

1104 Cause visual affordances of the different transient ink layers to be displayed

1106 Enable each transient ink layer to be individually accessible

Fig. 11
1200 Detect an action to invoke an ink menu

1202 Cause the ink menu to be presented

1204 Receive user input to the ink menu

1206 Perform an action in response to the user input

Fig. 12
1300
Record ink content added to a document

1302
Receive a request to playback the ink content

1304
Cause playback of the ink content

Fig. 13
1400
Detect that new ink content is added while a playback mode for stored ink content is active

1402
Insert the new ink content into the stored ink content
Although Charles Kenneth Scott Moncrieff's translation of *À la recherche du temps perdu* is considered by many journalists and writers to be the best translation of any foreign work into the English language, his choice of *Remembrance of Things Past* as the general title alarmed the seriously ill Proust and misled generations of readers as to the novelist's true intent. It wasn't until 1992 that the title was finally changed to *In Search of Lost Time.*
Although Charles Kenneth Scott Moncrieff’s translation of À la recherche du temps perdu is considered by many journalists and writers to be the best translation of any foreign work into the English language, his choice of Remembrance of Things Past as the general title alarmed the seriously ill Proust and misled generations of readers as to the novelist’s true intent. It wasn’t until 1992 that the title was finally changed to In Search of Lost Time.

Fig. 16
Although Charles Kenneth Scott Moncrieff's translation of _La recherche du temps perdu_ is considered to be the best translation of any foreign work in the English language, his choice of 'Remembrance of Things Past' as the general title alarmed the seriously ill Proust and several generations of readers. Proust's true intent, as to the second part of the novel, was conveyed in his sleepwalking essay 'The Time That Has Gone By'. When Scott Moncrieff chose this title, he did not know, of course, where Proust was going with the story and did not correctly interpret the title, which might indeed be taken to indicate a rather passive attempt by an elderly person to recollect days gone by.

Proust's theory of memory rejects the notion that we can simply sit and quietly resurrect the past in our vividness through the power of our minds. However, the novel is not just a series of anecdotes about life's unfolding events. It is a complex and multi-layered exploration of memory, perception, and the human condition. Proust was interested in the way our perceptions change over time and how the past is reconstructed through memory. He believed that the past is not simply a collection of events, but a subjective experience. Therefore, the novel's title, 'Remembrance of Things Past', is a reflection of Proust's own struggle with memory and the passage of time.
Monitor for changes to primary content of a document that includes an ink annotation

Ascertain that a threshold change occurs to the primary content

Capture an annotation layer for the document that includes the annotation and a version of the primary content prior to the threshold change

Enable the annotation layer to be accessed separately from a current version of the document
1900
Receive a request to present an annotation layer of a document

1902
Cause the annotation layer to be presented

Fig. 19
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed being his patents, embraced and
The Time Traveller (for so it will be convenient to speak of him) was expounding a recondite matter to us. His grey eyes shone and twinkled, and his usually pale face was flushed and animated. The fire burned brightly, and the soft radiance of the incandescent lights in the lilies of silver caught the bubbles that flashed and passed being his patents, embraced and
2200 Detect a user selection of content

2202 Ascertain that an ink note functionality is invoked

2204 Populate the selected content to an ink note

2206 Perform an action in relation to the ink note in response to user input

Fig. 22
2400
Receive ink content in an emphasis ink mode

2402
Initiate a timer for the ink content

2404
Remove the ink content when the timer expires

Fig. 24
Fig. 25
Fig. 27
Fig. 28
3000 Detect that freehand ink content is applied to a display via a pen

3002 Identify a touch gesture that is applied to the display

3004 Map the touch gesture to a particular operation to be performed on the ink content

3006 Modify the ink content by applying the particular operation to the ink content

Fig. 30
Fig. 31
3300
Detect that an ink line is applied to a document displayed on a display device via a pen

3302
Convert the ink line into a document divider that divides the document into a first portion on a first side of the document divider, and a second portion on a second side of the document divider

3304
Receive input to one or more of the first portion or the second portion of the document

3306
Cause a space to be inserted between the first portion and the second portion in response to the input
Travel from Home to Airport
Distance: 23 miles

- Directions
- Traffic
- Navigation

Fig. 35
3900  Receive ink input via a pen to trace a travel route within a map displayed on a display device

3902  Retrieve information about the travel route

3904  Present the information on the display device
4000
Receive a user selection of a travel path in a map

4002
Highlight the travel path for ink input within the map

Fig. 40
4100
Propagate ink input tracing a travel route on a map to a transient ink layer

4102
Enable the transient ink layer to be accessible to retrieve the ink input

Fig. 41
INK FOR INTERACTION

RELATED APPLICATION


BACKGROUND

[0002] Devices today (e.g., computing devices) typically support a variety of different input techniques. For instance, a particular device may receive input from a user via a keyboard, a mouse, voice input, touch input (e.g., to a touchscreen), and so forth. One particularly intuitive input technique enables a user to utilize a touch instrument (e.g., a pen, a stylus, a finger, and so forth) to provide freehand input to a touch-sensing functionality such as a touchscreen, which is interpreted as digital ink. The freehand input may be converted to a corresponding visual representation on a display, such as for taking notes, for creating and editing an electronic document, and so forth.

SUMMARY

[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 as an aid in determining the scope of the claimed subject matter.

[0004] Techniques for ink for interaction are described. According to various embodiments, ink and touch input may be combined to provide diverse input scenarios. According to various embodiments, ink can be used to reconfigure a document. According to various embodiments, ink can be employed to interact with a map in various ways.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] The detailed description is described with reference to the accompanying figures. In the figures, the left-most digit(s) of a reference number identifies the figure in which the reference number first appears. The use of the same reference numbers in different instances in the description and the figures may indicate similar or identical items.

[0006] FIG. 1 is an illustration of an environment in an example implementation that is operable to employ techniques discussed herein in accordance with one or more embodiments.

[0007] FIG. 2 depicts an example implementation scenario for a permanent ink mode in accordance with one or more embodiments.

[0008] FIG. 3 depicts an example implementation scenario for a transient ink mode in accordance with one or more embodiments.

[0009] FIG. 4 depicts an example implementation scenario for a transient ink mode in accordance with one or more embodiments.

[0010] FIG. 5 depicts an example implementation scenario for a transient ink mode in accordance with one or more embodiments.

[0011] FIG. 6 depicts an example implementation scenario for a transient ink mode in accordance with one or more embodiments.

[0012] FIG. 7 depicts an example implementation scenario for a multiple transient ink layers in accordance with one or more embodiments.

[0013] FIG. 8 depicts an example implementation scenario for presenting an ink menu in accordance with one or more embodiments.

[0014] FIG. 9 is a flow diagram that describes steps in a method for processing ink in accordance with one or more embodiments.

[0015] FIG. 10 is a flow diagram that describes steps in a method for a transient ink timer in accordance with one or more embodiments.

[0016] FIG. 11 is a flow diagram that describes steps in a method for propagating transient ink to different transient ink layers for different users in accordance with one or more embodiments.

[0017] FIG. 12 is a flow diagram that describes steps in a method for presenting an ink menu in accordance with one or more embodiments.

[0018] FIG. 13 is a flow diagram that describes steps in a method for ink playback in accordance with one or more embodiments.

[0019] FIG. 14 is a flow diagram that describes steps in a method for inserting ink content in an ink playback mode in accordance with one or more embodiments.

[0020] FIG. 15 depicts an example implementation scenario for ink versioning in accordance with one or more embodiments.

[0021] FIG. 16 depicts an example implementation scenario for ink versioning in accordance with one or more embodiments.

[0022] FIG. 17 depicts an example implementation scenario for presenting a current version of a document along with an annotation layer in accordance with one or more embodiments.

[0023] FIG. 18 is a flow diagram that describes steps in a method for ink versioning in accordance with one or more embodiments.

[0024] FIG. 19 is a flow diagram that describes steps in a method for enabling access to an annotation layer in accordance with one or more embodiments.

[0025] FIG. 20 depicts an example implementation scenario for ink notes in accordance with one or more embodiments.

[0026] FIG. 21 depicts an example implementation scenario for ink notes in accordance with one or more embodiments.

[0027] FIG. 22 is a flow diagram that describes steps in a method for generating an ink note in accordance with one or more embodiments.

[0028] FIG. 23 depicts an example implementation scenario for emphasis in accordance with one or more embodiments.

[0029] FIG. 24 is a flow diagram that describes steps in a method for emphasis in accordance with one or more embodiments.

[0030] FIG. 25 depicts an example implementation scenario for ink and touch in accordance with one or more embodiments.

[0031] FIG. 26 depicts an example implementation scenario for ink and touch in accordance with one or more embodiments.
FIG. 27 depicts an example implementation scenario for ink and touch in accordance with one or more embodiments.

FIG. 28 depicts an example implementation scenario for ink and touch in accordance with one or more embodiments.

FIG. 29 depicts an example implementation scenario for ink and touch in accordance with one or more embodiments.

FIG. 30 is a flow diagram that describes steps in a method for ink and touch in accordance with one or more embodiments.

FIG. 31 depicts an example implementation scenario for ink for adding space in accordance with one or more embodiments.

FIG. 32 depicts an example implementation scenario for ink for adding space in accordance with one or more embodiments.

FIG. 33 is a flow diagram that describes steps in a method for ink for document reconfiguration in accordance with one or more embodiments.

FIG. 34 depicts an example implementation scenario for ink and maps in accordance with one or more embodiments.

FIG. 35 depicts an example implementation scenario for ink and maps in accordance with one or more embodiments.

FIG. 36 depicts an example implementation scenario for ink and maps in accordance with one or more embodiments.

FIG. 37 depicts an example implementation scenario for ink and maps in accordance with one or more embodiments.

FIG. 38 depicts an example implementation scenario for ink and maps in accordance with one or more embodiments.

FIG. 39 is a flow diagram that describes steps in a method for providing information for a travel route based on ink input in accordance with one or more embodiments.

FIG. 40 is a flow diagram that describes steps in a method for highlighting a portion of a travel path to receive ink input in accordance with one or more embodiments.

FIG. 41 is a flow diagram that describes steps in a method for propagating an ink travel route to a transient layer in accordance with one or more embodiments.

FIG. 42 illustrates an example system and computing device as described with reference to FIG. 1, which are configured to implement embodiments of techniques described herein.

DETAILED DESCRIPTION

[0048] Overview

[0049] Techniques for ink for interaction are described. Generally, ink refers to hand-held input to a touch-sensing functionality such as a touchscreen, which is interpreted as digital ink, referred to herein as “ink.” Ink may be provided in various ways, such as using a pen (e.g., an active pen, a passive pen, and so forth), a stylus, a finger, and so forth.

[0050] According to various implementations, ink and touch input may be combined to provide diverse inputs scenarios. For instance, a user may apply ink contact to a document displayed on a device and provide a touch gesture that causes the ink content to be transformed and/or moved. As further detailed below, different gestures can cause different respective operations to be performed on ink content. Combining ink and touch input enables different operations to be performed with minimal user interaction. Thus, a number of user interactions required to transform and/or move ink content is reduced.

[0051] According to various implementations, ink can be used to reconfigure a document. For instance, ink can be applied to divide a document into different sections that are separately manipulatable. A user can then manipulate the sections to reconfigure the document, such as to add space between the sections. Enabling document reconfiguration via ink simplifies document reconfiguration tasks and reduces user interaction required to reconfigure a document.

[0052] According to various implementations, ink can be employed to interact with a map in various ways. For instance, a user can apply ink to trace a travel route on a map. In response to tracing the travel route, information about the travel route is presented, such as to aid the user in traveling the travel route. Generally, ink for map interaction enables a user to easily obtain navigation information for navigating different travel routes. For instance, a user may obtain information about a travel route by simply tracing the travel route with ink, thus reducing a number of user interactions required to obtain travel-related information.

[0053] In the following discussion, an example environment is first described that is operable to employ techniques described herein. Next, a section entitled “Example Implementation Scenarios and Procedures” describes some example implementation scenarios and methods for ink for interaction in accordance with one or more embodiments. Finally, a section entitled “Example System and Device” describes an example system and device that are operable to employ techniques discussed herein in accordance with one or more embodiments.

[0054] Example Environment

[0055] FIG. 1 is an illustration of an environment 100 in an example implementation that is operable to employ techniques for ink for interaction discussed herein. Environment 100 includes a client device 102 which can be embodied as any suitable device such as, by way of example and not limitation, a smartphone, a tablet computer, a portable computer (e.g., a laptop), a desktop computer, a wearable device, and so forth. In at least some implementations, the client device 102 represents a smart appliance, such as an Internet of Things (“IoT”) device. Thus, the client device 102 may range from a system with significant processing power, to a lightweight device with minimal processing power. One of a variety of different examples of a client device 102 is shown and described below in FIG. 42.

[0056] The client device 102 includes a variety of different functionalities that enable various activities and tasks to be performed. For instance, the client device 102 includes an operating system 104, applications 106, a map module 108, and a communication module 110. Generally, the operating system 104 is representative of functionality for abstracting various system components of the client device 102, such as hardware, kernel-level modules and services, and so forth. The operating system 104, for instance, can abstract various components of the client device 102 to the applications 106 to enable interaction between the components and the applications 106.

[0057] The applications 106 represents functionalities for performing different tasks via the client device 102. Examples of the applications 106 include a word processing
application, a spreadsheet application, a web browser, a gaming application, and so forth. The applications 106 may be installed locally on the client device 102 to be executed via a local runtime environment, and/or may represent portals to remote functionality, such as cloud-based services, web apps, and so forth. Thus, the applications 106 may take a variety of forms, such as locally-executed code, portals to remotely hosted services, and so forth.

[0058] The map module 108 is representative of functionality to provide map functionality for the client device 102. For instance, the map module 108 is configured to generate a map graphical user interface (GUI) and populate the map GUI with geographical information, such as visual representations of different geographical locations. Further functionality of the map module 108 is discussed below.

[0059] The communication module 110 is representative of functionality for enabling the client device 102 to communicate over wired and/or wireless connections. For instance, the communication module 110 represents hardware and logic for communication via a variety of different wired and/or wireless technologies and protocols.

[0060] The client device 102 further includes a display device 112, input mechanisms 114 including a digitizer 116 and touch input devices 118, and an ink module 120. The display device 112 generally represents functionality for visual output for the client device 102. Additionally, the display device 112 represents functionality for receiving various types of input, such as touch input, pen input, and so forth. The input mechanisms 114 generally represent different functionalities for receiving input to the client device 102. Examples of the input mechanisms 114 include gesture-sensitive sensors and devices (e.g., such as touch-based sensors and movement-tracking sensors (e.g., camera-based)), a mouse, a keyboard, a stylus, a touch pad, accelerometers, a microphone with accompanying voice recognition software, and so forth. The input mechanisms 114 may be separate or integral with the displays 110; integral examples include gesture-sensitive displays with integrated touch-sensitive or motion-sensitive sensors. The digitizer 116 represents functionality for converting various types of input to the display device 112 and the touch input devices 118 into digital data that can be used by the client device 102 in various ways, such as for generating digital ink.

[0061] According to various implementations, the ink module 120 represents functionality for performing various aspects of techniques for ink for interaction discussed herein. Various functionalities of the ink module 120 are discussed below. The ink module 120 includes a transient layer application programming interface (API) 122, a permanent layer API 124, and an ink & touch API 126. The transient layer API 122 represents functionality for enabling interaction with a transient ink layer, and the permanent layer API 124 represents functionality for enabling interaction with a permanent object (e.g., a document) layer. In at least some implementations, the transient layer API 122 and the permanent layer API 124 may be utilized (e.g., by the applications 106) to access transient ink functionality and permanent ink functionality, respectively.

[0062] The ink & touch API 126 is representative of functionality to enable ink input and touch input to be used together to provide for diverse input and interaction scenarios. Examples functionalities that are enabled by the ink & touch API 126 are detailed below.

[0063] The environment 100 further includes a pen 128, which is representative of an input device for providing input to the display device 112. Generally, the pen 128 is in a form factor of a traditional pen but includes functionality for interacting with the display device 112 and other functionality of the client device 102. In at least some implementations, the pen 128 is an active pen that includes electronic components for interacting with the client device 102. The pen 128, for instance, includes a battery that can provide power to internal components of the pen 128.

[0064] Alternatively or additionally, the pen 128 may include a magnet or other functionality that supports hover detection over the display device 112. This is not intended to be limiting, however, and in at least some implementations the pen 128 may be passive, e.g., a stylus without internal electronics. Generally, the pen 128 is representative of an input device that can provide input that can be differentiated from other types of input by the client device 102. For instance, the digitizer 116 is configured to differentiate between input provided via the pen 128, and input provided by a different input mechanism such as a user's finger or other appendage.

[0065] The pen 128 includes a pen mode button 130, which represents a selectable control (e.g., a switch) for switching the pen 128 between different pen input modes. Generally, different pen input modes enable input from the pen 128 to be utilized and/or interpreted by the ink module 120 in different ways. Examples of different pen input modes are detailed below.

[0066] Having described an example environment in which the techniques described herein may operate, consider now a discussion of an example implementation scenario in accordance with one or more embodiments.

[0067] Transient Ink and Permanent Ink

[0068] According to various implementations, ink can be applied in different ink modes including a transient ink mode and a permanent ink mode. Generally, transient ink refers to ink that is temporary and that can be used for various purposes, such as invoking particular actions, annotating a document, and so forth. For instance, in transient implementations, ink can be used for ink and touch interactions, annotation layers for electronic documents, temporary visual emphasis, text recognition, invoking various commands and functionalities, and so forth.

[0069] Permanent ink generally refers to implementations where ink becomes a part of the underlying object, such as for creating a document, writing on a document (e.g., for annotation and/or editing), applying ink to graphics, and so forth. Permanent ink, for example, can be considered as a graphics object, such as for note taking, for creating visual content, and so forth.

[0070] In at least some implementations, a pen (e.g., the pen 124) applies ink whenever the pen is in contact with an input surface, such as the display device 112 and/or other input surface. Further, a pen can apply ink across many different applications, platforms, and services. In one or more implementations, an application and/or service can specify how ink is used in relation to an underlying object, such as a word processing document, a spreadsheet and so forth. For instance, in some scenarios ink is applied as transient ink, and other scenarios ink is applied as permanent ink. Examples of different implementations and attributes of transient ink and permanent ink are detailed below.
Example Implementation Scenarios and Procedures

This section describes some example implementation scenarios and example procedures for ink modes in accordance with one or more implementations. The implementation scenarios and procedures may be implemented in the environment described above, the system 4200 of FIG. 42, and/or any other suitable environment. The implementation scenarios and procedures, for example, describe example operations of the client device 102. While the implementation scenarios and procedures are discussed with reference to a particular application, it is to be appreciated that techniques for ink for interaction discussed herein are applicable across a variety of different applications, services, and environments. In at least some embodiments, steps described for the various procedures are implemented automatically and independent of user interaction.

FIG. 2 depicts an example implementation scenario 200 for a permanent ink mode in accordance with one or more implementations. The upper portion of the scenario 200 includes a graphical user interface (GUI) 202 displayed on the display device 112. Generally, the GUI 202 represents a GUI for a particular functionality, such as an instance of the applications 106. Also depicted is a user holding the pen 128. Displayed within the GUI 202 is a document 204, e.g., an electronic document generated via one of the applications 106.

Proceeding to the lower portion of the scenario 200, the user brings the pen 128 in proximity to the surface of the display device 112 and within the GUI 202. The pen 128, for instance, is placed within a particular distance of the display device 112 (e.g., less than 2 centimeters) but not in contact with the display device 112. This behavior is generally referred to herein as “hovering” the pen 128. In response to detecting proximity of the pen 128, a hover target 206 is displayed within the GUI 202 and at a point within the GUI 202 that is directly beneath the tip of the pen 128. Generally, the hover target 206 represents a visual affordance that indicates that ink functionality is active such that a user may apply ink to the document 204.

According to various implementations, the visual appearance (e.g., shape, color, shading, and so forth) of the hover target 206 provides a visual cue indicating a current ink mode that is active. In the scenario 200, the hover target is presented as a solid circle, which indicates that a permanent ink mode is active. For instance, if the user proceeds to put the pen 128 in contact with the display device 112 to apply ink to the document 204 in a permanent ink mode, the ink will become part of the document 204, e.g., will be added to a primary content layer of the document 204. Consider, for example, that the text (e.g., primary content) displayed in the document 204 was created via ink input in a permanent ink mode. Thus, ink applied in a permanent ink mode represents a permanent ink layer that is added to a primary content layer of the document 204.

In further response to detecting hovering of the pen 128, an ink flag 208 is visually presented adjacent to and/or at least partially overlaying a portion of the document 204. Generally, the ink flag 208 represents a visual affordance that indicates that ink functionality is active such that a user may apply ink to the document 204. In at least some implementations, the ink flag 208 may be presented additionally or alternatively to the hover target 206. In this particular example, the ink flag 208 includes a visual cue indicating a current ink mode that is active. In the scenario 200, the ink flag 208 includes a solid circle, which indicates that a permanent ink mode is active. As further detailed below, the ink flag 208 is selectable to cause an ink menu to be displayed that includes various ink-related functionalities, options, and settings that can be applied.

FIG. 3 depicts an example implementation scenario 300 for a transient ink mode in accordance with one or more implementations. The upper portion of the scenario 300 includes a graphical user interface (GUI) 302 displayed on the display device 112. Generally, the GUI 302 represents a GUI for a particular functionality, such as an instance of the applications 106. Displayed within the GUI 302 is a document 304, e.g., an electronic document generating via one of the applications 106. The document 304 includes primary content 306, which represents content generated as part of a primary content layer for the document 304. For instance, in this particular example the document 304 is a text-based document, and thus the primary content 306 includes text that is populated to the document. Various other types of documents and primary content may be employed, such as for graphics, multimedia, web content, and so forth.

As further illustrated, a user is hovering the pen 128 within a certain proximity of the surface of the display device 112, such as discussed above with reference to the scenario 200. In response, a hover target 308 is displayed within the document 304 and beneath the tip of the pen. In this particular example, the hover target 308 is presented as a hollow circle, thus indicating that a transient ink mode is active. For instance, if the user proceeds to apply ink to the document 304, the ink will behave according to a transient ink mode. Examples of different transient ink behaviors are detailed elsewhere herein.

Further in response to the user hovering the pen 128 over the display device 112, an ink flag 310 is presented. In this particular example, the ink flag 310 includes a hollow circle 312, providing visual cue that a transient ink mode is active.

Proceeding to the lower portion of the scenario 300, the user removes the pen 128 from proximity to the display device 112. In response, the hover target 308 and the ink flag 310 are removed from the display device 112. For instance, in at least some implementations, a hover target and/or an ink flag are presented when the pen 128 is detected as being hovered over the display device 112, and are removed from the display device 112 when the pen 128 is removed such that the pen 128 is no longer detected as being hovered over the display device 112. This is not intended to be limiting, however, and in at least some implementations, an ink flag may be persistently displayed to indicate that inking functionality is active and/or available.

FIG. 4 depicts an example implementation scenario 400 for a transient ink mode in accordance with one or more implementations. The upper portion of the scenario 400 includes the GUI 302 with the document 304 (introduced above) displayed on the display device 112. In at least some implementations, the scenario 400 represents an extension of the scenario 300, above.

In the upper portion of the scenario 400, a user applies ink content 402 to the document 304 using the pen 128. In this particular scenario, the ink content 402 corresponds to an annotation of the document 402. It is to be appreciated, however, that a variety of different types of transient ink other than annotations may be employed. Notice that as the user is applying the ink content 402, a hover target is not
displayed. For instance, in at least some implementations when the pen 128 transitions from a hover position to contact with the display device 112, a hover target is removed. Notice also that the ink flag 310 includes a hollow circle 312, indicating that the ink content 402 is applied according to a transient ink mode.

[0083] Proceeding to the lower portion of the scenario 400, the user lifts the pen 128 from the display device 112 such that the pen 128 is not detected, e.g., the pen 128 is not in contact with the display device 112 and is not in close enough proximity to the display device 112 to be detected as hovering. In response to the pen 128 no longer being detected in contact with or in proximity to the display device 112, an ink timer 406 begins running. For instance, the ink timer 406 begins counting down from a specific time value, such as 30 seconds, 60 seconds, and so forth. Generally, the ink timer is representative of functionality to implement a countdown function, such as for tracking time between user interactions with the display device 112 via the pen 128. The ink timer 406, for example, represents a functionality of the ink module 120.

[0084] As a visual cue that the ink counter 406 is elapsing, the hollow circle 312 begins to unwind, e.g., begins to disappear from the ink flag 310. In at least some implementations, the hollow circle 312 unwinds at a rate that corresponds to the countdown of the ink timer 406. For instance, when the ink timer 406 is elapsed by 50%, then 50% of the hollow circle 312 is removed from the ink flag 310. Thus, unwinding of the hollow circle 312 provides a visual cue that the ink timer 406 is elapsing, and how much of the ink timer has elapsed and/or remains to be elapsed.

[0085] In at least some implementations, if the ink timer 406 is elapsing as in the lower portion of the scenario 400 and the user proceeds to place the pen 128 in proximity to the display device 112 (e.g., hovered or in contact with the display 110), the ink timer 406 will reset and will not begin elapsing again until the user removes the pen 128 from the display device 112 such that the pen 128 is not detected. In such implementations, the hollow circle 312 will be restored within the ink flag 310 as in the upper portion of the scenario 400.

[0086] FIG. 5 depicts an example implementation scenario 500 for a transient ink mode in accordance with one or more implementations. The upper portion of the scenario 500 includes the GUI 302 with the document 304 (introduced above) displayed on the display device 112. In at least some implementations, the scenario 500 represents an extension of the scenario 400, above.

[0087] In the upper portion of the scenario 500, the ink timer 406 has elapsed. For instance, notice that the hollow circle 312 has completely unwound within the ink flag 310, e.g., is visually removed from the ink flag 310. According to various implementations, this provides a visual cue that the ink timer 406 has completely elapsed.

[0088] Proceeding to the lower portion of the scenario 500, and in response to expiry of the ink timer 406, the ink content 402 is removed from the GUI 302 and saved as part of a transient ink layer 504 for the document 304. Further, the ink flag 310 is populated with a user icon 502. The user icon 502, for example, represents a user that is currently logged in to the client device 102, and/or a user that is interacting with the document 304. Alternatively or additionally, the pen 128 includes user identification data that is detected by the client device 102 and thus is leveraged to track which user is interacting with the document 304. For example, the pen 128 includes a tagging mechanism (e.g., a radio-frequency identifier (RFID) chip) embedded with a user identity for a particular user. Thus, when the pen 128 is placed in proximity to the display device 112, the tagging mechanism is detected by the client device 102 and utilized to attribute ink input and/or other types of input to a particular user. As used herein, the term “user” may be used to refer to an identity for an individual person, and/or an identity for a discrete group of users that are grouped under a single user identity.

[0089] According to various implementations, population of the user icon 502 to the ink flag 310 represents a visual indication that the transient ink layer 504 exists for the document 304, and that the transient ink layer 504 is associated with (e.g., was generated by) a particular user. Generally, the transient ink layer 504 represents a data layer that is a part of the primary content layer of the document 304, but that is persisted and can be referenced for various purposes. Further attributes of transient ink layers are described elsewhere herein.

[0090] FIG. 6 depicts an example implementation scenario 600 for a transient ink mode in accordance with one or more implementations. The upper portion of the scenario 600 includes the GUI 302 with the document 304 (introduced above) displayed on the display device 112. In at least some implementations, the scenario 600 represents an extension of the scenario 500, above.

[0091] In the upper portion of the scenario 600, the ink flag 310 is displayed indicating that a transient ink layer (e.g., the transient ink layer 504) exists for the document 304, and that the transient ink layer is linked to a particular user represented by the user icon 502 in the ink flag 310.

[0092] Proceeding to the lower portion of the scenario 600, a user selects the ink flag 310 with the pen 128, which causes the ink content 402 to be returned to display as part of the document 304. The ink content 402, for example, is bound to the transient ink layer 504, along with other transient ink content generated for the transient ink layer 504. Thus, in at least some implementations, the transient ink layer 504 is accessible by various techniques, such as by selection of the ink flag 310.

[0093] Additionally or alternatively to selection of the ink flag 310, if the user proceeds to apply further ink content to the document 304 while in the transient ink mode, the transient ink layer 504 is retrieved and transient ink content included as part of the transient ink layer 504 is displayed as part of the document 504. In at least some implementations, transient ink content of the transient ink layer 504 is bound (e.g., anchored) to particular portions (e.g., pages, lines, text, and so forth) of the document 304. For instance, the user generated the ink content 402 adjacent to a particular section of text. Thus, when the transient ink layer 504 is recalled as depicted in the scenario 600, the ink content 402 is displayed adjacent to the particular section of text.

[0094] According to various implementations, the transient ink layer 504 is cumulative such that a user may add ink content to and remove ink content from the transient ink layer 504 over a span of time and during multiple different interactivity sessions. Thus, the transient ink layer 504 generally represents a record of multiple user interactions with the document 304, such as for annotations, proofreading, commenting, and so forth. Alternatively or additionally, multiple transient layers may be created for the document 304, such as
when significant changes are made to the primary content 306, when other users apply transient ink to the document 304, and so forth.

[0095] In at least some implementations, when the user pauses interaction with the document 304, the ink timer 406 begins elapsing such as discussed above with reference to the scenarios 400, 500. Accordingly, the scenario 600 may return to the scenario 400.

[0096] FIG. 7 depicts an example implementation scenario 700 for a multiple transient ink layers in accordance with one or more implementations. The upper portion of the scenario 600 includes the GUI 302 with the document 304 (introduced above) displayed on the display device 112. In at least some implementations, the scenario 700 represents an extension of the scenario 600, above.

[0097] Displayed as part of the GUI 302 is the ink flag 310 with the user icon 502, along with an ink flag 702 with a user icon 704, and an ink flag 706 with a user icon 708. Generally, the ink flags 702, 706 represent other users that have interacted with the document 304, and the user icons 704, 708 represent users associated with their respective ink flags. Thus, each individual ink flag represents a different respective user.

[0098] The scenario 700 further includes the transient ink layer 504 associated with the ink flag 310, along with a transient ink layer 710 linked to the ink flag 702, and a transient ink layer 712 linked to the ink flag 706. Generally, the transient ink layers 710, 712 represent individual transient ink layers that are bound to individual user identities. Each individual transient ink layer 504, 710, 712 is individually accessible can be viewed and edited separately. In at least some implementations, the multiple of the transient ink layers 504, 710, 712 can be invoked such that ink content for the multiple layers is displayed concurrently as part of the document 304. Example ways of invoking a transient ink layer are detailed elsewhere herein. Further, transient ink layer behaviors discussed elsewhere herein applicable to the scenario 700.

[0099] FIG. 8 depicts an example implementation scenario 800 for presenting an inking menu in accordance with one or more implementations. The upper portion of the scenario 800 includes the GUI 302 with the document 304 (introduced above) displayed on the display device 112. In at least some implementations, the scenario 800 represents an extension of the scenarios discussed above. Further to the scenario 800, the user selects the ink flag 310 while the transient layer 504 is active. For instance, the user manipulates the pen 128 to first tap the ink flag 310, which invokes the transient ink layer 504 such that the ink content 402 is retrieved and displayed, and then taps the ink flag 310 a second time within a particular period of time, e.g., 3 seconds.

[0100] Proceeding to the lower portion of the scenario 800 and in response to the second selection of the ink flag 310, the ink flag 310 expands to present an ink menu 802. Generally, the ink menu 802 includes multiple selectable inicia that is selectable to cause different ink-related actions to be performed, such as to apply and/or change various settings, invoke various functionalities, and so forth. To aid in visual understanding, an expanded representation 802a of the ink menu 802 is depicted. The example visual inicia included in the ink menu 802 are now discussed in turn.

[0101] Play control 804—according to various implementations, when ink content (e.g., transient and/or permanent ink) is applied to a document, application of the ink content is recorded in real-time. For instance, application of ink content is recorded as an animation that shows the ink content being applied to a document as it was initially applied by a user. Accordingly, selection of the play control 804 causes a playback of ink content as it was originally applied. Further details concerning ink playback are presented below.

[0102] Transient Ink Control 806—selection of this control causes a transition from a different ink mode (e.g., a permanent ink mode) to a transient ink mode.

[0103] Permanent Ink Control 808—selection of this control causes a transition from a different ink mode (e.g., a transient ink mode) to a permanent ink mode.

[0104] Text Recognition Control 810—selection of this control causes a transition to a text recognition mode. For instance, in a text recognition mode, characters applied using ink are converted into machine-encoded text.

[0105] Shape Recognition Control 812—selection of this control causes a transition to a shape recognition mode. For instance, in a shape recognition mode, shapes applied using ink are converted into machine-encoded shapes, such as quadrilaterals, triangles, circles, and so forth.

[0106] Selection Mode Control 814—selection of this control causes a transition to a selection mode. Generally, in a selection mode, input from a pen is interpreted as a selection action, such as to select text and/or other objects displayed in a document.

[0107] Erase Mode Control 816—selection of this control causes a transition to an erase mode. Generally, in an erase mode, input from a pen is interpreted as an erase action, such as to erase ink, text, and/or other objects displayed in a document.

[0108] Command Control 818—selection of this control causes a transition to a command mode. For instance, in a command mode, input from a pen is interpreted as a command to perform a particular action and/or task.

[0109] Color Control 820—selection of this control enables a user to change an ink color that is applied to a document. For example, selection of this control causes a color menu to be presented that includes multiple different selectable colors. Section of a color from the color menu specifies the color for ink content that is applied to a document.

[0110] Ink Note Control 822—this control is selectable to invoke ink note functionality, such as to enable ink content to be propagated to a note ink note functionality is described in more detail below.

[0111] Emphasis Control 824—selection of this control causes a transition from a different ink mode (e.g., a permanent or transient ink mode) to an emphasis ink mode. Generally, in an emphasis ink mode, ink is temporary and fades and disappears after a period of time. Emphasis ink, for example, is not saved as part of primary content or a transient ink layer, but is used for temporary purposes, such as for visually identifying content, emphasizing content, and so forth.

[0112] Pin control 826—this control is selectable to pin a transient ink layer to a document, and to unpinn the transient ink layer from the document. For instance, selecting the pin control 826 causes transient ink of a transient ink layer to be persistently displayed as part of a document. With reference to the scenario 500, for example, selection of the pin control 826 prevents the ink timer 406 from being initiated when a user removes the pen 128 from proximity to the display device 112.
The pin control 826 is also selectable to unpin a transient ink layer from a document. For instance, with reference to the scenario 500, selection of the pin control 826 unpins a transient ink layer such that the ink timer 406 begins to elapse when a user removes the pen 128 from proximity to the display device 112. In at least some implementations, when the user selects the ink flag 310 to cause the ink menu 802 to be presented, the pin control 826 is presented within the ink menu 802 in the same region of the display in which the user icon 502 is displayed in the ink flag 310. Thus, a user can double-tap on the same spot in the over the ink flag 310 to cause the ink menu to be presented, and to pin and unpin transient ink from the document 304.

In at least some implementations, the visuals presented for the individual controls represent hover targets that are displayed when the respective modes are active. The example implementation scenarios above depict examples of hover targets for transient and permanent ink modes, and similar scenarios apply for other ink modes and the visuals displayed for their respective controls in the ink menu 802.

According to one or more implementations, providing input outside of the ink menu 802 causes the ink menu 802 to collapse. For instance, if the user taps the pen 128 in the GUI 302 outside of the ink menu 802, the ink menu 802 collapses such that the ink flag 310 is again displayed.

FIG. 9 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for processing ink according to a current ink mode in accordance with one or more embodiments.

Step 900 detects a pen in proximity to an input surface. The touch input device 118, for instance, detects that the pen 128 is hovered and/or in contact with the touch input device 118. As referenced above, a hover operation can be associated with a particular threshold proximity to an input surface such that hovering the pen 128 at or within the threshold proximity to the input surface is interpreted as a hover operation, but placing the pen 128 further than the threshold proximity from the input surface is not interpreted as a hover operation.

Step 902 ascertains a current ink mode. The ink module 120, for example, ascertains an ink mode that is currently active on the client device 102. Examples of different ink modes are detailed elsewhere herein, and include a permanent ink mode, a transient ink mode, a text recognition mode, a shape recognition mode, a selection mode, an erase mode, a command mode, and so forth.

In at least some implementations, a current ink mode may be automatically selected by the ink module 120, such as based on an application and/or document context that is currently in focus. For instance, an application 106 may specify a default ink mode that is to be active for the application. Further, some applications may specify ink mode permissions that indicate allowed and disallowed ink modes. A particular application 106, for example, may specify that a permanent ink mode is not allowed for documents presented by the application, such as to protect documents from being edited.

Alternatively or additionally, a current ink mode is user-selectable, such as in response to user input selecting an ink mode from the ink menu 802. For instance, a user may cause a switch from a default ink mode for an application to a different ink mode.

Step 904 causes a visual affordance identifying the current ink mode to be displayed. Examples of such an affordance include a hover target, a visual included as part of an ink flag and/or ink-related menu, and so forth. Examples of different visual affordances are detailed throughout this description and the accompanying drawings.

Step 906 processes ink content applied to the input surface according to the current ink mode. The ink content, for instance, is processed as permanent ink, transient ink, and so forth. For example, if a permanent ink mode is active, the ink content is saved as permanent ink, such as part of a primary content layer of a document. If the transient ink mode is active, the ink content is propagated to a transient ink layer of a document. Examples of different mode-specific ink behaviors and actions are detailed elsewhere herein.

FIG. 10 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for a transient ink timer in accordance with one or more implementations. In at least some implementations, the method represents an extension of the method described above with reference to FIG. 9.

Step 1000 receives ink content applied to a document via input from a pen to an input surface while in a transient ink mode. The ink module 120, for example, processes ink content received from the pen 128 to the display device 112 as transient ink.

Step 1002 detects that the pen is removed from proximity to the input surface. For instance, the touch input device 118 detects that the pen 128 is not in contact with and is not hovering over a surface of the touch input device 118, e.g., the display device 112.

Step 1004 initiates a timer. The timer, for example, is initiated in response to detecting that the pen is removed from proximity to the input surface. In at least some implementations, a visual representation of the timer is presented. For instance, the visual representation provides a visual cue that the timer is elapsing, and indicates a relative amount (e.g., percentage) of the timer that has elapsed. The visual representation, for example, is animated to visually convey that the timer is elapsing. One example of a visual representation of a timer is discussed above with reference to FIGS. 4 and 5.

Step 1006 ascertains whether the pen is detected at the input surface before the timer expires. For instance, the ink module 120 ascertains whether the pen 128 is detected in contact with and/or hovering over the touch input device 118 prior to expiry of the timer. If the pen is detected at the input surface prior to expiry of the timer (“Yes”), step 1008 resets the timer and the process returns to step 1000.

If the pen is not detected at the input surface prior to expiry of the timer (“No”), step 1010 removes the ink content from the document and propagates the ink content to a transient layer for the document. For instance, response to expiry of the timer, the transient ink content is removed from display and propagated to a transient data layer for the document that is separate from a primary content layer of the document. In at least some implementations, a new transient ink layer is created for the document, and the transient ink content is propagated to the new transient ink layer. Alternatively or additionally, the transient ink content is propagated to an existing transient ink layer. For example, the transient ink layer may represent an accumulation of transient ink pro-
vided by a user over multiple different interactions with the document and over a period of time.

[0129] As discussed above, the transient ink layer may be associated with a particular user, e.g., a user that applies the transient ink content to the document. Thus, the transient ink is linked to the particular user and may subsequently be accessed by the user.

[0130] FIG. 11 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for propagating transient ink to different transient ink layers for different users in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above with reference to FIGS. 9 and 10.

[0131] Step 1100 receives transient ink content to a document from multiple different users. The transient ink content, for instance, is received during different interactivity sessions with the document that are individually associated with a different user. The document, for example, a shared among different users, such as part of a group collaboration on the document.

[0132] Step 1102 propagates transient ink content from each user to a different respective transient ink layer for the document. A different transient ink layer, for example, is generated for each user, and transient ink content applied by each user is propagated to a respective transient ink layer for each user.

[0133] Step 1104 causes visual affordances of the different transient ink layers to be displayed. Each transient ink layer, for example, is represented by a visual affordance that visually identifies the transient ink layer and a user linked to the transient ink layer. Examples of such affordances are discussed above with reference to ink flags.

[0134] Step 1106 enables each transient ink layer to be individually accessible. The ink module 120, for example, enables each transient ink layer to be accessed (e.g., displayed) separately from the other transient ink layers. In at least some implementations, a transient ink layer is accessible by selecting a visual affordance that represents the transient ink layer. Further, multiple transient ink layers may be accessed concurrently, such as by selecting visual affordances that identify the transient ink layers.

[0135] While implementations are discussed herein with reference to the display of transient ink layers, it is to be appreciated that a transient ink layer may be accessible in various other ways and separately from a document to which the transient ink layer is bound. For instance, a transient ink layer may be printed, shared (e.g., emailed) separately from a document for which the transient ink layer is created, published to a website, and so forth.

[0136] FIG. 12 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for presenting an ink menu in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above.

[0137] Step 1200 detects an action to invoke an ink menu. The ink module 120, for example, detects that a user requests an ink menu. For instance, as described in the scenario 800, a user may select a visual control (e.g., the ink flag 310) to request an ink menu. Alternatively or additionally, an ink menu can be automatically invoked in response to various events, such as the pen 128 detected in proximity to the surface of the display surface, an ink-related application and/or service being launched, an application and/or service querying a user to select an ink mode, and so forth.

[0138] Step 1202 causes the ink menu to be presented. The ink module 120, for example, causes the ink menu 802 to be displayed. In at least some implementations, a default set of functionalities is associated by the ink module 120 with the ink menu 802. Accordingly, different applications may modify the default set of functionalities, such as by adding a functionality to the default set of functionalities, removing a functionality from the default set of functionalities, and so forth. Thus, according to one or more implementations, the ink menu 802 is populated with a set of functionalities (e.g., selectable controls) based on a customized set of functionalities specified by and/or for an application that is currently in focus.

[0139] Step 1204 receives user input to the ink menu. For example, the ink module 120 detects that a user manipulates the pen 128 to select a control displayed as part of the ink menu 802.

[0140] Step 1206 performs an action in response to the user input. The ink module 120, for instance, causes an action to be performed based on which control is selected by the user. Examples of such actions include changing an ink mode, initiating ink playback, applying different ink formatting, and so forth.

[0141] FIG. 13 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for ink playback in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above.

[0142] Step 1300 records ink content added to a document. The ink module 120, for example, records ink content as it is generated in the document, such as in real-time. For instance, a time-aligned record of ink content is generated that correlates ink content with a particular time at which the ink content is added to a document. In an example implementation, frames of ink content are generated that represent a timeline of ink content applied to a document over a period of time. In a transient ink implementation, the ink content is recorded as part of a transient ink layer for the document that is separate from primary content of the document.

[0143] In at least some implementations, the ink module 120 maintains records of different ink content for different documents and different users. Thus, ink content can be retrieved and played back on a per-user and/or per-document basis.

[0144] Step 1302 receives a request to playback the ink content. A user, for instance, requests playback of ink content, such as via selection of the playback control 804 presented as part of the ink menu 802.

[0145] Step 1304 causes playback of the ink content. For example, the ink module 120 receives the playback request, and initiates playback of the ink content. The ink playback may take a variety of different forms. For instance, the ink content may be played back on the display device 112 as an animation that reproduces the ink content in a manner in which it was originally applied. In at least some implementations, transient ink content from a transient ink layer can be played back within a document to which the transient ink layer is bound. Alternatively or additionally, transient ink
content can be played back separately from a linked document, such as part of a separate playback experience.

According to various implementations, playback of ink content represents a real-time playback that reflects characteristics of the original application of the ink content, such as a rate at which the ink content was originally applied. Further, ink content playback can be manipulated in various ways, such as a rewind of ink content, a fast forward of ink content, skipping between different sections of ink content, and so forth.

FIG. 14 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for inserting ink content in an ink playback mode in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above.

Step 1400 detects that new ink content is added while a playback mode for stored ink content is active. For instance, while playback of previously-recorded ink content within a document is in progress as described above, a user applies new ink content within document. Alternatively or additionally, a user pauses and/or stops playback of previously-recorded ink content, and adds new ink content while the playback is paused and/or stopped.

Step 1402 inserts the new ink content into the stored ink content. The ink module 120, for example, inserts the new ink content into the stored ink content at a temporal point at which the new ink content was applied during playback of the stored ink content. Thus, the new ink content becomes integrated into the stored ink content such that the new ink content is presented during a subsequent playback of the stored ink content.

As an alternative or addition to insertion of new ink content, portions of recorded ink content can be removed during playback. For instance, a user can use an erase functionality to erase portions of stored ink content such that during a subsequent playback of the stored ink content, the erased portions are not presented.

The next portion of this discussion presents example implementation scenarios and procedures for ink versioning in accordance with various implementations. Generally, ink versioning relates to the ability to discern an ink annotation as it relates to a document which it annotates. If a document changes (e.g., in length, size, content, and so on), techniques for ink versioning seek to maintain a correlation between an ink annotation and a portion of a document to which the annotation relates.

FIG. 15 depicts an example implementation scenario 1500 for ink versioning in accordance with one or more implementations. The upper portion of the scenario 1500 includes a GUI 1502 with a document 1504 displayed on the display device 112. Generally, the GUI 1502 represents a GUI for a particular functionality, such as an instance of the applications 106. The document 1504 includes primary content 1506, which in this example includes text content.

In the upper portion of the scenario 1500, a user provides transient ink annotation to the document 1504 in a transient ink mode. The transient ink annotation includes an annotation 1508a and an annotation 1508b, which specify various proofreading suggestions for editing the primary content 1506. For instance, the annotation 1508a suggests increasing font size of the primary content 1506, and the annotation 1508b suggests adding content at a particular point in the primary content 1506.

Proceeding to the lower portion of the scenario 1500, a user applies edits 1510 to the document 1504 to increase a font size of the primary content 1506 and to insert a sentence at a point identified by the annotation 1508b. As illustrated, the edits 1510 cause changes to the primary content 1506 such that the primary content is reflowed (e.g., reformatted) within the document 1504.

Further, based on the edits 1510, the annotation 1508a is no longer relevant since the font size has been increased, and the annotation 1508b does not retain its original spatial context indicated in the upper portion of the scenario 1500. For instance, after the edits 1510 the annotation 1508b points to a portion of the primary content 1506 that is not pertinent to the suggestion of the annotation 1508a. Thus, the edits 1510 have caused the annotations 1508a, 1508b to lose their context within the document 1504. Accordingly, an annotation layer 1512 is generated for the document 1504. Generally, the annotation layer 1512 represents a snapshot of the document 1504 including the primary content 1506 and the annotations 1508a, 1508b as illustrated in the upper portion of the scenario 1500 and before the edits 1510 are applied.

According to various implementations, a threshold change in a document is specified that when met and/or exceeded, causes the annotation layer 1512 to be generated. The threshold change can be specified in various ways, such as a threshold distance between the annotation 1508a and adjacent text of the primary content 1506. For example, when the annotation 1508b is initially generated as in the upper portion of the scenario 1500, the annotation 1508b is adjacent to a portion of text 1514 of the primary content 1506. However, application of the edits 1510 causes the portion of text 1514 to be moved elsewhere within the document 1504 such that the portion of text 1514 is no longer adjacent to the annotation 1508b. In the scenario 1500, the portion of text 1514 has moved a threshold distance from the annotation 1508b such that the annotation layer 1512 is generated. The threshold distance may be specified in various ways, such as in pixels, distance on a display (e.g., in millimeters, centimeters, and so on), distance with reference to sections of the document 1504, and so forth.

A threshold change in a document may be specified in various other ways, such as a particular format change for primary content, a number of format changes for primary content, an insertion and/or deletion of a specified portion of primary content, and so forth. The scenario 1600 discussed below provides further details concerning the annotation layer 1512.

FIG. 16 depicts an example implementation scenario 1600 for ink versioning in accordance with one or more implementations. The scenario 1600, for example, represents a continuation of the scenario 1500, above. The upper portion of the scenario 1500 includes the GUI 1502 with the document 1504 (introduced above) displayed on the display device 112. The version of the document presented in the upper portion of the scenario represents the document 1504 with the edits 1510 applied to the primary content 1506, such as depicted in the lower portion of the scenario 1500. Notice further that the annotations 1508a, 1508b are removed. According to various implementations, when the annotation layer 1512 is generated, existing annotations of the document
1504 are saved to the annotation layer 1512 and removed from display as part of the primary content 1506.

[0159] Further presented in the upper portion of the scenario 1600 is a layer flag 1602, which represents a visual cue that the document 1504 includes an annotation layer, e.g., the annotation layer 1512. The layer flag 1602 includes a user icon 1604 and a layer count 1606. The user icon 1604 represents a particular user associated with the annotation layer 1512, e.g., a user whose interaction with the document 1504 caused the annotation layer 1512 to be generated. The layer count 1606 specifies a number of annotation layers linked to the document 1504. The number of annotation layers identified by the layer count 1606 may correspond to a number of annotation layers generated by the user associated with the user icon 1604, or may correspond to an aggregate number of annotation layers linked to the document 1504 and for multiple users.

[0160] Notice that in the scenario 1600, the layer flag 1602 is displayed in a different region of the GUI 1502 than is an ink flag 1608. The layer flag 1602, for instance, is pinned to an outer edge of the GUI 1502, whereas the ink flag 1608 is pinned to an edge of the document 1504. According to various implementations, this difference in positioning provides a further visual cue that the document 1504 is linked to one or more annotation layers.

[0161] Proceeding to the lower portion of the scenario 1600, a user selects the layer flag 1602 (e.g., using the pen interface and/or other input technique), which causes the annotation layer 1512 to be displayed in the GUI 1502. For instance, the snapshot of the document 1504 stored as part of the annotation layer 1512 is retrieved and displayed. In this particular implementation, the annotation layer replaces the current version of the document 1504 in the GUI 1502. As discussed below and illustrated with reference to FIG. 17, however, an annotation layer may be illustrated along with a current version of a document.

[0162] The scenario 1600 further depicts that the layer flag 1602 is attached to an edge of the annotation layer 1512, which provides a visual affordance that the annotation layer 1512 is displayed. For instance, selection of the layer flag 1602 in the upper portion of the scenario 1600 causes the layer flag 1602 and the ink flag 1608 to switch positions in the GUI 1502. Further, the layer count 1606 is reduced (e.g., to “2”) to indicate that other annotation layers for the document 1504 are available. For instance, a further selection of the layer flag 1602 while the annotation layer 1512 is displayed will cause a different annotation layer to be retrieved and displayed. A user, for example, can scroll through multiple annotation layers by selecting the layer flag 1602 multiple times. In at least some implementations, the annotation layers are not editable and represent static representations of historical versions of the document 1504.

[0163] According to one or more implementations, a user may return to a current version of the document 1504 by selecting the ink flag 1608 while the annotation layer 1512 is displayed. For instance, selection of the ink flag 1608 in the lower portion of the scenario 1600 causes a return to the upper portion of the scenario 1600.

[0164] FIG. 17 depicts an example implementation scenario 1700 for presenting a current version of a document along with an annotation layer in accordance with one or more implementations. The scenario 1700, for example, represents a variation of the scenario 1600, above. The scenario 1700 includes the GUI 1502 with the document 1504 (introduced above) displayed on the display device 112. Generally, the document 1504 represents a current version of the document 1504 (e.g., a “live” version), and a user may interact with document 1504 to edit the primary content 1506 in various ways. The document 1504 includes the ink flag 1608, which provides a visual affordance that the document 1504 is a current version, and that the user may edit the document 1504 as well as provide ink annotations to the document 1504.

[0165] Displayed to the right of the document 1504 in the GUI 1502 is the annotation layer 1512, details of which are described above. The annotation layer 1504 includes the layer flag 1602 with the layer count 1606 indicating that 2 other annotation layers are available to be accessed in addition to the annotation layer 1512. As referenced above, the layer flag 1602 is selectable to cause other annotation layers to be displayed, such as to navigate through different annotation layers.

[0166] According to various implementations, if a user edits the document 1504 such that the edits exceed or exceed a threshold change, a new annotation layer will be created and added to the group of annotation layers to the right. Thus, the scenario 1700 illustrates that a current version of a document and an annotation layer of a document may be displayed together, e.g., side-by-side. In at least some implementations, annotation layers are grouped (e.g., stacked) in a chronological order according to a date and time on which they were created. A user is thus provided with an edit history and an annotation history for a document, which gives the user context to when and why certain changes to a document occurred.

[0167] FIG. 18 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for ink versioning in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above.

[0168] Step 1800 monitors for changes to primary content of a document that includes an ink annotation. The ink module 120, for example, monitors a state of a document for edits that cause changes to the document.

[0169] Step 1802 ascertains that a threshold change occurs to the primary content. Generally, the threshold change refers to a change to primary content of the document. The threshold change, for instance, refers to a difference between an edited document and a version of the document prior to the edits being applied. As referenced above, a threshold change can be specified based on various factors, such as an amount of content that is added or removed, a threshold movement of content (such as measured in pixel displacement, distance moved, and so forth), a number of lines of text that are moved, a threshold distance between an ink annotation and primary content to which the ink annotation refers, and so forth.

[0170] Step 1804 captures an annotation layer for the document that includes the annotation and a version of the primary content prior to the threshold change. For instance, the annotation layer is captured and saved responsive to ascertaining that the threshold change occurs. The annotation layer, for example, represents a snapshot of the document as it appears prior to the threshold change being applied to the document.

[0171] Generally, an annotation layer can be captured in various ways. For instance, as a document is being annotated and edited, cached versions of the document can be periodically saved. When edits to the document exceed the threshold...
change, a cached version of the document from prior to the threshold change is saved as an annotation layer that is linked to the document. Alternatively or additionally, when edits to a document are received (e.g., via user input) that cause a threshold change to the document, an annotation layer for the document is captured prior to the edits being applied.

Step 1806 enables the annotation layer to be accessed separately from a current version of the document. The annotation, for instance, is removed from the current version of the document and is saved as part of the annotation layer and can be accessed along with the annotation layer. Example ways of enabling access to an annotation layer are presented above and below.

FIG. 19 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for enabling access to an annotation layer in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above.

Step 1900 receives a request to present an annotation layer of a document. The ink module 120, for example, receives a user request to view an annotation layer for a document, such as while a current version of the document is displayed. As referenced above, an annotation layer may be presented in response to selection of the layer flag 1602.

Step 1902 causes the annotation layer to be presented. For instance, the ink module 120 retrieves the annotation layer and causes the annotation layer to be displayed. As discussed above, the annotation layer can replace a current version of the document in a display region, or can be presented along with the current version of the document in a display region. As further discussed above, multiple annotation layers may be generated for a document, and thus the multiple annotation layers may be individually accessible and/or accessed as a group.

The next portion of this discussion presents example implementation scenarios and an example procedure for ink notes in accordance with various implementations. Generally, ink notes provide ways of preserving ink as notes that can be saved, shared, and accessed in various ways.

FIG. 20 depicts an example implementation scenario 2000 for ink notes in accordance with one or more implementations. The scenario 2000, for example, represents a continuation of the scenarios described above. The upper portion of the scenario 2000 includes the GUI 302 with the document 304 and the ink menu 802 (introduced above) displayed on the display device 112.

In the upper portion of the scenario 2000, a user selects the ink note control 822. For instance, the user taps the ink note control 822, and/or drags the ink note control 822 from the ink menu 802 into the body of the document 304.

Proceeding to the lower portion of the scenario 2000, and responsive to the user selection of the ink note control 822, an ink note 2002 is presented in the GUI 302. Generally, the ink note 2002 represents an electronic canvas on which notes can be applied using ink. A user then applies ink content 2004 to the ink note 2002.

In at least some implementations, the scenario 2000 occurs while the GUI 302 is in a transient ink mode. Accordingly, ink content applied to the document 304 itself will behave according to the transient ink mode. However, ink content applied within the ink note 2002 behaves according to an ink note mode. Thus, the ink note 2002 represents a separate inking environment from the document 304, and thus different behaviors apply to the ink content 2004 than to ink content within the document 304.

The ink note 2002 includes a save control 2006 and a share control 2008. According to various implementations, selecting the save control 2006 causes the ink content 2004 to be saved to a particular location, such as a pre-specified data storage location. In at least some implementations, a single selection of the save control 2006 causes the ink content 2004 to be saved and the ink note 2002 to be removed from display such that a user may return to interacting with the document 304.

The share control 2008 is selectable to share the ink content 2004, such as with another user. For instance, selecting the share control 2008 causes the ink content 2004 to be automatically propagated to a message, such as the body of an email message, an instant message, a text message, and so forth. A user may then address and send the message to one or more users. Alternatively or additionally, selecting the share control 2008 may cause the ink content 2004 to be posted to a web-based venue, such as a social networking service, a blog, a website, and so forth. According to various implementations, functionality of the share control 2008 is user-configurable such that a user may specify behaviors caused by selection of the share control 2008.

FIG. 21 depicts an example implementation scenario 2100 for ink notes in accordance with one or more implementations. The scenario 2100, for example, represents a continuation and/or variation of the scenarios described above. The upper portion of the scenario 2100 includes the GUI 302 with the document 304 and the ink menu 802 (introduced above) displayed on the display device 112.

In the upper portion of the scenario 2100, a user applies ink content 2102 to the document 302 and then applies a selection action 2104 to the ink content 2102. In this particular example, the selection action 2104 is implemented as inking a closed loop around the ink content 2102. It is to be appreciated, however, that a wide variety of other selection actions may be employed in accordance with various embodiments.

Proceeding to the lower portion of the scenario 2100, the user selects the ink note control 822, such as by tapping on the ink note control 822 and/or dragging the ink note control 822 out of the ink menu 802. In response, an ink note 2106 is automatically generated and populated with the ink content 2102. The ink module 120, for example, detects that the ink content 2102 is selected via the selection action 2104, and thus populates the ink content 2102 to the ink note 2106. Thus, the selection action 2104 followed by the selection of the ink note control 822 is interpreted as a command to generate the ink note 2106 and populate the ink note 2106 with the ink content 2102.

In this particular example, the ink content is moved (e.g., cut and paste) from the body of the document 304 into the ink note 2106. In some alternative implementations, the ink content is copied into the ink note 2106 such that the ink content 2102 remains in the body of the document 304. The user may then save the ink note 2106 by selecting the save control 2006, and may share the ink content 2102 by selecting the share control 2008. Example attributes and actions of the save control 2006 and the share control 2008 are described above.

While the scenario 2100 is discussed with reference to populating the ink content 2102 to the ink note 2106, it is to
be appreciated that a wide variety of other content may be populated to the ink note 2106. For instance, a user may select a portion of the primary content 306 from the document 304 (e.g., text content), and a subsequent selection of the ink note control 822 would cause the ink note 2102 to be generated and populated with the selected primary content. As another example implementation, a combination of ink content and primary content can be selected, and a subsequent selection of the ink note control would cause the ink note 2102 to be generated and populated with both the selected ink content and primary content.

[0188] The scenarios described above generally describe that ink note functionality is invocable via selection of the ink note control 822. In additional or alternative implementations, ink note functionality is invocable in other ways, such as in response to a dragging gesture from anywhere within the ink menu 802 into the body of the document 304, a custom gesture applied anywhere within the GUI 302, a gesture involving a combination of finger touch input and pen input, a voice command, a touchless gesture, and so forth.

[0189] Thus, the scenario 2100 illustrates that techniques discussed herein reduce the number of user interactions required to propagate content to a note (e.g., an ink note), since a user may simply select existing content and invoke an ink note functionality to populate the existing content to an ink note.

[0190] FIG. 22 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for generating an ink note in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above.

[0191] Step 2200 detects a user selection of content. The ink module 120, for example, ascertain that a user selects a portion of ink content, primary content, combinations thereof, and so forth.

[0192] Step 2202 ascertains that an ink note functionality is invoked. Various ways of invoking ink note functionality are detailed above.

[0193] Step 2204 populates the selected content to an ink note. For example, the ink module 120 generates an ink note and populates (e.g., copies or moves) the selected content to the ink note. In at least some implementations, the ink note is generated and the selected content is populated to the ink note automatically and in response to a single user invocation of ink note functionality, e.g., a single user action.

[0194] Step 2206 performs an action in relation to the ink note in response to user input. For instance, a user provides input that causes the ink note to be saved, to be shared, to be deleted, and so forth. Examples of user input include user selection of a selectable control, a user applying an ink and/or touch gesture, input via an input mechanism 114, and so forth.

[0195] The next portion of this discussion presents an example implementation scenario and procedure for using ink for emphasis in accordance with various implementations. Generally, ink for emphasis provides ways of using ink for temporary purposes, such as for identifying content, highlighting content, for temporary communication, and so forth.

[0196] FIG. 23 depicts an example implementation scenario 2300 for ink for emphasis in accordance with one or more implementations. The scenario 2300, for example, represents a continuation and/or variation of the scenarios described above. The upper portion of the scenario 2300 includes a GUI 2302 with primary content 2304 and the ink menu 802 (introduced above) displayed on the display device 112.

[0197] Further to the scenario 2300, a user applies emphasis ink 2306 to identify and/or emphasize a portion of the primary content 2304. The user, for instance, selects the emphasis control 824, which causes a transition to an emphasis ink mode. The user then applies the emphasis ink 2306 while in the emphasis ink mode. Notice that when the pen 128 is hovered over the display device 112, a hover target 2308 is presented that identifies the emphasis ink mode. As referenced above, different hover targets can be displayed that identify a particular ink mode that is active, such as utilizing visuals based on the respective controls presented in the ink menu 802.

[0198] Proceeding to the lower portion of the scenario 2300, the user removes the pen 128 from proximity to the display device 112. In response to detecting that the pen is not in proximity to the display device 112, the emphasis ink 2306 fades and disappears from the GUI 2302. For instance, as long as the pen 128 is detected in proximity to the display device 112, the emphasis ink 2306 remains displayed. However, when the pen 128 is not detected in proximity to the display device 112, a first timer begins running while the emphasis ink 2306 is displayed. When the first timer expires, a second timer begins running during which the emphasis ink 2306 fades from the GUI 2302. When the second timer expires, the emphasis ink 2306 disappears, e.g., is removed from display.

[0199] FIG. 24 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for ink for emphasis in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above.

[0200] Step 2400 receives ink content in an emphasis ink mode. The ink module 120, for example, detects that a user applies ink content while an emphasis ink mode is active. According to various implementations, an emphasis mode is activatable by a user, such as in response to selection of the emphasis control 824.

[0201] Step 2402 initiates a timer for the ink content. For instance, one or more timers are initiated while the ink content is displayed. The timers, for example, are specific to the emphasis ink mode. In at least some implementations, the timer is initiated in response to detecting that the pen 128 is not detected in proximity to the display device 112, such as after a user applies the ink content using the pen 128.

[0202] Step 2404 removes the ink content when the timer expires. The ink content, for instance, fades and is automatically removed from display in response to expiry of one or more timers and independent of user input to remove the ink content.

[0203] The next portion of this discussion presents example implementation scenarios and an example procedure for ink and touch in accordance with various implementations. Generally, ink and touch provides ways of combining ink with touch input to provide for diverse input scenarios. In at least
some implementations, ink content applied in the different scenarios is applied in a permanent ink mode, e.g., as permanent ink.

[0204] FIG. 25 depicts an example implementation scenario 2500 for ink and touch in accordance with one or more implementations. The scenario 2500, for example, represents a continuation and/or variation of the scenarios described above. The upper portion of the scenario 2500 includes a GUI 2502 with primary content 2504 displayed on the display device 112.

[0205] Further depicted in the upper portion of the scenario 2500 is that a user manipulates the pen 128 to apply ink to draw a freehand line 2506 within the GUI 2502. Notice that the freehand line 2506 is not straight but includes some curvature, such as due to variations in movement of the pen 128 while applying the freehand line 2506. While not expressly illustrated here, it is to be appreciated that a hover target may be displayed when the pen 128 is hovered over the display 112 to indicate a currently-active ink mode, examples of which are described above.

[0206] Proceeding to the lower portion of the scenario 2500, the user applies touch input to the freehand line 2506 using a finger 2508, which causes the freehand line 2506 to snap to a straight line 2510. For instance, touch input to the freehand line 2506 is interpreted as a command to convert the freehand line 2506 to the straight line 2510. Generally, the straight line 2510 represents a machine-encoded line with null curvature.

[0207] FIG. 26 depicts an example implementation scenario 2600 for ink and touch in accordance with one or more implementations. The scenario 2600, for example, represents a continuation of the scenario 2500 described above. The upper portion of the scenario 2600 includes the GUI 2502 with primary content 2504 displayed on the display device 112.

[0208] In the upper portion of the scenario 2600, the user manipulates the pen 128 to continue to apply ink to extend the straight line 2510 while maintaining the finger 2508 in contact with the display device 112 over the straight line 2510. Notice that as the pen 128 applies additional ink, the straight line 2510 remains anchored at the finger 2508. For instance, the continued touch input by the finger 2508 is interpreted as a command to continue straightening the straight line 2510 as additional ink is applied. Thus, even if the pen 128 deviates from drawing a straight line, the straight line 2510 will be drawn as a straight line between a contact point of the finger 2508 and a tip of the pen 128.

[0209] Proceeding to the lower portion of the scenario 2600, the user moves the pen 128 upward in the GUI 2502 while maintaining the finger 2508 in contact with the display device 112. Accordingly, the straight line 2510 pivots about the contact point of the finger 2508. Thus, maintaining the finger 2508 in contact with the display device 112 while drawing the straight line 2510 causes the contact point of the finger 2508 to function as a pivot point for the straight line 2510.

[0210] FIG. 27 depicts an example implementation scenario 2700 for ink and touch in accordance with one or more implementations. The scenario 2700, for example, represents a continuation of the scenarios 2500, 2600 described above. The upper portion of the scenario 2700 includes the GUI 2502 with primary content 2504 and the straight line 2510 displayed on the display device 112. Further depicted is that the user removes from the pen 128 from proximity to the display device 112.

[0211] Proceeding to the lower portion of the scenario 2700, the user applies a multi-finger gesture 2702 to the display device 112 while maintaining the finger 2508 in contact with the straight line 2510. In response to the multi-finger gesture 2702, the straight line 2510 snaps to a position parallel and/or an upper border 2704 of the GUI 2502. For instance, the straight line 2510 pivots about the contact point of the finger 2508 to a position in which the straight line 2510 is parallel to the upper border 2704. The two-finger gesture, for example, is interpreted as a command to automatically reposition the straight line 2510 to a particular angle, e.g., an angle relative to the upper border 2704.

[0212] In this particular example, the multi-finger gesture 2702 is implemented with two fingers, but it is to be appreciated that different numbers and combinations of fingers may be employed in accordance with the claimed embodiments.

[0213] FIG. 28 depicts an example implementation scenario 2800 for ink and touch in accordance with one or more implementations. The scenario 2800, for example, represents a continuation of the scenarios 2500-2700 described above. The upper portion of the scenario 2800 includes the GUI 2502 with primary content 2504 and the straight line 2510 displayed on the display device 112. Further depicted is that the user places the finger 2508 in contact with the display device 112 on the straight line 2510. The straight line 2510, for instance, is in a position parallel to the upper border 2704 such as discussed with reference to the scenario 2700.

[0214] Proceeding to the lower portion of the scenario 2800, the user taps a second finger 2802 on the display device 112, which causes the straight line 2510 to pivot about the finger 2508 to a position perpendicular to the upper border 2704. The straight line 2510, for example, pivots 90° about the contact point of the finger 2508. Some portions of the straight line 2510 are shown as a dashed line to indicate a portion of the straight line 2510 underneath the user’s hand.

[0215] While the scenario 2800 is discussed with reference to a 90° rotation in response to the tap input, it is to be appreciated that implementations may utilize other increments of rotations, such as 22.5°, 45°, and so forth.

[0216] Accordingly, the scenario 2800 shows that a multi-finger gesture combined with a finger tap or other input can be used to cause a shape to move and/or rotate in various ways. For instance, if the user again taps the second finger 2802 on the display device 112 while holding the finger 2508 on the straight line 2510, the straight line will again pivot about the finger 2508. Thus, the user may cause the straight line 2510 to pivot about the finger 2508 to different positions by repeatedly tapping the second finger 2802 while holding the finger 2508 at a particular position on the straight line 2510.

[0217] FIG. 29 depicts an example implementation scenario 2900 for ink and touch in accordance with one or more implementations. The scenario 2900, for example, represents a continuation and/or variation of one or more of the scenarios described above. The upper portion of the scenario 2900 includes the GUI 2502 with primary content 2504 displayed on the display device 112. Further depicted is that the user manipulates the pen 128 to apply ink within GUI 2502 to draw a freehand loop 2902. Notice that the freehand loop 2902 includes some variation in curvature, e.g., is not a true circle.
Proceeding to the lower portion of the scenario 2900, the user applies a multi-finger gesture 2904 to the display device 112 on the freehand loop 2902, which causes the freehand loop 2902 to be converted to a machine-encoded circle 2906. The multi-finger gesture 2904, for example, is interpreted as a command to convert the freehand loop 2902 to the circle 2906.

Thus, the above-described scenarios present example implementations in which techniques discussed herein may employ an interplay between touch and pen input to perform various actions, such as to convert freehand ink input into machine-encoded shapes, and to manipulate shapes in various ways. The particular shapes and gestures included in the scenarios are presented for purpose of example only, and it is to be appreciated that implementations for ink and touch can employ a variety of different gestures and shapes not expressly discussed herein. In at least some implementations, ink content applied in the scenarios for ink and touch described above is applied in one or more of a permanent ink layer or a translucent ink layer of an associated document/GUI.

FIG. 30 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for ink and touch in accordance with one or more implementations. In at least some implementations, the method represents an extension of the methods described above. According to one or more implementations, the method represents operations of the client device 102, such as performed by the ink module 120 and/or other functionalities of the client device 102. For example, the method represents an example procedure for performing the implementation scenarios for ink and touch described above.

Step 3000 detects that freehand ink content is applied to a display via a pen. The ink module 120, for instance, detects that ink content is applied to the display device 112 with the pen 128. Generally, the ink content represents freehand ink applied to the display device 112.

Step 3002 identifies a touch gesture that is applied to the display. The touch gesture, for instance, involves one or more fingers applied to the display.

In at least some implementations, the touch gesture may be static, such as placing or tapping of a finger on the display. Alternatively or additionally, the touch gesture may involve movement across the display, such as to generate a particular gesture motion. In one or more implementations, the touch gesture is applied at least partially to the ink content.

Step 3004 maps the touch gesture to a particular operation to be performed on the ink content. For instance, different gestures are pre-specified to be mapped to different respective operations.

Step 3006 modifies the ink content by applying the particular operation to the ink content. For instance, freehand ink content is converted into a machine-encoded shape, such as a line, a circle, a triangle, a quadrilateral, and so forth. As another example, the ink content is modified by being moved within the display in response to the touch gesture, such as via rotation, translation, and so forth. Generally, different operations specify different visual transformations that are applicable to the ink content.

According to various implementations, the ink content is modified independently of further input from the user after the touch gesture. Thus, techniques discussed herein provide for simplified ways of interacting and transforming ink content by reducing a number of user interactions required to modify ink content.

The next portion of this discussion presents example implementation scenarios and an example procedure for using ink for document reconfiguration in accordance with various implementations. Generally, ink for document reconfiguration provides ways of using ink to modify document formats, such as for creating space between visual objects included in a document.

FIG. 31 depicts an example implementation scenario 3100 for ink for adding space in accordance with one or more implementations. The scenario 3100, for example, represents a continuation and/or variation of the scenarios described above. The upper portion of the scenario 3100 includes a GUI 3102 with an electronic document 3104 displayed on the display device 112. The GUI 3102 generally represents a GUI for an application, such as an instance of the applications 106.

Further depicted in the upper portion of the scenario 3100 is that a user applies ink using the pen 128 to draw a dashed line 3106 through the document 3104. While not expressly illustrated here, it is to be appreciated that a hover target may be displayed when the pen 128 is hovered over the display 112 to indicate a currently-active ink mode, examples of which are described above.

Proceeding to the lower portion of the scenario 3100 and responsive to the dashed line 3106 being drawn, the dashed line 3106 is converted into a document divider 3108 and handles 3110a and 3110b are presented. Further, the document 3104 is divided into document portions 3112a and 3112b that are divided by the document divider 3108. For example, drawing the dashed line 3106 is recognized (e.g., by the ink module 120) as a command to convert the dashed line 3106 into the document divider 3108. As further detailed below, the handles 3110a, 3110b represent visual controls that can receive input to add space between the document portions 3112a, 3112b.

In at least some implementations, the user holds the pen mode button 130 while drawing the dashed line 3106. For instance, selecting and/or holding the pen mode button 130 activates a functionality of the ink module 120 such that the dashed line 3106 is interpreted as a command to create the document divider 3108.

FIG. 32 depicts an example implementation scenario 3200 for ink for adding space in accordance with one or more implementations. The scenario 3200, for example, represents a continuation of the scenario 3100 described above. In the upper portion of the scenario 3200, a user selects and drags the handle 3110b to the right within the GUI 3102. Accordingly, the document 3104 splits at the document divider 3108 such that a space 3202 is created between the document portions 3112a, 3112b. For instance, the space 3202 in inserted between the document divider 3108 and a boundary 3204 of the document portion 3112b. Content within the document portion 3112a, for example, moves to the right within the document 3104 such that the space 3202 is inserted between content of the document portion 3112a and content of the document portion 3112b. Similarly, the user may drag the handle 3110a to the left to create space (e.g., additional space) between the document portions 3112a, 3112b. Generally, the size of the space 3202 depends on and/or is proportional to a distance that the user drags a particular document portions, e.g., is dependent on a size of a drag gesture.
Further, while these particular scenarios are described with reference to interaction with the handles 3110a, 3110b, it is to be appreciated that in at least some implementations, the user may provide drag input anywhere within the document 3104 on either side of the document divider 3108 to insert space between the document portions 3112a, 3112b. The handles 3110a, 3110b, for instance, represent visual affordances to provide visual cues that the document portions 3112a, 3112b are manipulateable to insert space.

Proceeding to the lower portion of the scenario 3200, the user releases the handle 3110b, which causes the handles 3110a, 3110b to be removed from display along with the document divider 3108 and the boundary 3204. Thus, the space 3202 is integrated into the document 3104 such that the user may then provide content (e.g., ink content) into the portion of the document 3104 added via the space 3202.

While these scenarios are discussed with added space at a particular portion of a document, it is to be appreciated that space can be added at any particular portion of a document. For instance, with reference to the document 3104, the user may generate a document divider that divides the document 3104 into horizontal portions, diagonal portions, and so forth. Further, a user may add multiple document dividers such that multiple spaces may be added at different regions of the document 3104.

Thus, these scenarios describe ink input can be used to reconfigure a layout of a document, such as to add space between regions of a document.

FIG. 33 is a flow diagram that describes steps in a method in accordance with one or more embodiments. The method, for instance, describes an example procedure for ink for document reconfiguration in accordance with one or more implementations. In at least some implementations, the step represents an extension of the methods described above. According to one or more implementations, the step represents operations of the client device 102, such as performed by the ink module 120 and/or other functionalities of the client device 102. For example, the step represents an example procedure for performing the implementation scenarios for ink for document reconfiguration described above.

Step 3300 detects that an ink line is applied to a document displayed on a display device via a pen. Generally, the ink line represents a freehand line applied via the pen 128. In at least some implementations, the ink line is not a straight line but includes some curvature and/or variation in linearity. The ink line, for instance, is a dashed line that has a particular meaning to the ink module 120.

Step 3302 converts the ink line into a document divider that divides the document into a first portion on a first side of the document divider, and a second portion on a second side of the document divider. The document divider, for instance, divides the document into different portions that are separately manipulateable to reconfigure the document. In at least some implementations, a visual affordance is displayed that provides a visual cue that one or more of the first portion or the second portion are manipulateable to reconfigure the document. One example of such as visual affordance are the handles 3110a, 3110b described above.

Step 3304 receives input to one or more of the first portion or the second portion of the document. For instance, a user provides input to the first portion or the second portion, such as a drag gesture. In at least some implementations, the input is provided to a selectable control for manipulating the different portions of the document, such as to one or more of the handles 3110a, 3110b.

Step 3306 causes a space to be inserted between the first portion and the second portion in response to the input. For instance, space is added to the document between the first portion and the second portion of the document. In at least some implementations, the amount of space that is added is proportional to the input, such as to a size of a gesture (e.g., a drag operation) that is provided by a user. Thus, the space becomes part of the document such that content (e.g., ink content) can be added within the space.

The next portion of this discussion presents example implementation scenarios and procedures for ink and maps in accordance with various implementations. Generally, ink and maps provide ways of using ink to interact with maps.

FIG. 34 depicts an example implementation scenario 3400 for ink and maps in accordance with one or more implementations. The scenario 3400, for instance, represents a continuation and/or variation of the scenarios described above. The upper portion of the scenario 3400 includes a map GUI 3402 displayed on the display device 112. The map GUI 3402, for instance, is presented by an instance of the application 106. The map GUI 3402 includes a representation of a particular geographical area and includes several travel paths and locations within the geographical area. For instance, the travel paths include a highway 3404 and a highway 3406. The locations include a home location 3408 and an airport location 3410.

Proceeding to the lower portion of the scenario 3400, a user manipulates the pen 128 to apply ink to trace a travel route 3412 along the highways 3404, 3406 from the home location 3408 to the airport location 3410. As further detailed below, tracing the travel route 3412 causes various actions to be invoked. While not expressly illustrated here, it is to be appreciated that a hover target may be displayed when the pen 128 is hovered over the display 112 to indicate a currently-active ink mode, examples of which are described above.

FIG. 35 depicts an example implementation scenario 3500 for ink and maps in accordance with one or more implementations. The scenario 3500, for instance, represents a continuation and/or variation of the scenario 3400 described above. The scenario 3500 includes the map GUI 3402 displayed on the display device 112. Presented within the map GUI 3402 is a navigation window 3502 that includes various information and actions based on the travel route 3412 generated in the scenario 3400. In at least some implementations, the navigation window 3502 is automatically generated (e.g., by the map module 108) in response to the user tracing the travel route 3412 with the pen 128. For instance, when the user lifts the pen 128 from the display device 112 after tracing the travel route 3412, the map module 108 recognizes that the travel route 3412 starts at the home location 3408 where the pen 128 was initially placed on the display device 112, and that the travel route 3412 ends at the airport location 3410 where the pen 128 was lifted from the display. Accordingly, and independent of further user input after tracing the travel route 3412, the map module 108 aggregates information about the travel route 3412 and presents the navigation window 3502 in the map GUI 3402.

As illustrated, the navigation window 3502 identifies the home location 3408 and the airport location 3410, a distance between the locations, and presents selectable
options for obtaining additional information about the travel route 3412. For instance, a directions control 3504 is presented that is selectable to cause directions for traversing the travel route 3412 to be presented. A traffic control 3506 is also presented that is selectable to cause traffic information about the travel route 3412 to be presented. The traffic information, for instance, includes current traffic conditions along the travel route 3412 and an estimated travel time for traversing the travel route 3412. A navigation control 3508 is also presented that is selectable to initiate a navigation experience for navigating the travel route 3412. For instance, selection of the navigation control 3508 causes real-time navigation instructions to be presented and based on a detected location of a user. Generally, the different information about the travel route 3412 can be provided in various forms, such as in text form, audible form, visual form such as animation and/or video, and so forth.

[0247] In at least some implementations, the information for the travel route 3412 is retrieved from a transient ink layer for the map GUI 3402. For instance, a transient layer is preconfigured with information about geographical locations presented in the map GUI 3402 and is bound to the map GUI 3402. Accordingly, in response to user interaction with the map GUI to apply ink, relevant geographical information is retrieved from the transient layer and presented to the user, such as via the navigation window 1502.

[0248] FIG. 36 depicts an example implementation scenario 3600 for ink and maps in accordance with one or more implementations. The scenario 3600, for example, represents a continuation and/or variation of one or more of the scenarios described above. The scenario 3600 includes the map GUI 3402 displayed on the display device 112. In the upper portion of the scenario 3600, a user provides touch input with their finger to select the highway 3406, which causes the highway 3406 to be visually highlighted in the map GUI 3402. According to various implementations, the visual highlighting indicates that the highway 3406 is activated within the map GUI 3402 such that the user may apply ink to the highway 3406 and the ink will be bound to the highway 3406.

[0249] Accordingly, and proceeding to the lower portion of the scenario 3600, the user manipulates the pen 128 to apply ink to trace a partial travel route 3602 over the highway 3406. The scenario 3600 then proceeds to the scenario 3700, discussed below.

[0250] FIG. 37 depicts an example implementation scenario 3700 for ink and maps in accordance with one or more implementations. The scenario 3700, for example, represents a continuation of the scenario 3600 described above. The scenario 3700 includes the map GUI 3402 displayed on the display device 112. In the upper portion of the scenario 3700, a user provides touch input with their finger to select the highway 3404, which causes the highway 3404 to be visually highlighted in the map GUI 3402. According to various implementations, the visual highlighting indicates that the highway 3404 is activated within the map GUI 3402 such that the user may apply ink to the highway 3404 and the ink will be bound to the highway 3404.

[0251] Accordingly, and proceeding to the lower portion of the scenario 3700, the user manipulates the pen 128 to apply ink to trace a partial travel route 3702 over the highway 3404. The scenario 3600 then proceeds to the scenario 3800, discussed below.

[0252] FIG. 38 depicts an example implementation scenario 3800 for ink and maps in accordance with one or more implementations. The scenario 3800, for example, represents a continuation of the scenarios 3600, 3700 described above. The scenario 3800 includes the map GUI 3402 displayed on the display device 112. In the upper portion of the scenario 3800, a user provides touch input with their finger to select the airport location 3410, which causes a road leading to the airport location 3410 to be visually highlighted in the map GUI 3402. According to various implementations, the visual highlighting indicates that the airport location 3410 is activated within the map GUI 3402 such that the user may apply ink to the airport location 3410 and/or the road leading to the airport location 3410.

[0253] Accordingly, and proceeding to the lower portion of the scenario 3800, the user manipulates the pen 128 to apply ink to trace a partial travel route 3802 to the airport location 3410.

[0254] According to various implementations, the user may then provide further input to cause the partial travel routes 3602, 3702, 3802 to be combined into a single integrated travel route, e.g., the travel route 3412 discussed above. For instance, the user may apply a particular gesture to the display device 112 using the pen 128 and/or their finger, and the gesture is interpreted by the map module 108 as a command to interpret the travel routes 3602, 3702, 3802 as a single integrated travel route, and to present information about the travel route. Example ways of presenting information about a travel route are discussed above, such as with reference to the navigation window 3502.

[0255] Thus, the scenarios 3500-3800 illustrate implementations where particular travel paths are selectable to activate the travel paths for receiving ink input. Further, ink input to the travel paths can be utilized to retrieve information about the travel paths, such as to provide information about a travel route along the travel paths. As illustrated in the scenarios 3500-3800, an entire span of a particular travel path can be highlighted and a user may apply ink to select a portion (e.g., less than all) of the travel path that the user is interested in for a particular travel route.

[0256] In at least some implementations, ink used to trace a travel route is applied in a transient ink mode. Thus, a travel route may be saved as part of a transient ink layer for a map such that the travel route may be accessed at a later time. Thus, multiple travel routes for a particular map may be saved in respective transient layers that are bound to the map.

[0257] While the scenarios presented above are described with reference to highways, it is to be appreciated that similar scenarios apply for a variety of different travel paths, such as public transit routes, pedestrian routes, marine travel routes, air travel routes, and so forth.

[0258] FIGS. 39 and 40 are a flow diagrams that describe steps in methods in accordance with one or more embodiments. The methods, for instance, describe example procedures for ink and maps in accordance with one or more implementations. In at least some implementations, the methods represent extensions of the methods described above. According to one or more implementations, the methods represent operations of the client device 102, such as performed by the ink module 120 and/or the map module 108. For example, the methods represent example procedures for performing the implementation scenarios for ink and maps described above.

[0259] FIG. 39 is a flow diagram that describes steps in a method for providing information for a travel route based on ink input in accordance with one or more implementations.
Step 3900 receives ink input via a pen to trace a travel route within a map displayed on a display device. The ink input, for instance, traces a travel route between a set of destinations. In at least some implementations, the user may activate a travel path involved in the travel route, such as to highlight the travel path prior to tracing the travel route.

Step 3902 retrieves information about the travel route. The map module 108, for instance, retrieves various types of information about the travel route, such as directions for navigating the travel route, landmarks positioned adjacent to the travel route, a distance of the travel route, and so forth.

Step 3904 presents the information on the display device. The information, for instance, is output as graphics on the display device. Additionally or alternatively, the information may be output in other ways, such as via audible output, via an electronic message, and so forth. An example way of presenting geographic information is presented above with reference to the scenario 3500.

FIG. 40 is a flow diagram that describes steps in a method for highlighting a portion of a travel path to receive ink input in accordance with one or more implementations. In at least the implementations, the method is performed prior to receiving ink input to a travel path, such as prior to performing the method described above with reference to FIG. 39.

Step 4000 receives a user selection of a travel path in a map. The map is displayed in a GUI, such as discussed in the example scenarios described above. A user, for instance, selects a travel path via a suitable input technique, such as touch input, mouse input, stylus input, and so forth.

Step 4002 highlights the travel path for ink input within the map. The travel path, for example, is visually emphasized within the map. In at least some implementations, an entire span of the travel path is highlighted. Highlighting the travel path indicates that the travel path is activated such that a user can apply ink input to trace a travel route along a portion of the travel path. For instance, applying ink to a portion of the highlighted travel path causes data (e.g., metadata) about the portion of the travel path to be extracted and presented.

Accordingly, implementations for ink and maps provide diverse ways of utilizing ink to interact with maps and to extract information from maps. For instance, a user may simply use ink to trace a travel route on a map and be presented with information about the travel route, thus reducing the number of user interactions required to retrieve information about the travel route.

FIG. 41 is a flow diagram that describes steps in a method for propagating an ink travel route to a transient layer in accordance with one or more implementations. In at least the implementations, the method is performed after receiving ink input to a travel path, such as after performing the method described above with reference to FIG. 39.

Step 4100 propagates ink input tracing a travel route on a map to a transient ink layer. For instance, ink content traced over one or more travel paths on a map is saved to a transient ink layer, e.g., a transient ink layer that is bound to the map. In at least some implementations, a visual representation of the ink input is propagated to the transient layer. Further, information about the travel route may also be propagated to the transient ink layer. Examples of different travel route information are detailed above.

Step 4102 enables the transient ink layer to be accessible to retrieve the ink input. For instance, the ink module 120 enables a user to access the transient ink layer. In at least some implementations, accessing the transient ink layer causes the ink input to be overlaid on the map to recreate the tracing of the travel route. Further, travel route information about the travel route may be retrieved from the transient ink layer and presented. Thus, the various attributes of transient ink discussed above may be applied in ink and map implementations.

Having described some example implementation scenarios and procedures for ink for interaction, consider now a discussion of an example system and device in accordance with one or more embodiments.

Example System and Device

FIG. 42 illustrates an example system generally at 4200 that includes an example computing device 4202 that is representative of one or more computing systems and/or devices that may implement various techniques described herein. For example, the client device 102 discussed above with reference to FIG. 1 can be embodied as the computing device 4202. The computing device 4202 may be, for example, a server of a service provider, a device associated with the client (e.g., a client device), an on-chip system, and/or any other suitable computing device or computing system.

The example computing device 4202 as illustrated includes a processing system 4204, one or more computer-readable media 4206, and one or more Input/Output (I/O) Interfaces 4208 that are communicatively coupled, one to another. Although not shown, the computing device 4202 may further include a system bus or other data and command transfer system that couples the various components, one to another. A system bus can include any one or combination of different bus structures, such as a memory bus or memory controller, a peripheral bus, a universal serial bus, and/or a processor or local bus that utilizes any of a variety of bus architectures. A variety of other examples are also contemplated, such as control and data lines.

The processing system 4204 is representative of functionality to perform one or more operations using hardware. Accordingly, the processing system 4204 is illustrated as including hardware element 4210 that may be configured as processors, functional blocks, and so forth. This may include implementation in hardware as an application specific integrated circuit or other logic device formed using one or more semiconductors. The hardware elements 4210 are not limited by the materials from which they are formed or the processing mechanisms employed therein. For example, processors may be comprised of semiconductor(s) and/or transistors (e.g., electronic integrated circuits (ICs)). In such a context, processor-executable instructions may be electronically-executable instructions.

The computer-readable media 4206 is illustrated as including memory/storage 4212. The memory/storage 4212 represents memory/storage capacity associated with one or more computer-readable media. The memory/storage 4212 may include volatile media (such as random access memory (RAM)) and/or nonvolatile media (such as read only memory (ROM), Flash memory, optical disks, magnetic disks, and so forth). The memory/storage 4212 may include fixed media (e.g., RAM, ROM, a fixed hard drive, and so on) as well removable media (e.g., Flash memory, a removable hard drive, an optical disc, and so forth). The computer-readable media 4206 may be configured in a variety of other ways as further described below.
Input/output interface(s) 4208 are representative of functionality to allow a user to enter commands and information to computing device 4202, and also allow information to be presented to the user and/or other components or devices using various input/output devices. Examples of input devices include a keyboard, a cursor control device (e.g., a mouse), a microphone (e.g., for voice recognition and/or spoken input), a scanner, touch functionality (e.g., capacitive or other sensors that are configured to detect physical touch), a camera (e.g., which may employ visible or non-visible wavelengths such as infrared frequencies to detect movement that does not involve touch as gestures), and so forth. Examples of output devices include a display device (e.g., a monitor or projector), speakers, a printer, a network card, tactile-response device, and so forth. Thus, the computing device 4202 may be configured in a variety of ways as further described below to support user interaction.

Various techniques may be described herein in the general context of software, hardware elements, or program modules. Generally, such modules include routines, programs, objects, elements, components, data structures, and so forth that perform particular tasks or implement particular abstract data types. The terms “module,” “functionality,” “entity,” and “component” as used herein generally represent software, firmware, hardware, or a combination thereof. The features of the techniques described herein are platform-independent, meaning that the techniques may be implemented on a variety of commercial computing platforms having a variety of processors.

An implementation of the described modules and techniques may be stored on or transmitted across some form of computer-readable media. The computer-readable media may include a variety of media that may be accessed by the computing device 4202. By way of example, and not limitation, computer-readable media may include “computer-readable storage media” and “computer-readable signal media.”

“Computer-readable storage media” may refer to media and/or devices that enable persistent storage of information in contrast to mere signal transmission, carrier waves, or signals per se. Computer-readable storage media do not include signals per se. The computer-readable storage media includes hardware such as volatile and non-volatile, removable and non-removable media and/or storage devices implemented in a method or technology suitable for storage of information such as computer readable instructions, data structures, program modules, logic elements/circuits, or other data. Examples of computer-readable storage media may include, but are not limited to, RAM, ROM, EEPROM, flash memory or other memory technology, CD-ROM, digital versatile disks (DVD) or other optical storage, hard disks, magnetic cassettes, magnetic tape, magnetic disk storage or other magnetic storage devices, or other storage device, tangible media, or article of manufacture suitable to store the desired information and which may be accessed by a computer.

“Computer-readable signal media” may refer to a signal-bearing medium that is configured to transmit instructions to the hardware of the computing device 4202, such as via a network. Signal media typically may embody computer readable instructions, data structures, program modules, or other data in a modulated data signal, such as carrier waves, data signals, or other transport mechanism. Signal media also include any information delivery media. The term “modulated data signal” means a signal that has one or more of its characteristics set or changed in such a manner as to encode information in the signal. By way of example, and not limitation, communication media include wired media such as a wired network or direct-wire connection, and wireless media such as acoustic, radio frequency (RF), infrared, and other wireless media.

As previously described, hardware elements 4210 and computer-readable media 4206 are representative of instructions, modules, programmable device logic and/or fixed device logic implemented in a hardware form that may be employed in some embodiments to implement at least some aspects of the techniques described herein. Hardware elements may include components of an integrated circuit or on-chip system, an application-specific integrated circuit (ASIC), a field-programmable gate array (FPGA), a complex programmable logic device (CPLD), and other implementations in silicon or other hardware devices. In this context, a hardware element may operate as a processing device that performs program tasks defined by instructions, modules, and/or logic embodied by the hardware element as well as a hardware device utilized to store instructions for execution, e.g., the computer-readable storage media described previously.

Combinations of the foregoing may also be employed to implement various techniques and modules described herein. Accordingly, software, hardware, or program modules and other program modules may be implemented as one or more instructions and/or logic embodied on some form of computer-readable storage media and/or by one or more hardware elements 4210. The computing device 4202 may be configured to implement particular instructions and/or functions corresponding to the software and/or hardware modules. Accordingly, implementation of modules that are executable by the computing device 4202 as software may be achieved at least partially in hardware, e.g., through use of computer-readable storage media and/or hardware elements 4210 of the processing system. The instructions and/or functions may be executable/operable by one or more articles of manufacture (for example, one or more computing devices 4202 and/or processing systems 4204) to implement techniques, modules, and examples described herein.

As further illustrated in FIG. 42, the example system 4200 enables ubiquitous environments for a seamless user experience when running applications on a personal computer (PC), a television device, and/or a mobile device. Services and applications run substantially similar in all three environments for a common user experience when transitioning from one device to the next while utilizing an application, playing a video game, watching a video, and so on.

In the example system 4200, multiple devices are interconnected through a central computing device. The central computing device may be local to the multiple devices or may be located remotely from the multiple devices. In one embodiment, the central computing device may be a cloud of one or more server computers that are connected to the multiple devices through a network, the Internet, or other data communication link.

In one embodiment, this interconnection architecture enables functionality to be delivered across multiple devices to provide a common and seamless experience to a user of the multiple devices. Each of the multiple devices may have different physical requirements and capabilities, and the central computing device uses a platform to enable the delivery of an experience to the device that is both tailored to the device and yet common to all devices. In one embodiment, a
A class of target devices is created and experiences are tailored to the generic class of devices. A class of devices may be defined by physical features, types of usage, or other common characteristics of the devices.

[0286] In various implementations, the computing device 4202 may assume a variety of different configurations, such as for computer 4214, mobile 4216, and television 4218 uses. Each of these configurations includes devices that may have generally different constructs and capabilities, and thus the computing device 4202 may be configured according to one or more of the different device classes. For instance, the computing device 4202 may be implemented as the computer 4214 class of a device that includes a personal computer, desktop computer, a multi-screen computer, laptop computer, netbook, and so on.

[0287] The computing device 4202 may also be implemented as the mobile 4216 class of device that includes mobile devices, such as a mobile phone, portable music player, portable gaming device, a tablet computer, a wearable device, a multi-screen computer, and so on. The computing device 4202 may also be implemented as the television 4218 class of device that includes devices having or connected to generally larger screens in casual viewing environments. These devices include televisions, set-top boxes, gaming consoles, and so on.

[0288] The techniques described herein may be supported by these various configurations of the computing device 4202 and are not limited to the specific examples of the techniques described herein. For example, functionalities discussed with reference to the client device 102 may be implemented all or in part through use of a distributed system, such as over a “cloud” 4220 via a platform 4222 as described below.

[0289] The cloud 4220 includes and/or is representative of a platform 4222 for resources 4224. The platform 4222 abstracts underlying functionality of hardware (e.g., servers) and software resources of the cloud 4220. The resources 4224 may include applications and/or data that can be utilized while computer processing is executed on servers that are remote from the computing device 4202. Resources 4224 can also include services provided over the Internet and/or through a subscriber network, such as a cellular or Wi-Fi network.

[0290] The platform 4222 may abstract resources and functions to connect the computing device 4202 with other computing devices. The platform 4222 may also serve to abstract scaling of resources to provide a corresponding level of scale to encountered demand for the resources 4224 that are implemented via the platform 4222. Accordingly, in an interconnected device embodiment, implementation of functionality described herein may be distributed throughout the system 4200. For example, the functionality may be implemented in part on the computing device 4202 as well as via the platform 4222 that abstracts the functionality of the cloud 4220.

[0291] Discussed herein are a number of methods that may be implemented to perform techniques discussed herein. Aspects of the methods may be implemented in hardware, firmware, or software, or a combination thereof. The methods are shown as a set of steps that specify operations performed by one or more devices and are not necessarily limited to the orders shown for performing the operations by the respective blocks. Further, an operation shown with respect to a particular method may be combined and/or interchanged with an operation of a different method in accordance with one or more implementations. Aspects of the methods can be implemented via interaction between various entities discussed above with reference to the environment 100.

[0292] Implementations discussed herein include:

Example 1

[0293] A system for modifying ink content based on a gesture, the system including: one or more processors; and one or more computer-readable storage media storing computer-executable instructions that, responsive to execution by the one or more processors, cause the system to perform operations including: detecting that freehand ink content is applied to a display via a pen; identifying a touch gesture that is applied to the display; and modifying the ink content based on the touch gesture by mapping the touch gesture to a particular operation to be performed on the ink content.

Example 2

[0294] A system as described in example 1, wherein the touch gesture is identified as touch input via at least one finger and while the pen is in contact with the display.

Example 3

[0295] A system as described in one of more of examples 1 or 2, wherein said modifying includes converting the freehand ink content into a machine-encoded shape.

Example 4

[0296] A system as described in one of more of examples 1-3, wherein the touch gesture includes one or more fingers in contact with the display, and wherein touch gestures with different numbers of fingers in contact with the display are mapped to different respective operations to be performed on the ink content.

Example 5

[0297] A system as described in one of more of examples 1-4, wherein the touch gesture includes a finger motion on the display, and wherein touch gestures with different finger motions are mapped to different respective operations to be performed on the ink content.

Example 6

[0298] A system as described in one of more of examples 1-5, wherein freehand ink content includes a freehand line with at least some curvature, and wherein said modifying includes converting the freehand line to a straight line.

Example 7

[0299] A system as described in one of more of examples 1-6, wherein the touch gesture includes a finger in contact with the freehand line, said modifying includes converting the freehand line to a straight line, and wherein the operations further include causing the straight line to pivot about a point on the display with which the finger is in contact in response to user input to the straight line.

Example 8

[0300] A system as described in one of more of examples 1-7, wherein said modifying includes converting the freehand ink content into a machine-encoded shape, and wherein the operations further include: identifying a further touch gesture
that is applied to the display; and performing an operation on
the machine-encoded shape based on the further touch ges-
ture.

Example 9

[0301] A system for inserting space in a document based on
ink applied to the document, the system including: one or
more processors; and one or more computer-readable storage
media storing computer-executable instructions that, respon-
sive to execution by the one or more processors, cause the
system to perform operations including: detecting that an ink
line is applied to a document displayed on a display device via
a pen; converting by a computing system the ink line into a
document divider that divides the document into a first por-
tion on a first side of the document divider, and a second
portion on a second side of the document divider; and receiv-
ing input to one or more of the first portion or the second
portion to cause a space to be inserted between the first
portion and the second portion.

Example 10

[0302] A system as described in example 9, wherein the
operations further include causing a visual affordance to be
displayed that indicates that one or more of the first portion
or the second portion of the document are manipulable to cause
the space to be inserted.

Example 11

[0303] A system as described in one or more of examples 9
or 10, wherein said converting is performed in response to a
recognition operation that recognizes the ink line as a com-
mand to generate the document divider and that is performed
independent of user input after applying the ink line.

Example 12

[0304] A system as described in one or more of examples
9-11, wherein the input includes a drag operation to drag the
one or more of the first portion or the second portion with
in the display device, and wherein a size of the space is pro-
tional to a size of the drag operation.

Example 13

[0305] A system for presenting information about a travel
route based on ink input tracing the travel route, the system
including: one or more processors; and one or more com-
puter-readable storage media storing computer-executable
instructions that, responsive to execution by the one or more
processors, cause the system to perform operations including:
receiving ink input via a pen to trace a travel route within a
map displayed on a display device; retrieving information
about the travel route; and presenting the information on the
display device.

Example 14

[0306] A system as described in example 13, wherein the
ink input traces the travel route over one or more travel paths
displayed within the map, and wherein the information
includes information about the one or more travel paths.

Example 15

[0307] A system as described in one or more of examples
13 or 14, wherein said retrieving and said presenting are
performed automatically in response to said receiving and
independent of user input after tracing the travel route.

Example 16

[0308] A system as described in one or more of examples
13-15, wherein said retrieving includes: recognizing, inde-
dependent of user input after tracing the travel route, that
the travel route occurs between two locations, wherein the in-
formation identifies the two locations and includes information
about navigating the travel route between the two locations.

Example 17

[0309] A system as described in one or more of examples
13-16, wherein operations further include, prior to said
receiving ink input: receiving a user selection of a travel path
within the map; and highlighting the travel path for ink input
within the map.

Example 18

[0310] A system as described in one or more of examples
13-17, wherein operations further include, prior to said
receiving ink input: receiving a user selection of a travel path
within the map; and highlighting the travel path for ink input
within the map, wherein the travel route is traced over less
than a highlighted portion of the travel path.

Example 19

[0311] A system as described in one or more of examples
13-18, wherein operations further include, prior to said
receiving ink input: receiving user selections of different
travel paths within the map; and highlighting the different
travel paths for ink input within the map, wherein the travel
route is traced over portions of the different travel paths.

Example 20

[0312] A system as described in one or more of examples
13-19, wherein operations further include: propagating the
ink input tracing the travel route to a transient ink layer; and
enabling the transient ink layer to be accessible to retrieve the
ink input.

CONCLUSION

[0313] Techniques for ink for interaction are described.
Although embodiments are described in language specific to
structural features and/or methodological acts, it is to be
understood that the embodiments defined in the appended
claims are not necessarily limited to the specific features or
acts described. Rather, the specific features and acts are
disclosed as example forms of implementing the claimed
embodiments.

What is claimed is:
1. A system comprising:
one or more processors; and
one or more computer-readable storage media storing
computer-executable instructions that, responsive to
execution by the one or more processors, cause the sys-
tem to perform operations including:
detecting that freehand ink content is applied to a display via a pen;
identifying a touch gesture that is applied to the display; and
modifying the ink content based on the touch gesture by
mapping the touch gesture to a particular operation to be performed on the ink content.

2. The system as described in claim 1, wherein the touch
gesture is identified as touch input via at least one finger and
while the pen is in contact with the display.

3. The system as described in claim 1, wherein said modifying comprises converting the freehand ink content into a
machine-encoded shape.

4. The system as described in claim 1, wherein the touch
gesture comprises one or more fingers in contact with the
display, and wherein touch gestures with different numbers of
fingers in contact with the display are mapped to different
respective operations to be performed on the ink content.

5. The system as described in claim 1, wherein the touch
gesture comprises a finger motion on the display, and wherein
touch gestures with different finger motions are mapped to
different respective operations to be performed on the ink content.

6. The system as described in claim 1, wherein freehand ink
content comprises a freehand line with at least some curve,
and wherein said modifying comprises converting the
freehand line to a straight line.

7. The system as described in claim 1, wherein the touch
gesture comprises a finger in contact with the freehand line,
said modifying comprises converting the freehand line to a
straight line, and wherein the operations further include caus-
ing the straight line to pivot about a point on the display with
which the finger is in contact in response to user input to the
straight line.

8. The system as described in claim 1, wherein said modifying comprises converting the freehand ink content into a
machine-encoded shape, and wherein the operations further include:
identifying a further touch gesture that is applied to the
display; and
performing an operation on the machine-encoded shape
based on the further touch gesture.

9. A system comprising:
one or more processors; and
one or more computer-readable storage media storing
computer-executable instructions that, responsive to
execution by the one or more processors, cause the system
to perform operations including:
detecting that an ink line is applied to a document displayed
on a display device via a pen;
converting by a computing system the ink line into a docu-
ment divider that divides the document into a first por-
tion on a first side of the document divider, and a second
portion on a second side of the document divider; and
receiving input to one or more of the first portion or the
second portion to cause a space to be inserted between
the first portion and the second portion.

10. A system as recited in claim 9, wherein the operations
further include causing a visual affordance to be displayed
that indicates that one or more of the first portion or the
second portion of the document are manipulable to cause the
space to be inserted.

11. A system as recited in claim 9, wherein said converting
is performed in response to a recognition operation that rec-
ognizes the ink line as a command to generate the document
divider and that is performed independent of user input after
applying the ink line.

12. A system as recited in claim 9, wherein the input comprises
a drag operation to drag the one or more of the first
portion or the second portion within the display device, and
wherein a size of the space is proportional to a size of the drag
operation.

13. A system comprising:
one or more processors; and
one or more computer-readable storage media storing
computer-executable instructions that, responsive to
execution by the one or more processors, cause the system
to perform operations including:
receiving ink input via a pen to trace a travel route within a
map displayed on a display device;
retrieving information about the travel route; and
presenting the information on the display device.

14. A system as recited in claim 13, wherein the ink input
traces the travel route over one or more travel paths displayed
within the map, and wherein the information includes informa-
tion about the one or more travel paths.

15. A system as recited in claim 13, wherein said retrieving
and said presenting are performed automatically in response
to said receiving and independent of user input after tracing
the travel route.

16. A system as recited in claim 13, wherein said retrieving
comprises:
recognizing, independent of user input after tracing the
travel route, that the travel route occurs between two
locations,
wherein the information identifies the two locations and
includes information about navigating the travel route
between the two locations.

17. A system as recited in claim 13, wherein operations
further include, prior to said receiving ink input:
receiving a user selection of a travel path within the map;
and
highlighting the travel path for ink input within the map.

18. A system as recited in claim 13, wherein operations
further include, prior to said receiving ink input:
receiving a user selection of a travel path within the map;
and
highlighting the travel path for ink input within the map,
wherein the travel route is traced over less than a high-
lighted portion of the travel path.

19. A system as recited in claim 13, wherein operations
further include, prior to said receiving ink input:
receiving user selections of different travel paths within the
map; and
highlighting the different travel paths for ink input within
the map, wherein the travel route is traced over portions of
the different travel paths.

20. A system as recited in claim 13, wherein operations
further include:
propagating the ink input tracing the travel route to a tran-
sient ink layer; and
enabling the transient ink layer to be accessible to retrieve
the ink input.

* * * * *