



(86) Date de dépôt PCT/PCT Filing Date: 2014/09/29
(87) Date publication PCT/PCT Publication Date: 2015/04/02
(85) Entrée phase nationale/National Entry: 2016/03/01
(86) N° demande PCT/PCT Application No.: US 2014/057938
(87) N° publication PCT/PCT Publication No.: 2015/048600
(30) Priorités/Priorities: 2013/09/30 (US61/884,743);
2013/11/15 (US61/905,128); 2013/11/15 (US61/905,114);
2013/11/15 (US61/905,111); 2013/11/15 (US61/905,119);
2013/11/15 (US61/905,101); 2013/11/15 (US61/905,105);
2013/11/15 (US61/905,129); 2013/11/15 (US61/905,116);
2013/11/17 (US61/905,243); 2013/11/17 (US61/905,247);
2014/04/01 (US14/231,846)

(51) Cl.Int./Int.Cl. *G06F 3/0481* (2013.01)
(71) Demandeur/Applicant:
MICROSOFT TECHNOLOGY LICENSING, LLC, US
(72) Inventeurs/Inventors:
DANTON, STEPHEN MICHAEL, US;
JOSHI, VISHAL R., US;
ANAND, KARANDEEP SINGH, US;
STAPLES, WILLIAM J., US;
BHOJAWALA, NAFISA, US;
ALEXANDER, BRENDYN, US;
OLENICK, BRAD, US;
...
(74) Agent: SMART & BIGGAR

(54) Titre : SEQUENCE DE LAMES EXTENSIBLE LE LONG D'UNE DIRECTION DE CANEVAS POUVANT ETRE
RENDUE PANORAMIQUE

(54) Title: EXTENDABLE BLADE SEQUENCE ALONG PANNABLE CANVAS DIRECTION

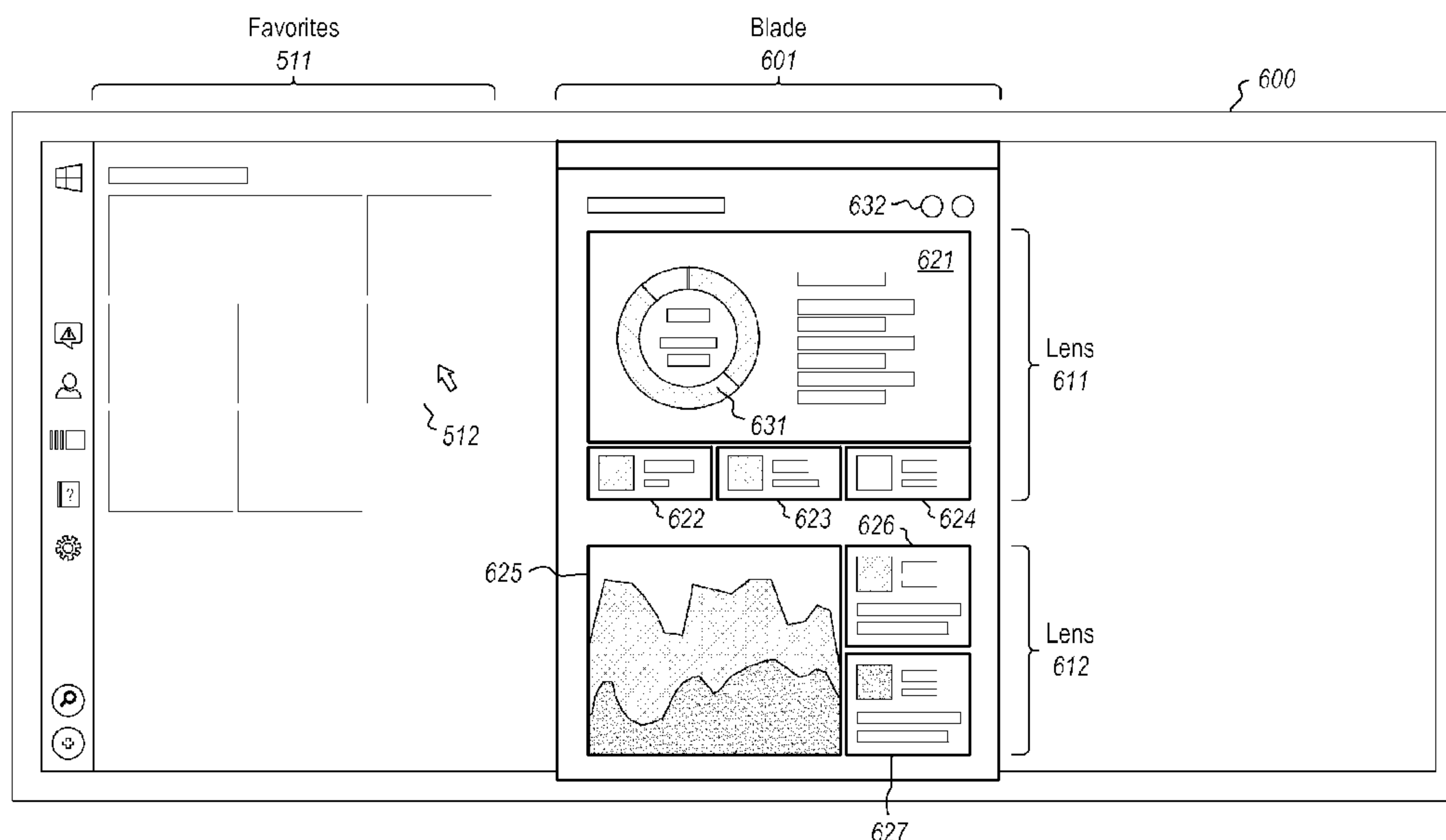


Figure 6

(57) **Abrégé/Abstract:**

A user interface that includes a canvas that extends in one direction. An activation mechanism may be used to generate an initial blade in the canvas. A blade is a user interface element that occupies a portion of the canvas in the dimension along which the canvas extends. The blade includes multiple selectable elements that each have a corresponding blade. The user interface includes an extension mechanism configured to present a corresponding subsequent blade on the canvas perhaps adjacent to the prior blade when a selected element from a prior blade is selected, the subsequent blade also including multiple selectable elements that may be hierarchically structured. In this manner, blade chains may be created representative of a journey that the user has taken since initiating the first blade.



(72) **Inventeurs(suite)/Inventors(continued)**: STERLING, JONAH BUSH, US; WELICKI, LEON EZEQUIEL, US;
JOSHI, MADHUR, US; HARRIS, JON, US; BECKWITH, JUSTIN, US

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

(19) World Intellectual Property
Organization
International Bureau(43) International Publication Date
2 April 2015 (02.04.2015)(10) International Publication Number
WO 2015/048600 A1(51) International Patent Classification:
G06F 3/0481 (2013.01)(21) International Application Number:
PCT/US2014/057938(22) International Filing Date:
29 September 2014 (29.09.2014)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

61/884,743	30 September 2013 (30.09.2013)	US
61/905,128	15 November 2013 (15.11.2013)	US
61/905,111	15 November 2013 (15.11.2013)	US
61/905,114	15 November 2013 (15.11.2013)	US
61/905,116	15 November 2013 (15.11.2013)	US
61/905,129	15 November 2013 (15.11.2013)	US
61/905,105	15 November 2013 (15.11.2013)	US
61/905,101	15 November 2013 (15.11.2013)	US
61/905,119	15 November 2013 (15.11.2013)	US
61/905,243	17 November 2013 (17.11.2013)	US
61/905,247	17 November 2013 (17.11.2013)	US
14/231,846	1 April 2014 (01.04.2014)	US

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

(72) Inventors: **DANTON, Stephen, Michael**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **JOSHI, Vishal, R.**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **ANAND, Karandeep, Singh**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **STAPLES, William, J.**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **BHOJAWALA, Nafisa**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **ALEXANDER, Brendyn**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **OLENICK, Brad**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **STERLING, Jonah, Bush**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **WELICKI, Leon, Ezequiel**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **JOSHI, Madhur**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **HARRIS, Jon**; c/o Microsoft Corporation, LCA - International Patents

[Continued on next page]

(54) Title: EXTENDABLE BLADE SEQUENCE ALONG PANNABLE CANVAS DIRECTION

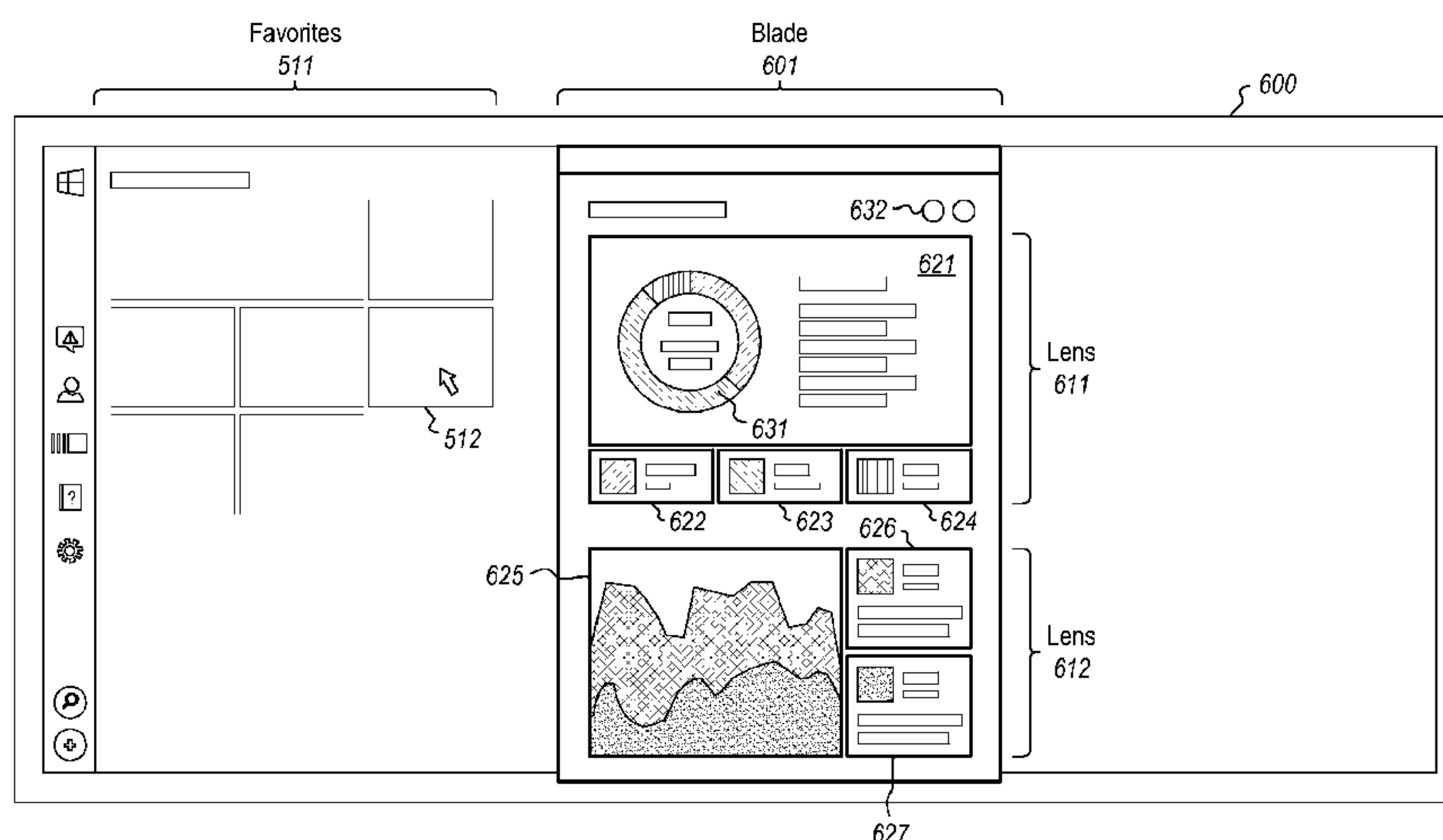


Figure 6

(57) Abstract: A user interface that includes a canvas that extends in one direction. An activation mechanism may be used to generate an initial blade in the canvas. A blade is a user interface element that occupies a portion of the canvas in the dimension along which the canvas extends. The blade includes multiple selectable elements that each have a corresponding blade. The user interface includes an extension mechanism configured to present a corresponding subsequent blade on the canvas perhaps adjacent to the prior blade when a selected element from a prior blade is selected, the subsequent blade also including multiple selectable elements that may be hierarchically structured. In this manner, blade chains may be created representative of a journey that the user has taken since initiating the first blade.



WO 2015/048600 A1

WO 2015/048600 A1

(8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US). **BECKWITH, Justin**; c/o Microsoft Corporation, LCA - International Patents (8/1172), One Microsoft Way, Redmond, Washington 98052-6399 (US).

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

(84) Designated States (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ,

TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

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

Published:

- *with international search report (Art. 21(3))*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

EXTENDABLE BLADE SEQUENCE ALONG PANNABLE CANVAS DIRECTION

BACKGROUND

5 [0001] A current paradigm for navigating through various information contexts is windows based. A classic example of this is the web browser experience. A user might begin with a home page that occupies the entire browser space. The user might then select a hyperlink, whereupon a new window appears. However, the previous window either disappears or, in the case of exercising an option to open the new page in a new window,
10 the previous window is fully, or at least partially, hidden.

BRIEF SUMMARY

[0002] At least some embodiments described herein relate to a user interface that includes a canvas that extends in one direction. An activation mechanism may be used to generate an initial blade in the canvas. A blade is a user interface element that occupies a
15 portion of the canvas in the dimension along which the canvas extends. For instance, the blade might occupy a majority or even all of the canvas in the portion to which it is assigned. The blade includes multiple selectable elements that each have a corresponding blade. If the selectable element is selected, then the corresponding blade is also presented on the canvas. For instance, the new blade might be displayed adjacent the first blade in
20 the extendable direction of the canvas. The elements in the first blade might be hierarchically structured so that a selectable element in the first blade might actually contain one or more other selectable elements.

[0003] The user interface might more generally include an extension mechanism configured to present a corresponding subsequent blade on the canvas when a selected
25 element from a prior blade is selected, the subsequent blade also including multiple selectable elements that may be hierarchically structured. In this manner, long chains of blades may be created representative of a journey that the user has taken since initiating the first blade.

[0004] This Summary is not intended to identify key features or essential features of
30 the claimed subject matter, nor is it intended to be used as an aid in determining the scope of the claimed subject matter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0005] In order to describe the manner in which the above-recited and other advantages and features can be obtained, a more particular description of various

embodiments will be rendered by reference to the appended drawings. Understanding that these drawings depict only sample embodiments and are not therefore to be considered to be limiting of the scope of the invention, the embodiments will be described and explained with additional specificity and detail through the use of the accompanying drawings in which:

[0006] Figure 1 abstractly illustrates a computing system in which some embodiments described herein may be employed;

[0007] Figure 2A abstractly illustrates an environment in which a portion of a pannable canvas is displayed on a displayable area of a display, and a portion of the canvas is not displayed in the displayable area of the display;

[0008] Figure 2B illustrates the user interface of Figure 2A in a state in which the user has selected an activation control to activate an initial blade in a journey;

[0009] Figure 2C illustrates the user interface of Figure 2B in a state in which the user has selected a selectable element of the initial blade to add an additional blade;

[0010] Figure 3 illustrates a supporting architecture for a user interface of Figures 2A through 2C;

[0011] Figure 4 illustrates a flowchart of a method for generating a user interface that has a pannable canvas and that presents a sequence of blades representing a user journey through information;

[0012] Figure 5 illustrates a user interface in which a portion of a canvas is displayed;

[0013] Figure 6 illustrates a user interface resulting from the interaction of the user with an activation control;

[0014] Figure 7 illustrates a user interface that shows how information that is global may still be accessed without having to leave the context of the journey;

[0015] Figure 8 illustrates a user interface that is similar to that of Figure 6, except that command window is activated;

[0016] Figure 9 illustrates a user interface having a single blade in the journey, but in which that single blade has a command space activated;

[0017] Figure 10 illustrates a user interface having a single blade in the journey, in a state in which the user is able to select a selectable element from that single blade;

[0018] Figure 11 illustrates a user interface that results from the user selecting the selectable element, and in which a second blade in the journey is correspondingly appended to for a sequence of blades in the journey;

[0019] Figure 12 illustrates the second blade of Figure 11 only in full representation mode;

[0020] Figure 13 illustrate a journey view in which a journey may be viewed in its entirety, saved, pinned, shared, or closed;

5 [0021] Figure 14 illustrates a flow associated with presenting a selectable item or part which then issues command to underlying information resources;

[0022] Figure 15 illustrates a part presented in different sizes and shapes, and exposing different information in different forms depending on the size and shape;

10 [0023] Figure 16 illustrates a notification as it might appear if not associated with any of the activation controls in the favorites area;

[0024] Figure 17 illustrates the activation of a temporary area and an initial blade in response to selection of the notification;

[0025] Figure 18 illustrates an initial state in which the user selects to create a website;

15 [0026] Figure 19 illustrates a creation pane in which the user enters information about the website and is given options for adding onto the original resource requested;

[0027] Figure 20 illustrates a creation pane in which the user selects a database to be associated with the website, and the system automatically selects an additional add on of a connection string between the website and the database;

20 [0028] Figure 21 illustrates the creation pane in which the user enters information about the database; and

[0029] Figure 22 illustrates the creation pane in which the user enters information about the connection string between the website and the database.

DETAILED DESCRIPTION

25 [0030] At least some embodiments described herein relate to a user interface that includes a canvas that extends in one direction along a pannable direction. An activation mechanism may be used to generate an initial blade in the canvas. A blade is a user interface element that occupies a portion of the canvas in the pannable dimension along which the canvas extends. For instance, the blade might occupy a majority or even all of the canvas in the portion to which it is assigned. The blade includes multiple selectable
30 elements that each have a corresponding blade. If the selectable element is selected, then the corresponding blade is also presented on the canvas. For instance, the new blade might be displayed adjacent the first blade in the extendable direction of the canvas. The elements in the first blade might be hierarchically structured so that a selectable element in the first blade might actually contain one or more other selectable elements.

[0031] The user interface might more generally include an extension mechanism configured to present a corresponding subsequent blade on the canvas when a selected element from a prior blade is selected, the subsequent blade also including multiple selectable elements that may be hierarchically structured. In this manner, long chains of blades may be created representative of a journey that the user has taken since initiating the first blade.

[0032] Some introductory discussion of a computing system will be described with respect to Figure 1. Then, example user interfaces, methods and supporting architectures will be described with respect to subsequent figures.

[0033] Computing systems are now increasingly taking a wide variety of forms. Computing systems may, for example, be handheld devices, appliances, laptop computers, desktop computers, mainframes, distributed computing systems, or even devices that have not conventionally been considered a computing system. In this description and in the claims, the term “computing system” is defined broadly as including any device or system (or combination thereof) that includes at least one physical and tangible processor, and a physical and tangible memory capable of having thereon computer-executable instructions that may be executed by the processor. The memory may take any form and may depend on the nature and form of the computing system. A computing system may be distributed over a network environment and may include multiple constituent computing systems.

[0034] As illustrated in Figure 1, in its most basic configuration, a computing system 100 typically includes at least one processing unit 102 and memory 104. The memory 104 may be physical system memory, which may be volatile, non-volatile, or some combination of the two. The term “memory” may also be used herein to refer to non-volatile mass storage such as physical storage media. If the computing system is distributed, the processing, memory and/or storage capability may be distributed as well. As used herein, the term “executable module” or “executable component” can refer to software objects, routines, or methods that may be executed on the computing system. The different components, modules, engines, and services described herein may be implemented as objects or processes that execute on the computing system (e.g., as separate threads).

[0035] In the description that follows, embodiments are described with reference to acts that are performed by one or more computing systems. If such acts are implemented in software, one or more processors of the associated computing system that performs the act direct the operation of the computing system in response to having executed computer-

executable instructions. For example, such computer-executable instructions may be embodied on one or more computer-readable media that form a computer program product. An example of such an operation involves the manipulation of data. The computer-executable instructions (and the manipulated data) may be stored in the memory
5 104 of the computing system 100. Computing system 100 may also contain communication channels 108 that allow the computing system 100 to communicate with other message processors over, for example, network 110.

[0036] The computing system 100 also includes a display 112 on which a user interface, such as the user interfaces described herein, may be rendered. Such user
10 interfaces may be generated in computer hardware or other computer-represented form prior to rendering. The presentation and/or rendering of such user interfaces may be performed by the computing system 100 by having the processing unit(s) 102 execute one or more computer-executable instructions that are embodied on one or more computer-readable media. Such computer-readable media may form all or a part of a computer
15 program product.

[0037] Embodiments described herein may comprise or utilize a special purpose or general-purpose computer including computer hardware, such as, for example, one or more processors and system memory, as discussed in greater detail below. Embodiments described herein also include physical and other computer-readable media for carrying or
20 storing computer-executable instructions and/or data structures. Such computer-readable media can be any available media that can be accessed by a general purpose or special purpose computer system. Computer-readable media that store computer-executable instructions are physical storage media. Computer-readable media that carry computer-executable instructions are transmission media. Thus, by way of example, and not
25 limitation, embodiments of the invention can comprise at least two distinctly different kinds of computer-readable media: computer storage media and transmission media.

[0038] Computer storage media includes RAM, ROM, EEPROM, CD-ROM or other optical disk storage, magnetic disk storage or other magnetic storage devices, or any other tangible medium which can be used to store desired program code means in the form of
30 computer-executable instructions or data structures and which can be accessed by a general purpose or special purpose computer.

[0039] A “network” is defined as one or more data links that enable the transport of electronic data between computer systems and/or modules and/or other electronic devices. When information is transferred or provided over a network or another communications

connection (either hardwired, wireless, or a combination of hardwired or wireless) to a computer, the computer properly views the connection as a transmission medium. Transmissions media can include a network and/or data links which can be used to carry or desired program code means in the form of computer-executable instructions or data structures and which can be accessed by a general purpose or special purpose computer. Combinations of the above should also be included within the scope of computer-readable media.

[0040] Further, upon reaching various computer system components, program code means in the form of computer-executable instructions or data structures can be transferred automatically from transmission media to computer storage media (or vice versa). For example, computer-executable instructions or data structures received over a network or data link can be buffered in RAM within a network interface module (e.g., a “NIC”), and then eventually transferred to computer system RAM and/or to less volatile computer storage media at a computer system. Thus, it should be understood that computer storage media can be included in computer system components that also (or even primarily) utilize transmission media.

[0041] Computer-executable instructions comprise, for example, instructions and data which, when executed at a processor, cause a general purpose computer, special purpose computer, or special purpose processing device to perform a certain function or group of functions. The computer executable instructions may be, for example, binaries, intermediate format instructions such as assembly language, or even source code. Although the subject matter has been described in language specific to structural features and/or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the described features or acts described above. Rather, the described features and acts are disclosed as example forms of implementing the claims.

[0042] Those skilled in the art will appreciate that the invention may be practiced in network computing environments with many types of computer system configurations, including, personal computers, desktop computers, laptop computers, message processors, hand-held devices, multi-processor systems, microprocessor-based or programmable consumer electronics, network PCs, minicomputers, mainframe computers, mobile telephones, PDAs, pagers, routers, switches, and the like. The invention may also be practiced in distributed system environments where local and remote computer systems, which are linked (either by hardwired data links, wireless data links, or by a combination

of hardwired and wireless data links) through a network, both perform tasks. In a distributed system environment, program modules may be located in both local and remote memory storage devices.

[0043] Figure 2A abstractly illustrates an environment 200A in which a portion 201A
5 of a canvas 202 is displayed on a displayable area 210 of a display, and a portion 201B of the canvas is not displayed in the displayable area 210 of the display. As an example, the display might be, for instance, the display 112 of the computing system 100 of Figure 1. The canvas 202 extends in an extendable direction 221 along a single dimension 222. For instance, in Figure 2A through 2C, the extendable direction 221 is rightward, and the
10 single dimension 222 is horizontally.

[0044] The user interface might include a panning control 231 that the user may manipulate in order to pan the canvas 202 along the single dimension 222 so as to gain a view of a desired portion of at least a populated portion of the canvas 202. Figure 3 illustrates a supporting architecture 300 for a user interface 301. For instance, the user
15 interface 301 is an example of the user interface of Figures 2A through 2C. The supporting architecture 300 is illustrated as including a panning mechanism 311 which operates to pan the canvas 202 in response to user direction.

[0045] In some embodiments, the single dimension 222 may be configurable to be vertical or horizontal according to the tastes of the user. Also, the extendable direction 221
20 may be configurable. For instance, if the dimension 222 is vertical, the user might configure the canvas to extend downward, or upward, according to the user's preference. If the dimension 222 is horizontal, the user might configure the canvas to extend leftward, or rightward, according to the user's preference. The panning control 231 might not activate until the populated portion of the canvas extends to a degree that the populated
25 portion of the canvas cannot be displayed all at once. The supporting architecture 300 illustrates a configuration mechanism 312 that is configured to respond to such user preference configuration.

[0046] As used herein, a "populated" portion of the canvas means that portion of the canvas that contains the sequence of one or more blades built up through user interaction
30 with the canvas. In accordance with the principles described herein, as the user interfaces with the canvas, one or more and potentially many blades may be added to the canvas. Some or all of these blades may be added contiguously. As such, the "extendable direction" of the canvas refers to the direction in which new blades are added to the prior

sequence of one or more contiguous blades upon interacting with selectable elements from the prior blade.

[0047] In this description and in the claims, a “blade” is a user interface element that occupies a position within the canvas. In this sense, a “position” is a range within the single dimension of the canvas. For instance, if the pannable dimension 222 is the horizontal dimension, as in the more specific examples to follow, the blade would occupy the canvas 202 from a left position to a right position. The blade “occupies the position” within the canvas in the sense that between the two boundaries of the blade, the blade occupies at least a majority of the canvas within those two boundaries for each of a majority of points between the two boundaries including at least the two boundaries themselves. In some embodiments, and in the more concrete examples to follow, the blade occupies all of the canvas between the two boundaries, which are left and right boundaries since the dimension is the horizontal dimension in the concrete examples to follow.

[0048] The user interface 210 also includes an activation control 232 that may be activated by a user to present an initial blade on the canvas 220. Each blade may include one or multiple selectable elements. Each selectable element has a corresponding blade such that if the user selects the corresponding selectable element, the corresponding blade appears on the canvas as a next blade on the canvas. For instance, in Figure 2B, the environment 200B is similar to that of environment 200A, except that the selection of activation control 232 causes an initial blade 241 to appear in the canvas 202. The initial blade includes multiple selectable controls 251A, 251B and 251C. Figure 3 illustrates that the supporting architecture includes an activation mechanism 313 that is responsive to activate the initial blade 241 in response to the user interfacing with the activation control 232. Furthermore, the blade correlator 314 maintains a correlation between a selectable element and a blade. Accordingly, the architecture 300 has a knowledge of what blades correspond to each of the selectable controls 251A, 251B, and 251C.

[0049] The user interface 210 also includes an extension mechanism 315, which causes one or more additional blades to be spawned off of the initial blade 241 in response to selection of one of the selectable elements 251A, 251B and 251C. For instance, suppose that the user selects selectable element 251A, the extension mechanism 315 might then reference the corresponding blade identified by correlator 314, and cause the corresponding blade to appear on the canvas 202. For instance, in Figure 2C, the environment 200C shows that the new blade 242 is caused to appear next to the first blade 241. The new blade 242 likewise has selectable controls 252A and 252B. Accordingly,

this process may be repeated by the user repeatedly interfacing with selectable controls from a prior blade, thereby causing a new blade to be added to the sequence of blades. In this sense, the sequence of blades may be thought of as a journey that the user has taken beginning with the first blade.

5 [0050] In some embodiments, each blade in the sequence is associated with an information context. The user may thus have at their view a sequence of blades representing a sequence of information contexts that the user has navigated through. Thus, the canvas represents an area that includes visual representations of various information contexts the user has reviewed as part of an information journey.

10 [0051] In some embodiments, the selectable elements within a particular blade are hierarchically structured. For instance, one selectable element might have one or more child selectable elements. In addition, those child selectable elements might have further child selectable elements, and so forth. Each selectable element has a corresponding blade representation. Accordingly, when generating choosing a selectable element to select in a
15 given prior blade, the user has fine grained control over what kind of blade is to show next. Examples of this hierarchical structuring of selectable elements will be described in subsequent concrete examples.

[0052] Figure 4 illustrates a flowchart of a method 400 for generating a user interface. As the method 400 may be performed by the architecture 300 of Figure 3 in the
20 environments of Figures 2A through 2C, the method 400 will now be described with frequent reference to Figures 2A through 2C, Figure 3, as well as of course Figure 4.

[0053] The method 400 includes providing a canvas that extends in a pannable direction in the user interface (act 401). For instance, in Figure 2A through 2C, the canvas 202 is provided with a pannable dimension 222 and which extends in an extendable
25 direction 221. If an initial blade is not activated through user interaction with the activation control 232 (“No” in decision block 402), the canvas 202 might remain in the state of Figure 2A with no blades populating the canvas.

[0054] However, upon detecting activation of the activation control 232 (“Yes” in decision block 402), the initial blade corresponding to the activation is presented on the
30 canvas (act 403). For instance, in Figure 2B, the initial blade 241 is presented in response to activation of activation control 232. If none of the selectable controls of the initial blade are selected (“No in decision block 404), then the sequence of blades might extend no further. On the other hand, if the user interacts with one of the selectable controls of the prior blade (“Yes” in decision block 404), then the corresponding blade is found (act 405),

and presented (act 406). For instance, in Figure 2C, the additional blade 242 is presented. As each newly added blade also has selectable elements, this process may repeat to build up a large sequence of blades, each representing a different information context. A more concrete example of the user interface will now be described with respect to Figure 5 and numerous subsequent figures.

[0055] Figure 5 illustrates a user interface 500 in which a portion of a canvas is 501 is displayed. The user interface 500 is an example of what a user interface might look like after completing act 401 of Figure 4. In this example, the panning dimension is the horizontal dimension. In other words, the canvas may be panned left and right within at least the populated portion of the canvas. Also, in this example, the extendable direction is towards the right. In other words, at least a majority (and in this example all) of the new blades that are added and that are to be appended to the sequence of one or more existing blades in the journey are to be added appended at the right end of the sequence.

[0056] The canvas includes a collection 511 of activation controls. The collection might be referred to hereinafter as “favorites”. In the illustrated case, there are eight activation controls. Each of the activation controls in the collection 511 is associated with an information context. An information context might be, for example, a web site, a project, a patent tool, or any other context in which information regarding a subject might be provided.

[0057] The user initiates a journey by selecting one of the activation controls. For instance, the user might select the activation control 512 using cursor 513, or by contacting the activation control 512 in the case of a touch screen. This would result in decision block 402 of Figure 4 branching on the “Yes” branch.

[0058] Figure 6 illustrates a user interface 600 resulting from the interaction of the user with the activation control 512. Note that a blade 601 has been presented (as an example of act 403). The blade 601 represents the beginning of a sequence of blades (or a journey) that the user has just embarked upon. The blade 601 is associated with the information context that was associated with the activation control 512 used to spawn the blade 601. In other words, the blade 601 provides a view into data in the information context. The activation control 512 might be visually highlighted so as to give the user the contextual reference that the blade 601 was generated by activation of the activation control 512, versus any of the other activation controls 511.

[0059] In one embodiment, the blade 601 (or any other blades in the journey for that matter) are scrollable in the direction perpendicular to the panning dimension. For

instance, in Figure 6, if the blade 601 was taller than the canvas, then the blade 601 might be scrolled upward and downward to reveal the vertical extent of the blade 601.

[0060] The blade 601 includes a hierarchy of selectable elements, each selectable element having a corresponding blade representation (not shown in Figure 6) which is displayed if the selectable element is selected. At the top of the hierarchy is what is called a “lens” in this example. If, for example, the information context were a web site, the lens might present at a high level the information that might be normally presented when selecting a tab of the web site. For instance, the blade 601 is illustrated as including two lenses 611 and 612.

[0061] At the next level in the hierarchy, each lens 611 includes one or more parts. For instance, lens 611 includes parts 621 through 624, and lens 612 includes parts 625 through 627. Each part represents a view on an aspect of the data, and may be defined by a collection of underlying data and a view definition that defines how that data is to be exposed on the user interface.

[0062] At the lowest level of the hierarchy of selectable elements is an item. Each part may have one or more selectable items. To avoid unnecessarily over-labeling Figure 6, only one selectable item is labeled. For instance, selectable part 621 is illustrated as having selectable item 631.

[0063] A new blade might be launched within the journey by the user selecting any of the lenses 611 and 612, any of the selectable parts 621 through 627, or any of the selectable items within any of the parts. Thus, the user has a large number of options for progressing in the journey through information and the canvas.

[0064] For instance, suppose that the information context of the blade 601 was a patent tool associated with a particular patent application. In that case, the selectable lenses, parts, and items of the blade might allow the user to see summary information in the blade 601 regarding the initial disclosure documents, legal notes, status of the application, actual patent application text, which countries or regions the patent application is being pursued in, and so forth. If the user wanted to see any further details about any of those aspects, the user might the select the element, spawning a new blade that shows further details and selectable items.

[0065] Figure 7 illustrates a user interface 700 that shows how information that is global may still be accessed without having to leave the context of the journey. In particular if the user selects control 701, a navigation pane 702 appears. Thus, the user might, for instance, see what other team members are doing (a global task) while

remaining within a word processing experience (a journey context). If desired, the user might select an item within the navigation pane 702 to start a new journey, thus enabling rapid switching between journeys.

[0066] Commanding may be performed in a constituent way across all artifacts in the user interface. For instance, if a command window is activated, context is used to determine what the prospective target for commands is. For instance, if the cursor is above a particular lens, the command window may then appear and show commands tailored towards operation on the lens given the type of the resource or item represented by the lens. In accordance with embodiments described herein, the command window shell might appear the same regardless of what the prospective target is for the commands in the command window, though the commands themselves may differ as appropriate given the prospective target. For instance, the command window might be the same regardless of whether the prospective target is the background canvas, a particular blade lens, a particular lens

[0067] Figure 8 illustrates a user interface 800 that is similar to that of Figure 6, except that command window 801 is activated. The command window 801 has a primary commands section 811, a secondary commands section 812, and an ecosystem peek section 813. For instance, the primary commands section 811 might include the most frequently used commands given a type of the prospective target. The secondary command section 812 might include other non-primary commands.

[0068] The ecosystem peek section might include information items that are not commands, but rather other information (such as notifications or messages) related to the prospective target. For instance, the ecosystem peek section might include an identification of when the prospective target was last updated, if anyone else is editing the prospective target. Furthermore, the ecosystem peek information may include help information that provides a quick insight into how to use the prospective target and where to get more information about the prospective target.

[0069] The end user can interface with a command in the command space or a command window, and see the command implementation as a declarative model. The user might then extend the declarative module using the same means used to implement the declarative model. Thus, users may author their own commands or extend other commands.

[0070] Figure 9 illustrates a user interface 900 that might appear if the user selects the control 632 of the blade 601 in Figure 6. The control may be present in each blade and

may be used to activate a command space within and corresponding to that blade, with commands tailored towards the type of the resource or item represented by the blade. For instance, in Figure 9, upon selecting control 632 of the blade 601, the command space 901 appears, allowing the user to execute commands on the resource or item represented by the blade 601.

[0071] Figure 10 illustrates a user interface 1000 that is the same as the user interface 700 of Figure 7, except to show that the user is about to select the part 622 of the blade 601. The selection of part 622 is an instance of the method 400 travelling along the “Yes” branch in decision block 404. In response, the corresponding blade of the part 622 is found (act 405), and presented (act 406). For instance, the user interface 1100 of Figure 11 might appear. Here, the entire pannable canvas has shifted to the left aligned with the left side of the previous blade 601, and the new blade 1101 is caused to appear. The new blade 1101 also has a number of selectable elements, thereby allowing the user an opportunity to further extend the sequence of blades if so desired.

[0072] When a new blade is launched, that new blade may be positioned by default on the canvas adjacent the last blade in a sequence of one or more blades that constitute the journey up to that point. However, default positions may vary by blade. For instance, the blade might be pinned to the right or left portion of the display (if the canvas pans horizontally), where panning of the canvas does not affect the position of the pinned blade. The blade might also be caused to float, in which case the blade is stationary in some portion of the display despite panning of the underlying canvas. Having blades be pinned or floating while the canvas is pannable allows the blades to be compared to other blades throughout the journey.

[0073] The default position of the blade might be changed due to some user interaction. For instance, if the user interacts with a certain blade that was originally appended to the sequence of blade prior to that blade in the journey, that certain blade may later be pinned to the left or right of the display, or caused to float, depending on the user interaction. Later, that certain blade might return to its default positioning in the sequence of blades.

[0074] Furthermore, when a part is selected from a prior blade, perhaps more than one blade is launched and appended to the sequence of blades to that point. For instance, if the user selects a monitoring part, they might want to see an overview of the monitoring events, all events that happened this week and all events that happened this month. These three sets might instead be displayed in separate blades that are siblings to each other.

Such sibling blades might reordered in the sequence of blades. The selection of some parts might result in multiple blades automatically, while some parts may permit multiple blades one at a time, upon each instance of the user selecting the part.

[0075] The use of a sequence of blades representing a user's journey through information allows for visualization of cause and effect of certain actions. For instance, when a user edits in the context of a certain part, that certain part may be visually highlighted to reflect to editing. For instance, suppose that a part within blade 1101 were edited, that part might be highlighted. However, the entire chain leading to that part in cause and effect is also highlighted. For instance, part 622, the selection of which caused blade 1101 to be launched in the first place, may likewise be visually highlighted to show editing. Furthermore, the activation control 512 that was selected in order to launch blade 601 might also be highlighted to reflect that there is editing that occurred. If the user saves the changes then the highlighting showing that editing cause and effect chain may be removed. On the other hand, if the user wants to exit that journey without saving, the activation control 512 remains highlighted, reminding the user that there is something along that chain that needs to be edited. There might be a command that allows the user to select that activation control 512 in a certain way so as to quickly return to the edited part within blade 1101, so that the editing may be completed and/or saved.

[0076] This visualized cause and effect between parts and blades may be utilized for other actions as well. For instance, consider confirmations. The conventional confirmation model (e.g. "Do you want to delete this file?") is modal, which means the model forces the user into a decision when they may or may not have all of the information needed to answer the question. As an example, the user may not know whether someone else is using the file. The user does have the option of cancelling out of the operation unless they know how to answer it. However, this lessens the likelihood that the user will remember to return to the action that caused the confirmation in the first place. The model also forces the user to operate in a one and a time fashion. For instance, the user is unable to work through a set of websites, selecting specific ones for a Stop Running operation, and then execute all of the operations at once from a single location. Lastly, working with contexts of different types (e.g. websites and databases) is even more difficult for users - despite this being a very common scenario.

[0077] In accordance with embodiments described herein, confirmations are not modal. Instead, the when a confirmation occurs in that context of a particular part, the causal chain from the activation control all the way through all of the parts and blades, to

the part that generated the confirmation will be highlighted to reflect that there is a confirmation. The user may now exit out of that journey, and evaluate whatever information is desired to evaluate to determine how to respect to the confirmation. The user might then interface with the source of the confirmation (e.g., the activation control) to quickly return to the part that generated the confirmation. The user may then answer the confirmation with more knowledge and security.

[0078] These highlighted paths may also be used to expose errors. For instance, if an error occurs in the context of a part, the user may step back and evaluate the error using other resources. Once the user is satisfied that the error is being handled, the user might close out the error message. Alternatively, the error message may remain until someone handles the problem that caused the error.

[0079] A highlighted path might also be provided for potential availability message that inform the user of available services or resources that may be used to enhance an experience or performance associated with a part.

[0080] Up until this point in the description, each new blade has defaulted to be less than the entire displayable area. For instance, the blade 601 is illustrated as occupying the entire vertical extent of the canvas, but just one third the horizontal extent of the visible portion of the canvas. Such might be considered to a “browse-mode” of the blade, as such allows panning action to more easily see a larger number of blade to allow the user to quickly browse the information contexts through which the user has navigated as part of the journey. Blade 1101 has likewise been opened in browse mode.

[0081] There might be some interaction that might cause the corresponding blade to have some other display mode. For instance, upon interacting with the part 621 of blade 601 as shown in Figure 11, the user interface 1200 of Figure 12 might appear. Here, the user interface 1200 includes a full representation 601' of the blade, the full representation occupying much of the displayed canvas. The full mode may contain a different number or combination of lens and parts than the browse mode. Furthermore, any parts that are in common may in fact be differently sized, and thus present different granularities of details on information, as described further below.

[0082] In some embodiments, if the full representation of the blade is arrived at by interacting with a browse-mode representation of the blade, the full representation of the blade may simply be added to the sequence of blades with the prior blade being the browse-mode representation. The canvas does, however, automatically pan in order to display the entire full representation of the blade. However, the user retains the ability to

pan the canvas despite the display of the full representation of the blade. Further interaction with the full representation may allow for a return to the browse-mode view, perhaps removing the full representation of the blade from the sequence of blades on the canvas.

5 [0083] In some embodiments, for some selectable elements, when selected, the part might default to a full representation, rather than the browse representation. The default setting for a selectable element might be a function of the underlying type of the resource or item represented by the selectable element. The user might also configure certain types to display in different ways by default, or have a chance to change the default setting. The
10 blade might have size control mechanisms that allow adjustment of the size of the blade to other sizes as well. In any case, if the blade being resized is not the last blade in the sequence of blades representing the journey, the position of the later blades may be adjusted so as to remain adjacent to and unobscured by the blade that has been readjusted. Thus, blades can be maximized, but that does not mean they grow overtop of existing
15 blades. Instead, the blade grows width-wise (or more generally along the pannable direction of the canvas), pushing adjacent blades to the left and right. A blade might also be collapsed. For instance, the blade might be made very thin, with little or no information being displayed, but with a control that allows the blade to be expanded again.

[0084] Up until this point, the user interface associated with navigation of a canvas has
20 been described. There might also be a journey view that allows the user to see the entire journey at a glance. This zoomed out view is a semantic representation of the journey, which means that information of importance is maintained and made easier to view, while finer grained details are dropped away. In addition to being able to see more of a navigation context, this view allows users to quickly jump from one blade to another, and
25 do so with an understanding of what they are going to see, rather than having to remember what the target blade contains.

[0085] In addition, the journey view may allow the user to see all of their open journeys, by perhaps scrolling in the perpendicular direction as the canvas pans. This ability to see all open journeys not only makes switching between journeys easy, but also
30 allows the ability to compare journeys. The journey view also enables a apply concept called light path (described later) across journeys.

[0086] The user might also have access to journey controls that allow the user to perform a journey commands (such as save the journey, share the journey, pin the journey as a tile, or close the journey) while viewing the semantics of the journey at a glance. For

instance, at any point in the journey navigation, the user might select a title bar (or any other journey view activation control), to cause the user interface to exit the navigation interface, and enter the journey view interface. The journey view might also have a pop-out comment that allows the journey to be displayed in a window, and thus allow the user to use the remaining portions of the display for other purposes. If there is some change to the underlying data that causes a change in the journey, that change may be reflected in the pop-out window of the journey.

[0087] Figure 13 illustrates a user interface 1300 that illustrates a journey view corresponding to the journey built up through Figure 12. The journey view 1300 includes a journey bar 1301 that spans the extent of all of the blade sequences to that point. The journey bar 1301 includes a command section 1302 that allows the user to pin the journey to a certain area of the screen, share the journey with others (via e-mail, social media, collaborative work environments, and the like), save the journey, or close the journey. For instance, saving a journey might be quite helpful if the journey is to be re-performed frequently, such as the journey being a navigation to a particular database. Sharing a journey might be helpful, for instance, in the case where a tester finds a specific bug and then send the journey they went on to find that bug to a developer. The developer could then quickly understand not just the context of the bug by the set of insights that lead to the bug.

[0088] Figure 14 illustrates a general system flow associated with any selectable item or any selectable part. The system includes information resources 1401. The transforms 1402 associated with the selectable item or part filter through a portion of the total resources 1401 that are relevant for a particular information context. For instance, the transforms 1402 might provide data associated with a website. The selectable part or item 1403 provides the data associated with the particular item or part to a presenter 1404. The presenter is configured to offer up to the display an appropriate view of the data. For instance, the presenter 1404 might present a chart view of the data. Since the selectable part or item is selectable, there are certain commands that may be issued by the part or item depending on its type. These commands may then be executed as represented by arrow 1405 on the underlying information resources 1401. This would affect potentially not only the experience of the item or part 1404, but might also affect any other item or part that uses that altered data.

[0089] This architecture allows developers significant opportunities. For instance, developers can use existing parts and items developed by others. In addition, the

developers can create their own selectable items by defining a new data source 1403 and/or defining a new presenter 1404. The developer might define a type for that item, and then define commands 1405 for that item, which would be exposed to the user if the user were to select the command window in the context of that new type. The developers can also create their own selectable parts by using existing items or creating new items to define a new data source 1403 and/or presenter 1404 for that selectable part. Also, the developer might define a type for that part, and then define commands 1405 for that type, which would be exposed to the user if the user were to select the command window in the context of that new type.

10 [0090] The developer might access pre-defined lenses, each comprises combinations of parts, depending on the goal to be accomplished. For instance, there might be lenses for presenting overviews, usage/billing, monitoring, configuration, linked resources, team views, authoring, and quick starting amongst others. Developers may further construct their own lens by gathering pre-existing and/or newly developed parts and positioning
15 them with respect to each other.

[0091] Figure 15 illustrates a single part in six different sizes and shapes. The larger version of the parts intelligently offers up more information to the user. The size of the part is actually taken into consideration by the presenter 1404 (see Figure 14) when considering what information to present, and how that information should be presented.
20 The intelligence may be applied by the developer of the part when defining a part. In this example, there is a mini-size part version 1501, a small part version 1502, a normal part version 1503, a large part version 1504, a wide part version 1505, and a tall part version 1506. Parts may appear both within blades of a journey, as well as they might actually be an activation control within the set of activation controls 511. In certain contexts, the user
25 might have the option to change the size of the part in accordance with the sizes available for that part. For instance, within the activation controls 511, the user might change a part size to be smaller or larger. The user might also move parts within the activation controls 511 for more preferred layout for the user. By having a discrete number of pre-defined part sizes and shapes, the designers of the part may more intelligently design what
30 information is to be displayed in the part, and how that information is to be displayed, depending on what they judge as most appropriate given the size and shape of the part.

[0092] Various items may be pinned to the activation controls 511 area as a part that may be activated. For instance, an item from the navigation pane 702 may be pinned as a part in the activation controls 511. Commands from a command window or space may be

pinned as a part in the activation controls 511. Likewise, component parts of a lens within a blade may be pinned in the activation controls 511. A pin-all function might also be provided that cases all parts of the blade to be pinned within the activation controls 511 (perhaps as a nested collection within another activation control). Resources, collections, feeds, curated content (such as help topics), and search or query results may likewise be pinned as parts in the activation controls 511. For instance, queries and searches may be performed over a wide variety of information contexts accessible to the computer system. The user can construct queries that go across multiple contexts and then turn the result of that query into a part. That part represents the result of the query and can then be sized and organized as per the part concepts defined above. That part might be provided into each blade that has a context in which the query results are desired.

[0093] Figure 16 illustrates a user interface 1600 in which there are no journeys started yet, but in which the activation controls are illustrated. Note that the activation controls have a variety of different sizes including six mini-sizes parts, four small size parts, four normal size parts, and one large size part. The notification 1601 appears. If the notification was tied to one of the activation controls, then the notification might have appeared within one of the activation controls. However, the notification 1601 was not tied to an activation control, and thus appeared separately.

[0094] Upon selecting the notification 1601, the user interface 1701 of Figure 17 may appear. Here, in response to the selection of the notification 1601, the canvas has panned automatically to show a temporary area 1702 just to the right of the activation controls, automatically populated the temporary area 1702 with a part 1703 corresponding to the notification 1601, and automatically selects the part 1703 so that the blade 1704 appears. Such a response might be performed whenever an initiating blade of journey is launched by means other than selection of one of the activation controls 511. If the user were to close the blade 1704, the temporary area 1702 with the temporary part 1703 may remain, giving the user the option of pinning that part into the activation controls 511. In this description, anything described as being pinned to the activation controls 511 may likewise be pinned to the navigation page 702 or to a blade. For instance, a part may be moved from one blade to another.

[0095] Users often want to create one artifact and then bundle it together with others. For instance, the user might want to create a website and then bundle the website with a database. Conventionally, this is often done in a one at a time fashion. The user creates a website, then the database, and then links them together. Or, perhaps the user picks a

website template that automatically has a website and database in it. If a new component comes along, each template that is to incorporate the component needs to be updated or the user is burdened with one more thing to link together.

[0096] In accordance with embodiments described herein, the user may create an item, and then ask the system what are the other commonly added items associated with the item they are creating. The user can then select those items, choosing to bundle them together. Bundling means they will automatically be created in the right order (e.g. the database has to be created before a caching service can be linked to it) and then automatically linked together. This saves the user a tremendous amount of time, and also gives the user the opportunity to improve the overall functionality of the resource. An example walkthrough of this will be described with respect to Figures 18 through 22.

[0097] Figure 18 illustrates a user interface 1800 that includes a number of activation controls. Suppose that the user selects the activation control 1801 in order to create a new web site. The user interface 1900 of Figure 19 might then appear. A create website pane 1901 appears that allows the user to enter in values for the web site. However, the website pane 1901 also presents also add options 1902 for possible resources that could help augment the performance of the website. Figure 20 illustrates a user interface 2000 in which the user has selected a database option. The system is aware that a database and a website will need some kind of connection string, and thus automatically selects a option 2001. A progress bar 2002 then appears showing that the user has completed one of three steps in the total package. If the user selects next, the user interface 2100 of Figure 21 appears. Here, the user enters information associated with the database. Upon selecting next, the user interface 2200 of Figure 22 appears. The user enters information associated with the link, and completes the application. The system automatically builds the application.

[0098] Accordingly, the principles described herein provide a robust interface for operating in which the user takes an intuitive journey through information, and can quickly understand the context in which he or she is working.

[0099] The present invention may be embodied in other specific forms without departing from its spirit or essential characteristics. The described embodiments are to be considered in all respects only as illustrative and not restrictive. The scope of the invention is, therefore, indicated by the appended claims rather than by the foregoing description. All changes which come within the meaning and range of equivalency of the claims are to be embraced within their scope.

CLAIMS

1. A computer program product comprising one or more computer-readable storage media having thereon computer-executable instructions that are structured such that, when executed by one or more processors of a computing system, cause the computing system to operate a user interface program that is configured to display a user interface on a display of the computing system, the user interface comprising:

a canvas that extends in an extendable direction on the user interface; and

an activation mechanism that may be activated to generate a first blade on the canvas, the first blade including a plurality of elements, each of at least some of the plurality of elements having at least one corresponding blade such that if the corresponding element is selected, the corresponding blade appears on the canvas as a second blade on the canvas.

2. The computer program product in accordance with Claim 1, wherein the second blade is contiguous the first blade in the panning direction of the canvas.

3. The computer program product in accordance with Claim 1, wherein the first blade is scrollable in a direction perpendicular to the extendable direction.

4. The computer program product in accordance with Claim 1, wherein the plurality of elements are hierarchically structured and include at a first selectable element, a parent element, that contains a plurality of child elements at least one of which being selectable, such that a blade corresponding to the parent element appears on the canvas if the parent element is selected, and such that a blade corresponding to the child element appears on the canvas if the selectable child element is selected.

5. A method for generating a user interface, the method comprising:

an act of providing a canvas that extends in a pannable direction in the user interface;

an act of presenting a first blade in a canvas at a first position of the canvas in the pannable direction, the first blade including a first plurality of elements, each of at least some of the first plurality of elements having a corresponding blade such that if the corresponding element is selected, the corresponding blade appears on the canvas as a second blade on the canvas;

an act of detecting a selection of one of the first plurality of elements; and
in response to the detection of the selection of the element in the first plurality of elements, an act of presenting a second blade in a canvas at a second position of the canvas in the pannable direction.

6. The method in accordance with Claim 5, wherein the second blade includes a second plurality of elements, each of at least some of the second plurality of elements having a corresponding blade such that if the corresponding element is selected, the corresponding blade appears on the canvas as a third blade on the canvas.

7. The method in accordance with Claim 6, further comprising:
in response to user interaction, an act presenting a viewer that allows an entire sequence of blades including at least the first, second, and third blades to be viewed on the display.

8. The method in accordance with Claim 6, further comprising:
in response to user interaction, an act of saving an entire sequence of blades including at least the first blade, the second blade, and the third blade.

9. The method in accordance with Claim 6, the first blade occupying a majority of the canvas at the first position, and the second blade occupying a majority of the canvas at the second position.

10. The method in accordance with Claim 6, wherein the first blade occupies all of the canvas at the first position, and the second blade occupies all of the canvas at the second position.

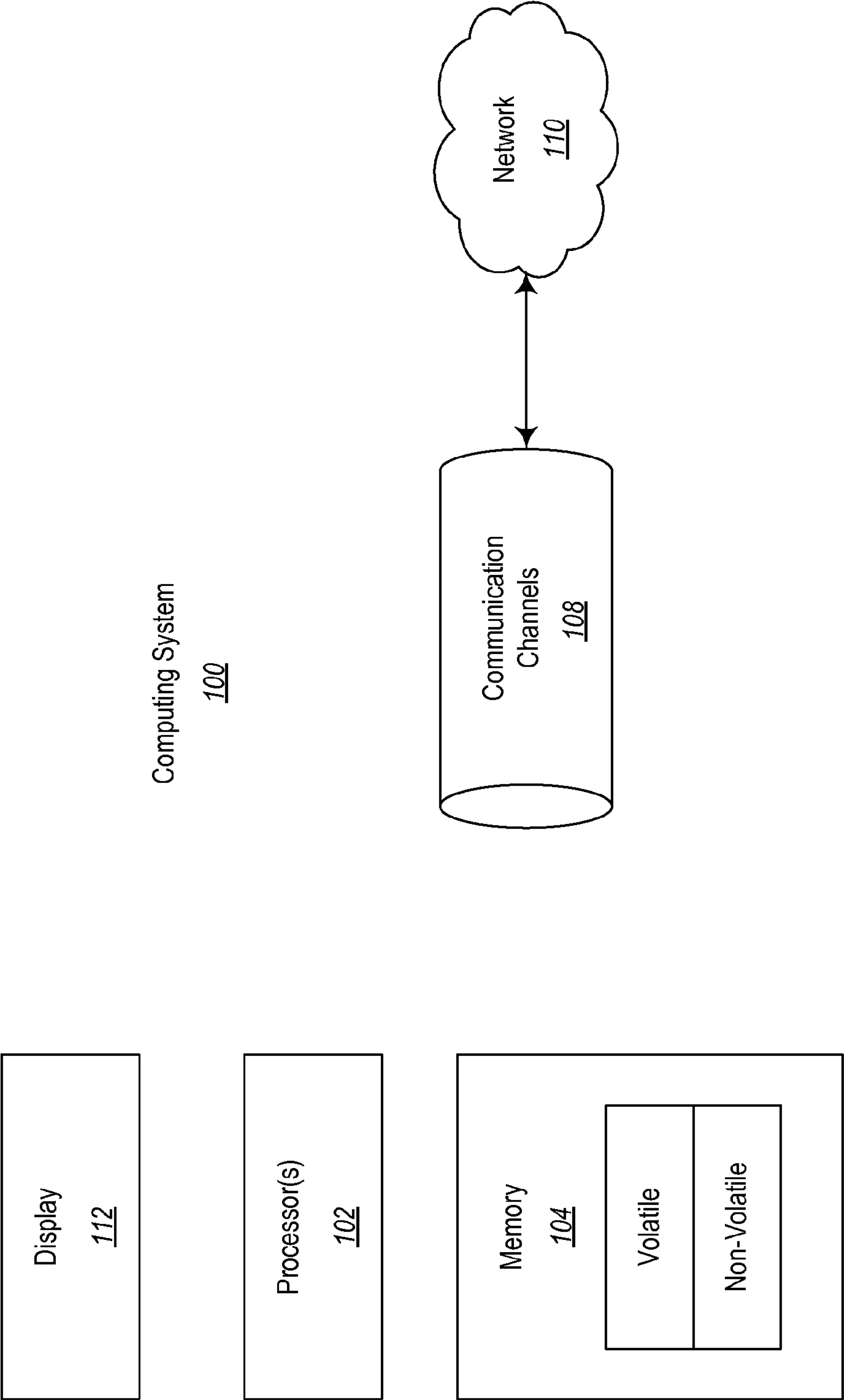


Figure 1

200A

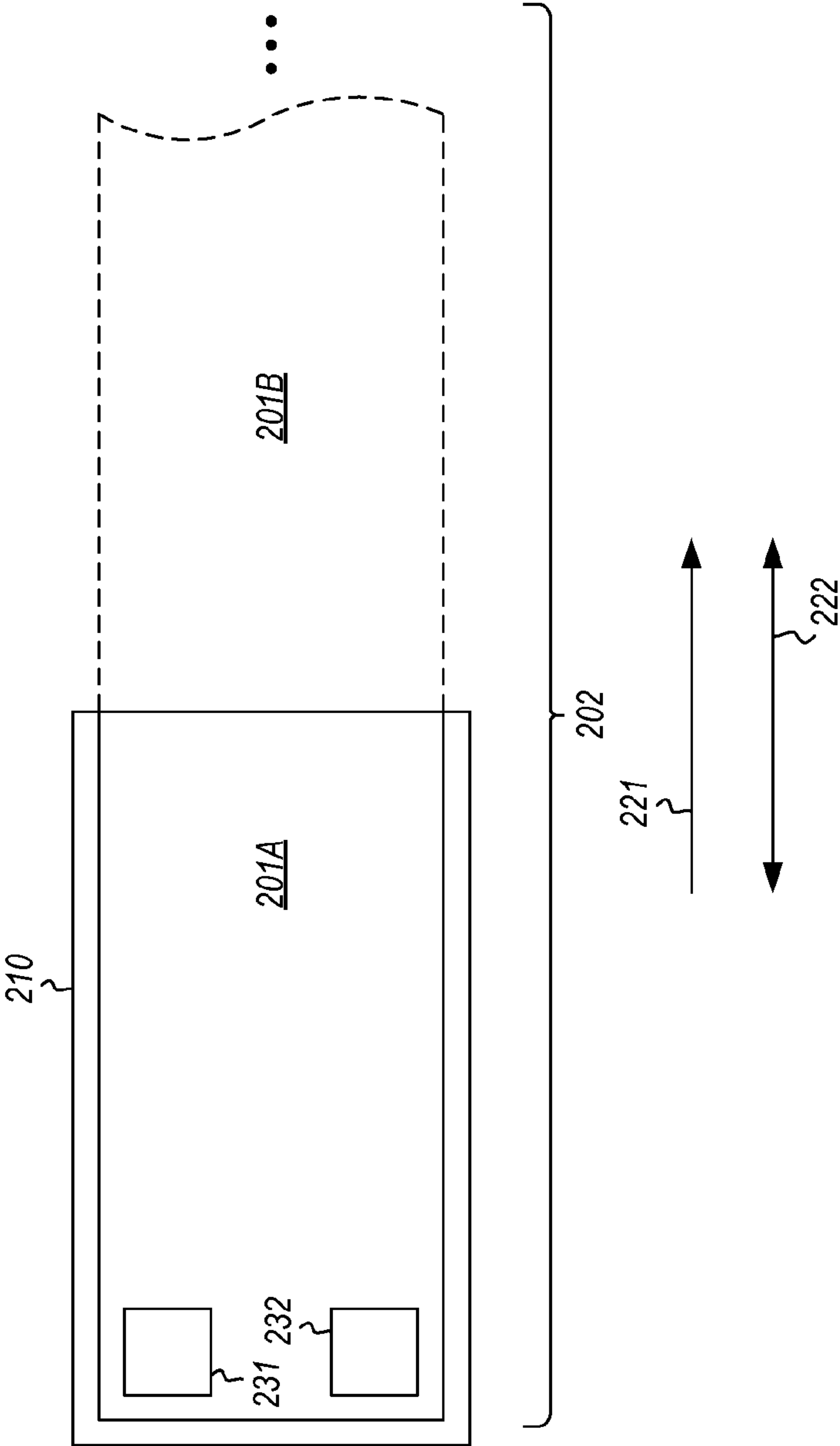


Figure 2A

200B

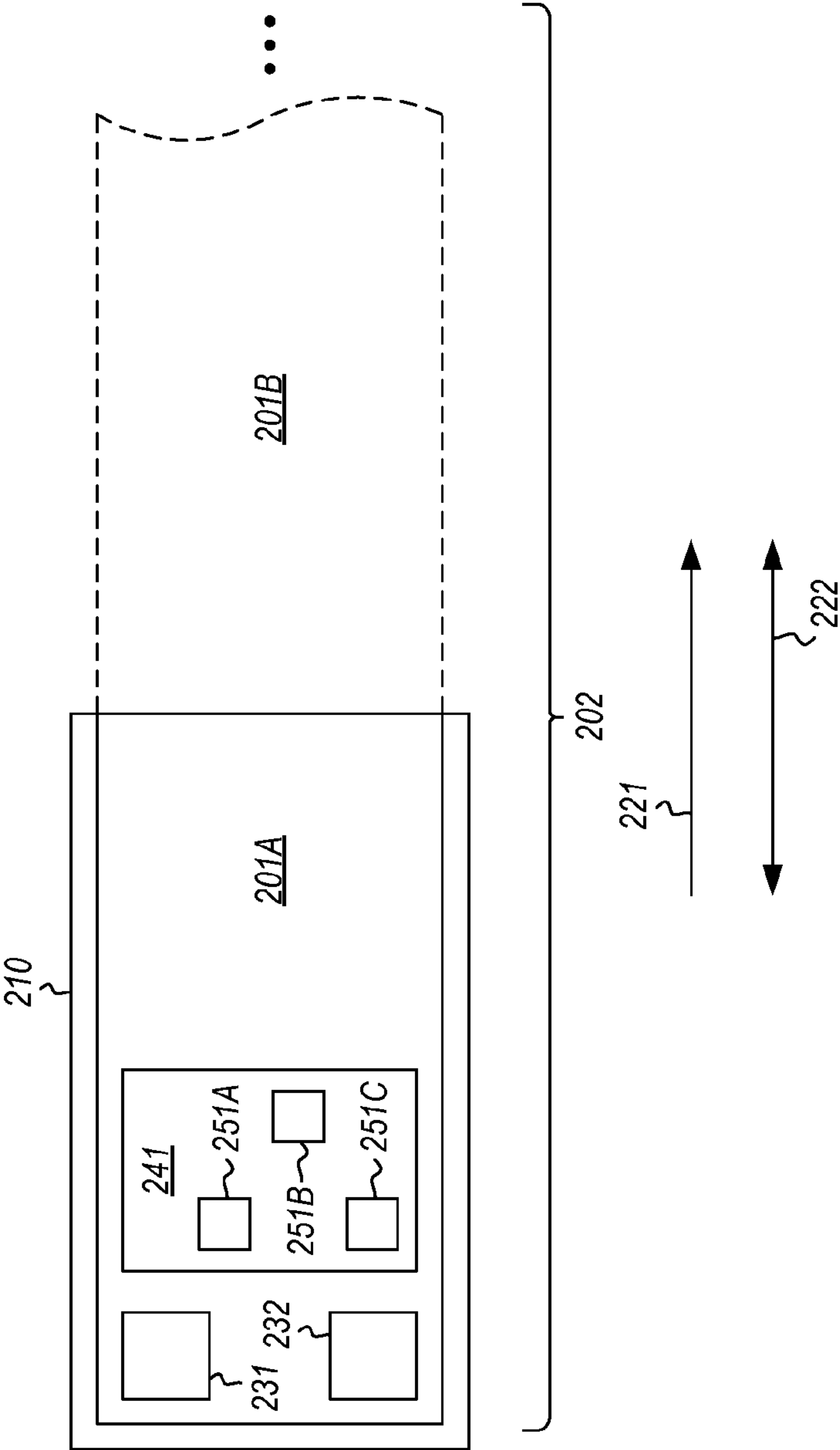


Figure 2B

200C

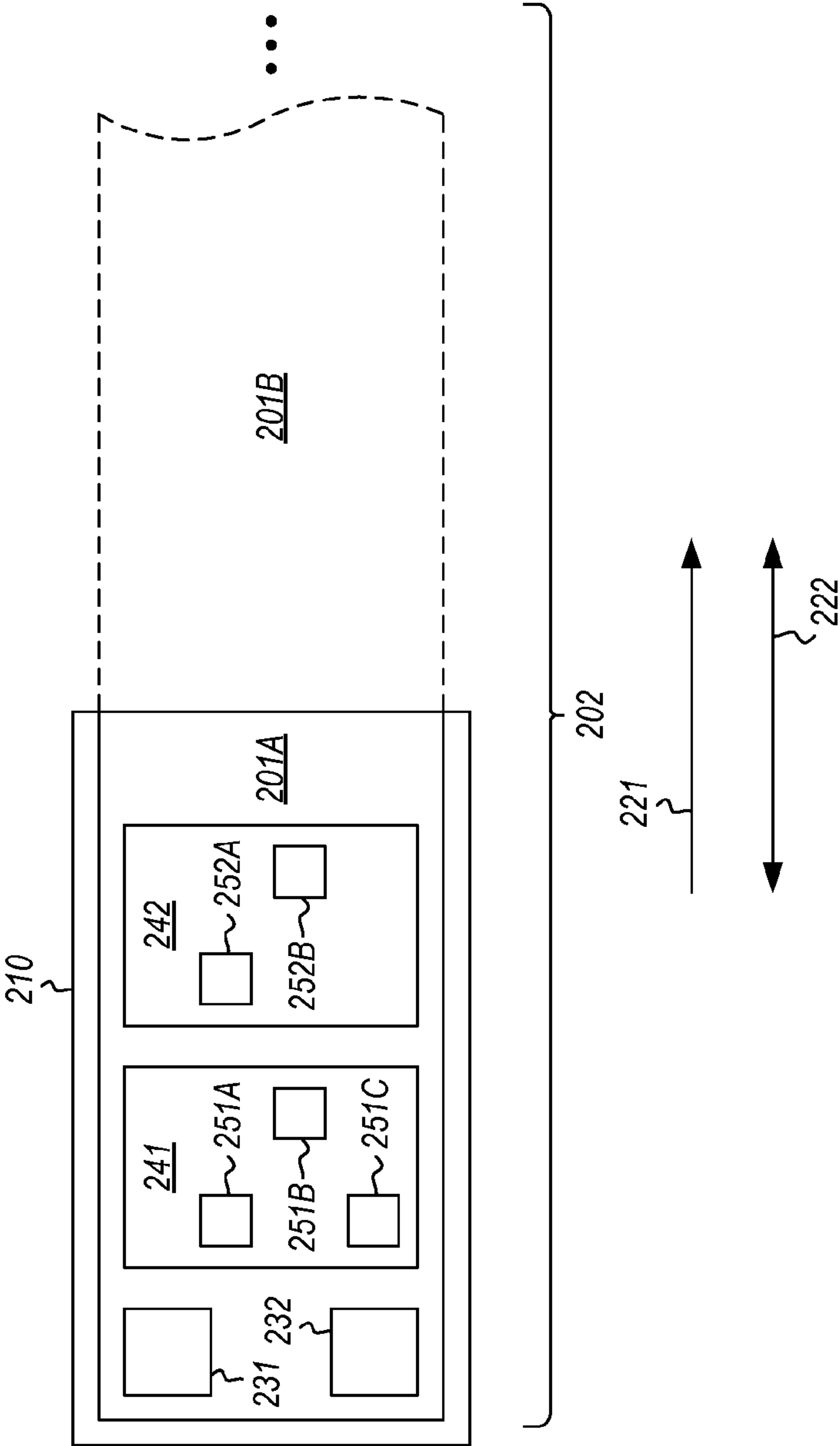


Figure 2C

300

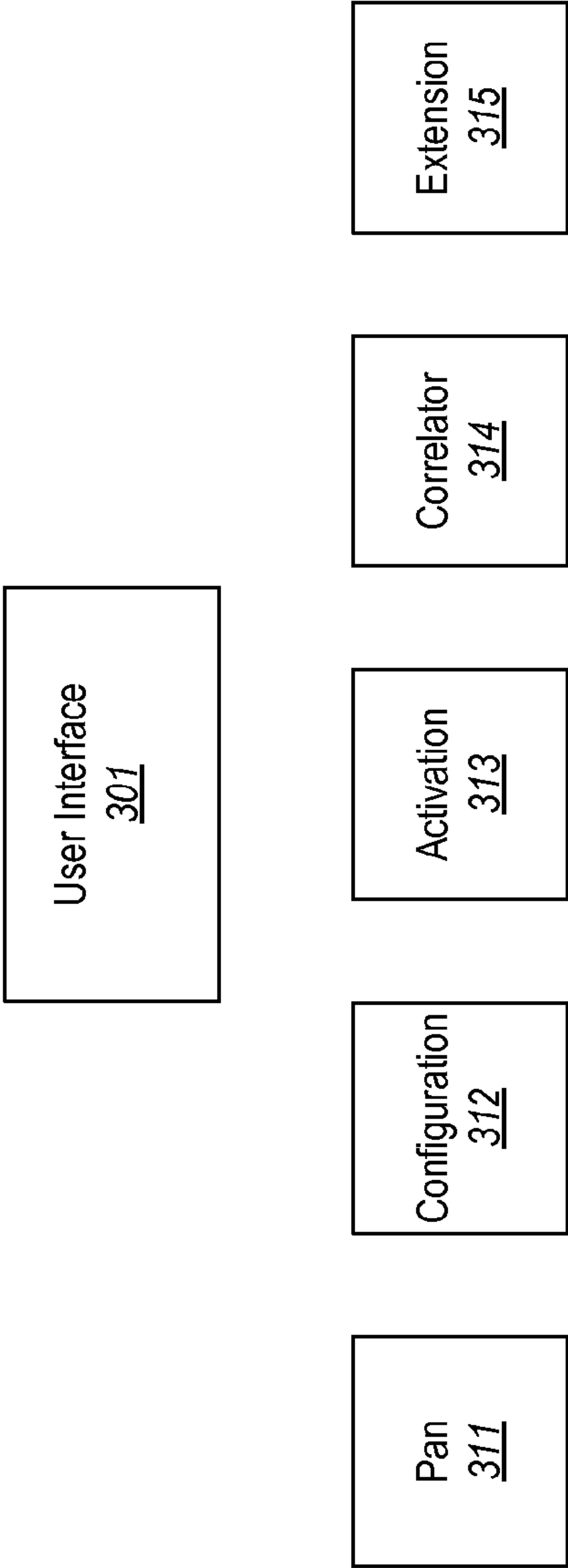
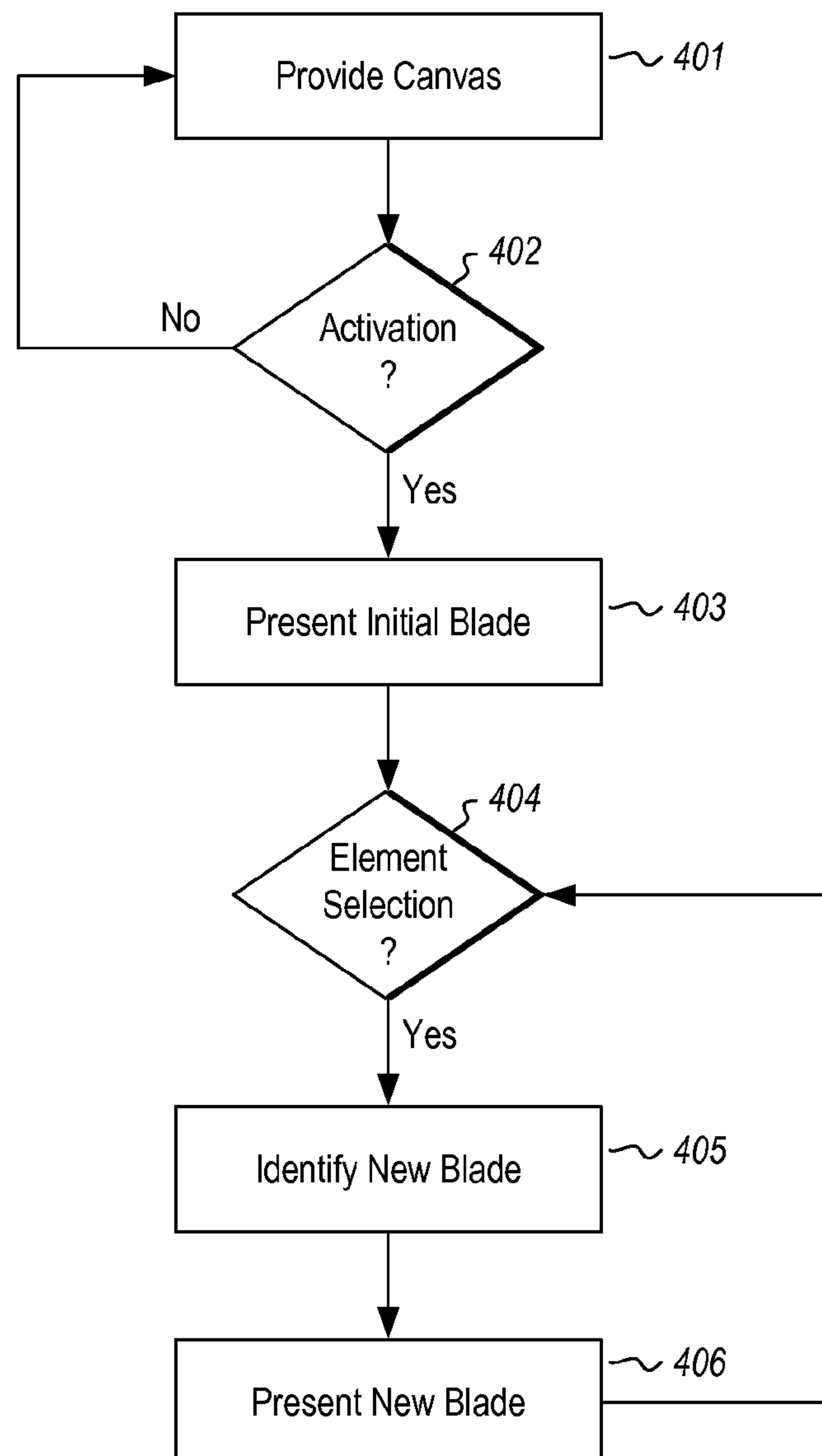


Figure 3

6 / 24400**Figure 4**

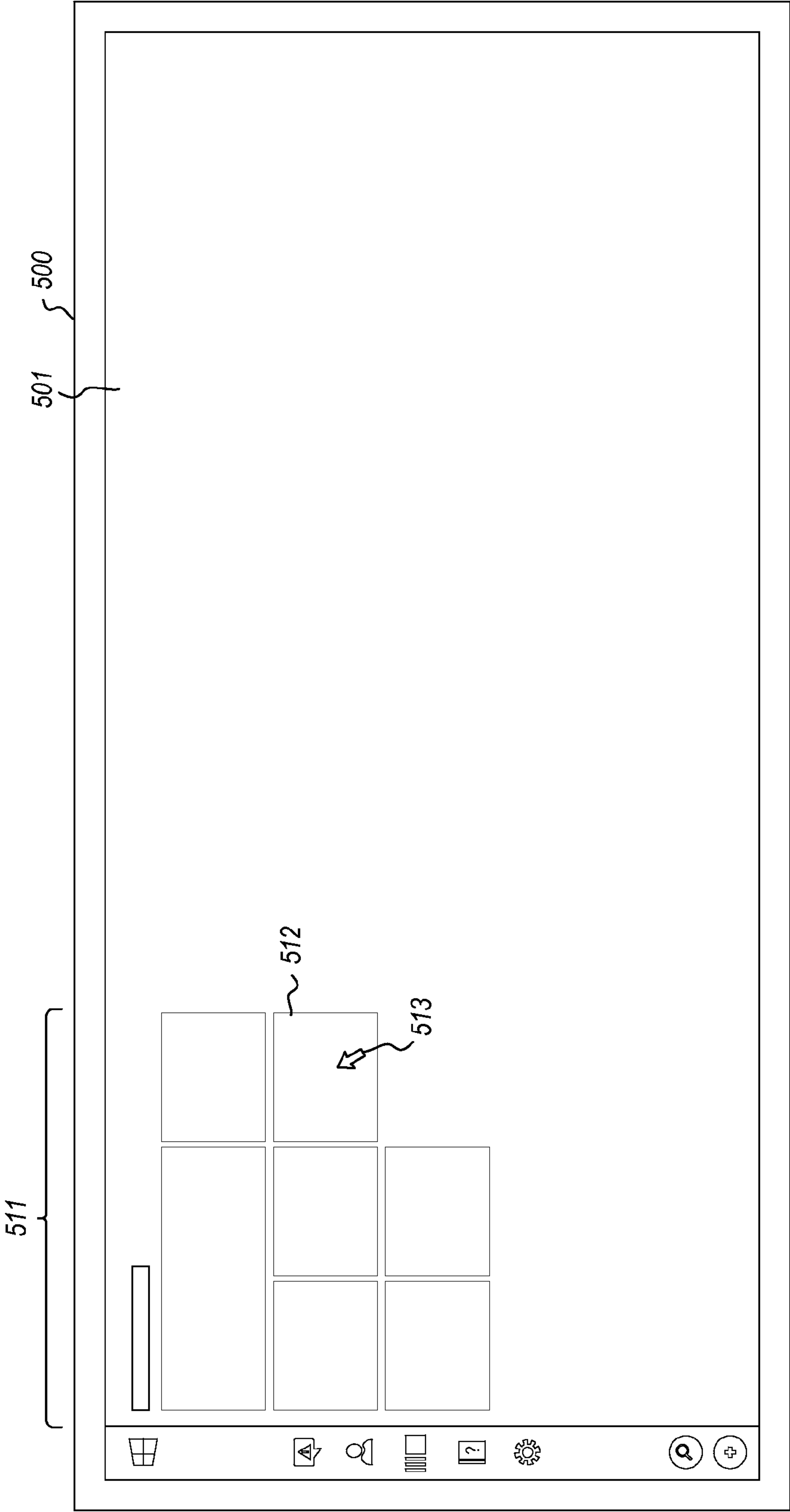


Figure 5

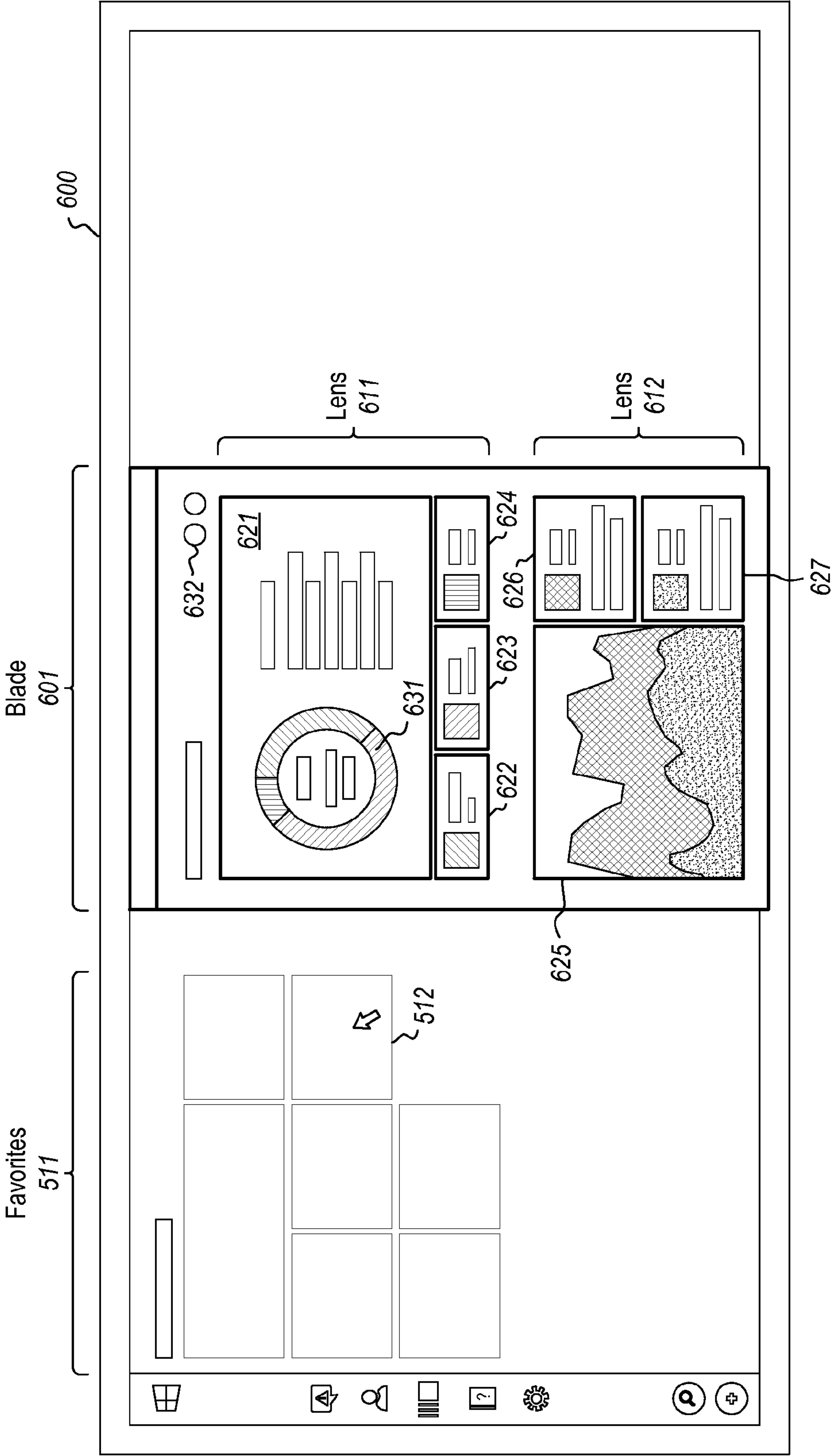


Figure 6

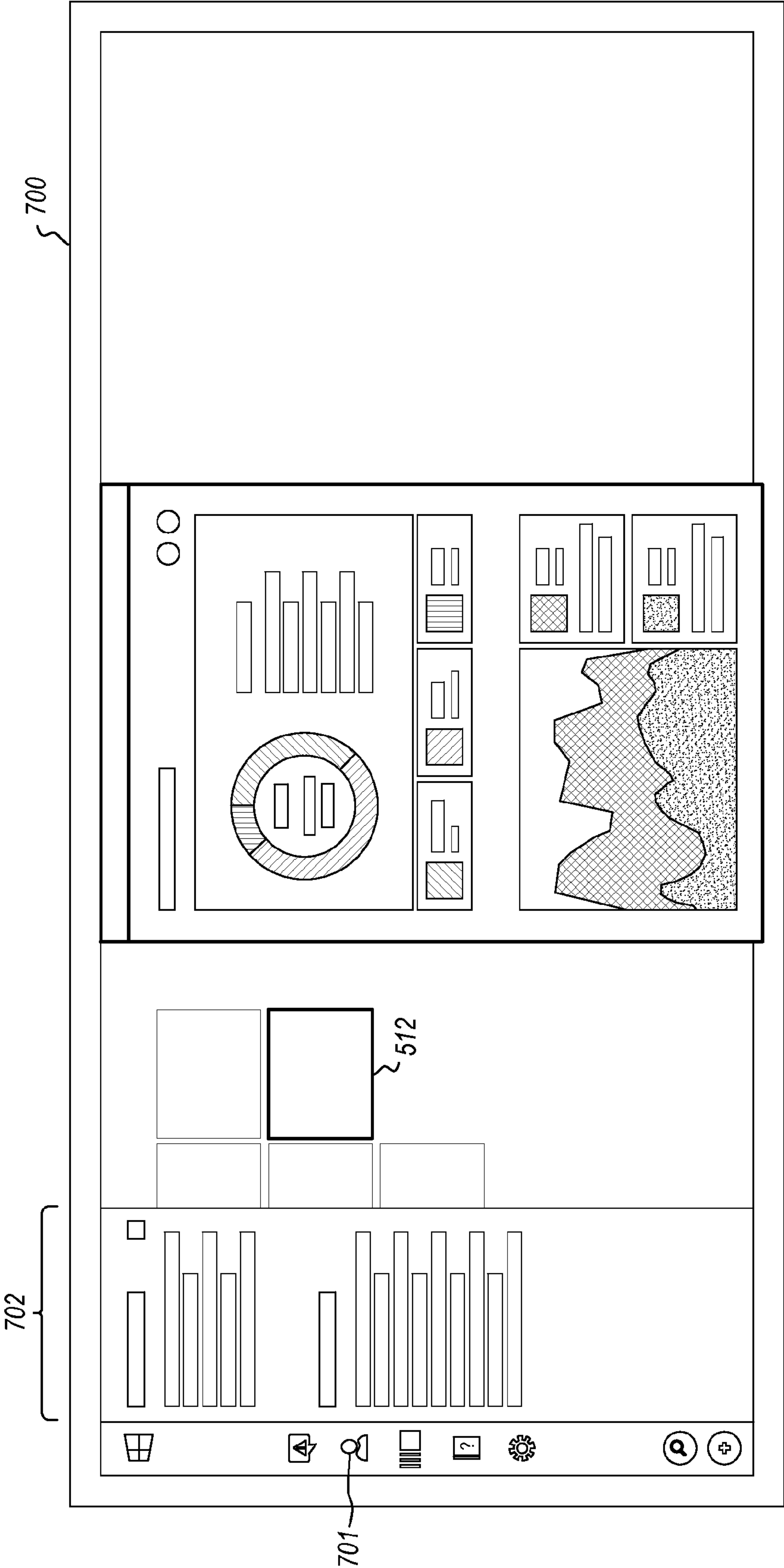


Figure 7

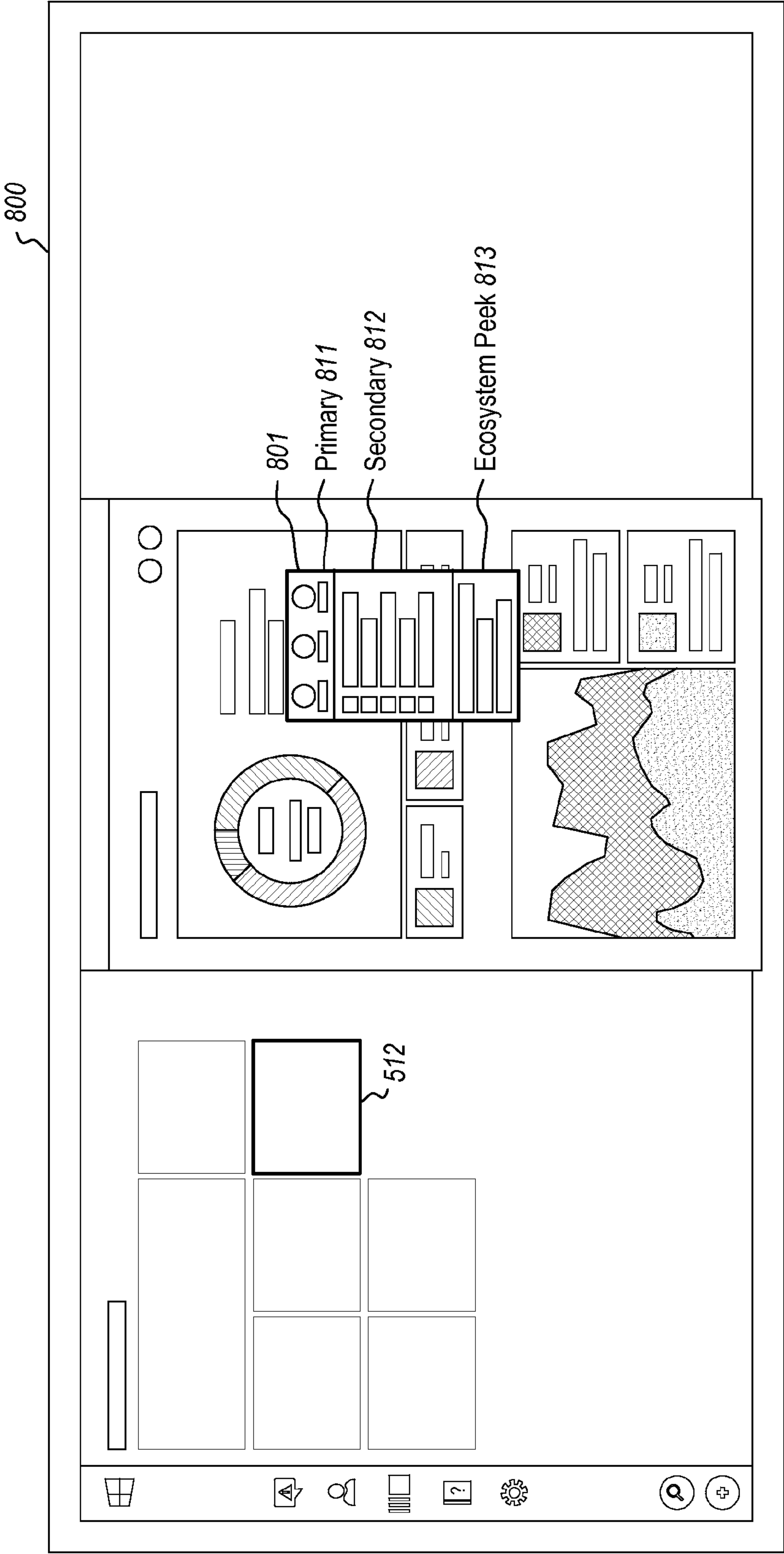


Figure 8

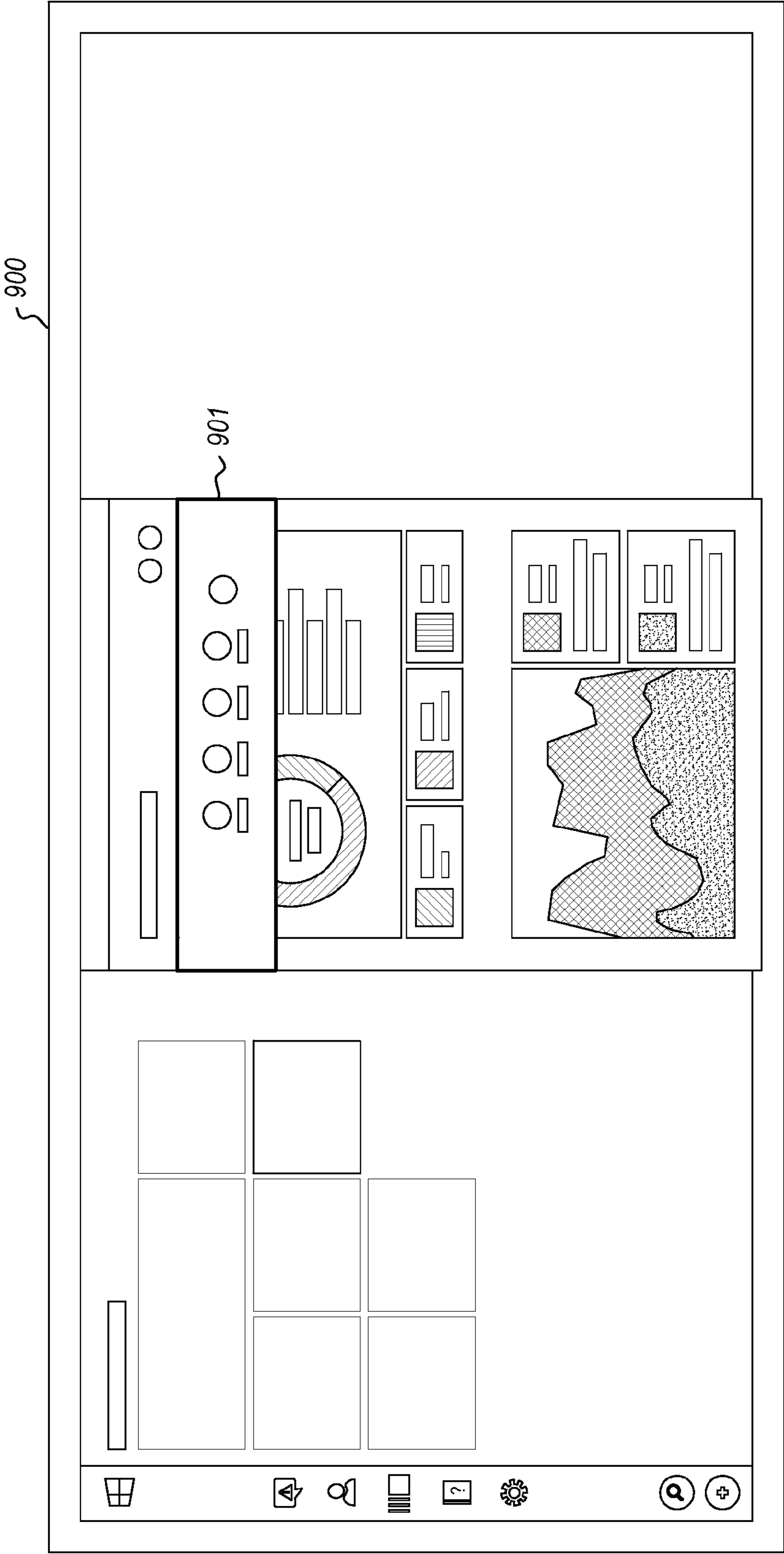


Figure 9

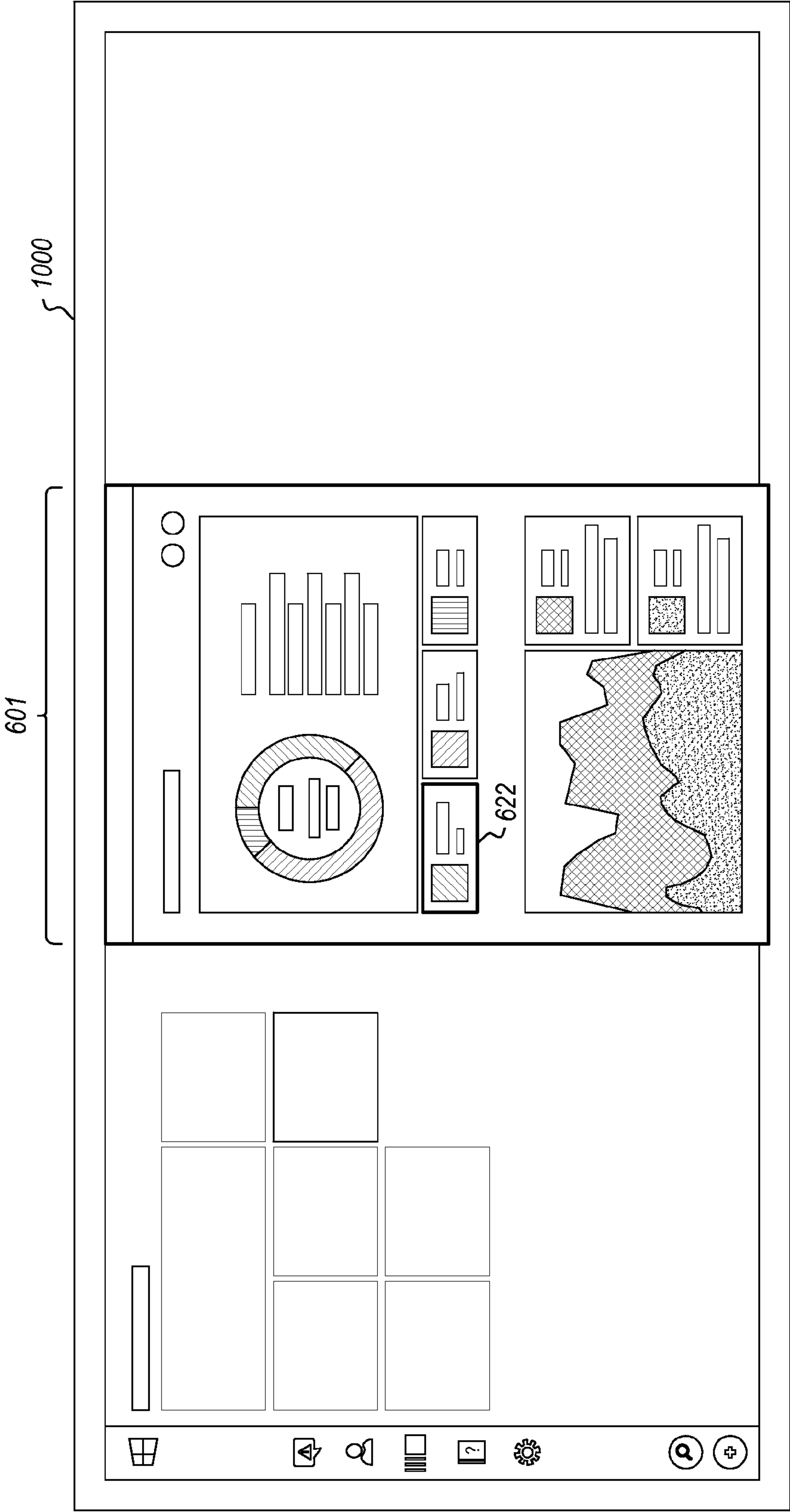


Figure 10

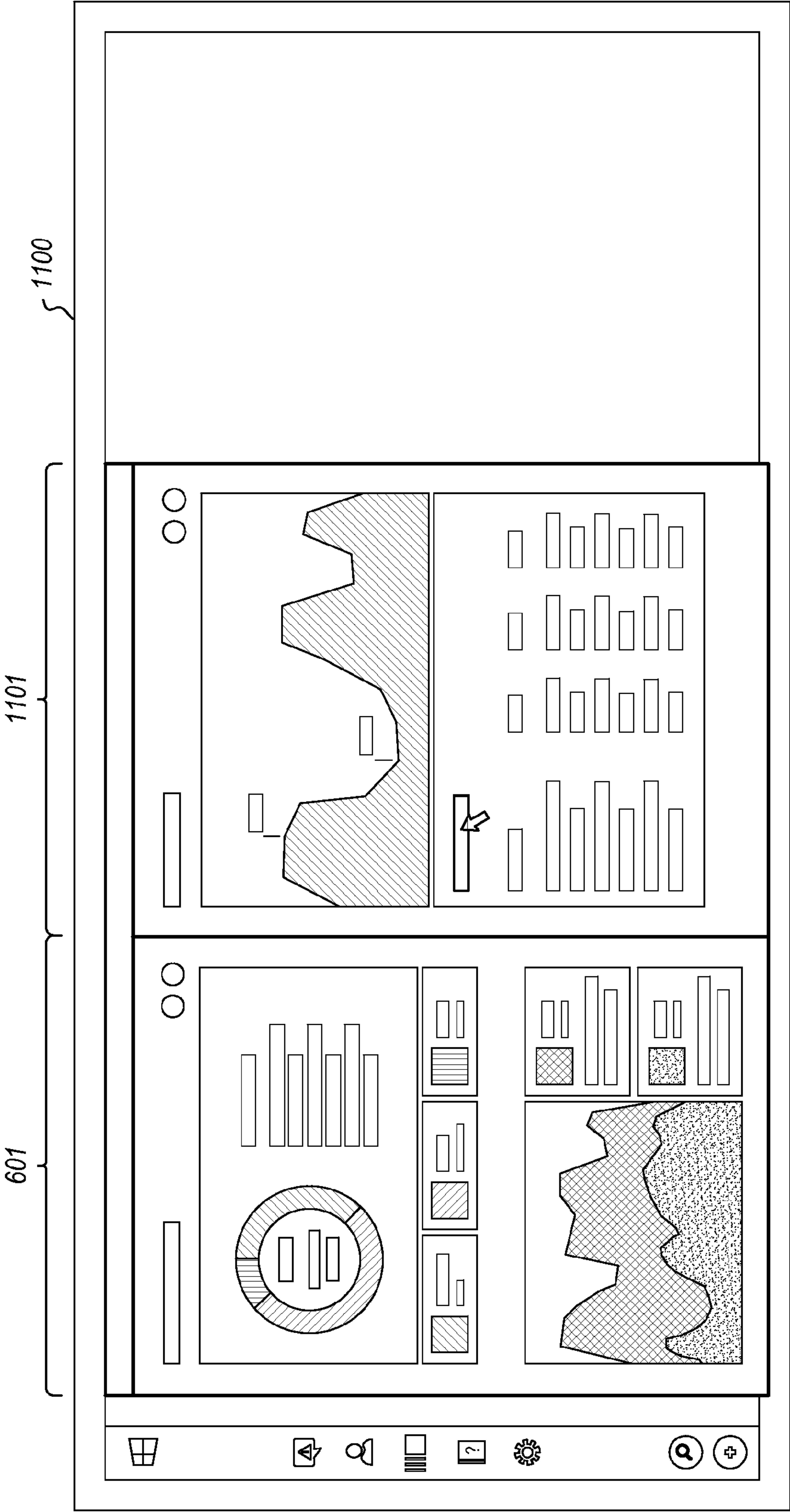


Figure 11

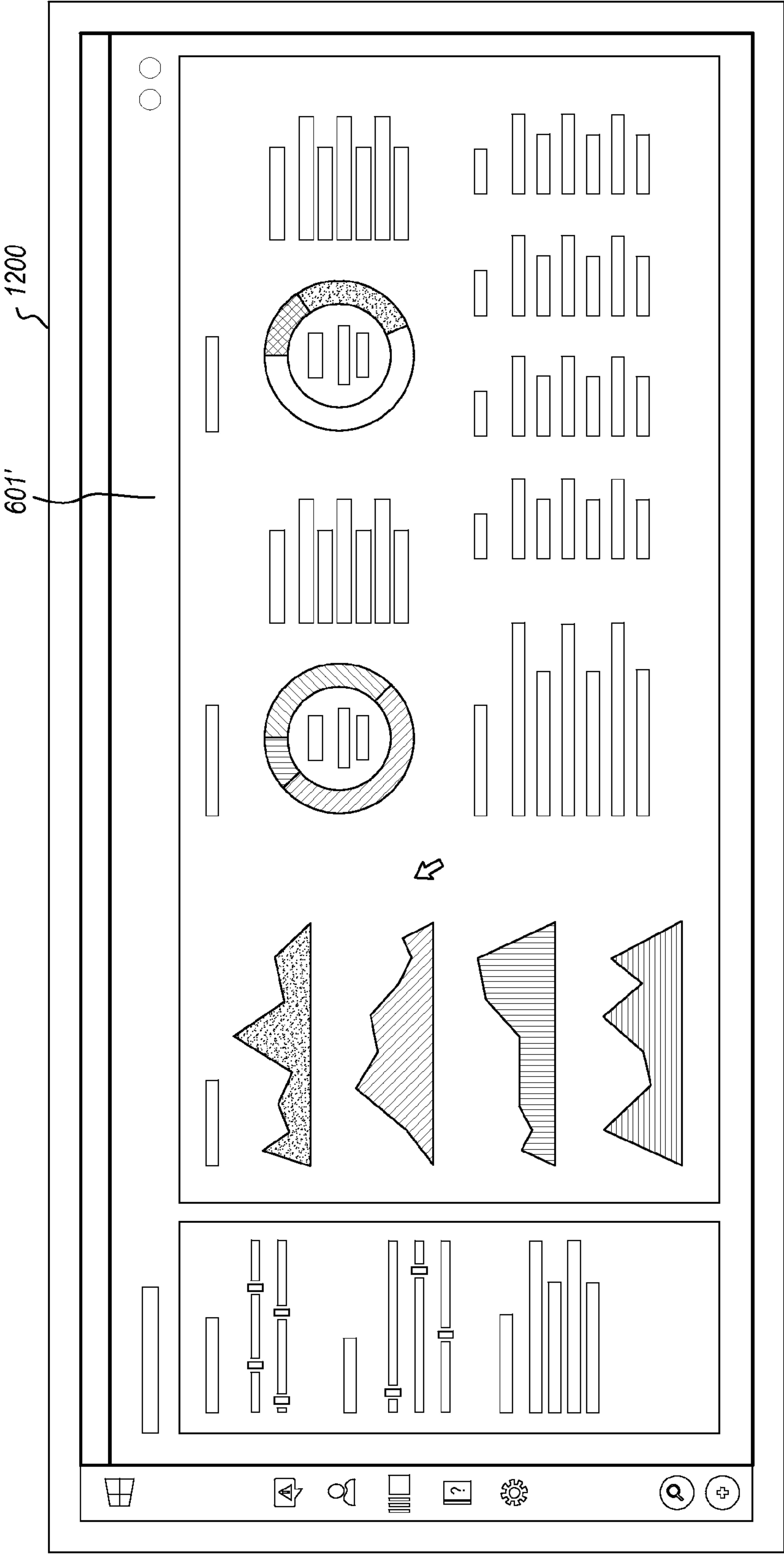


Figure 12

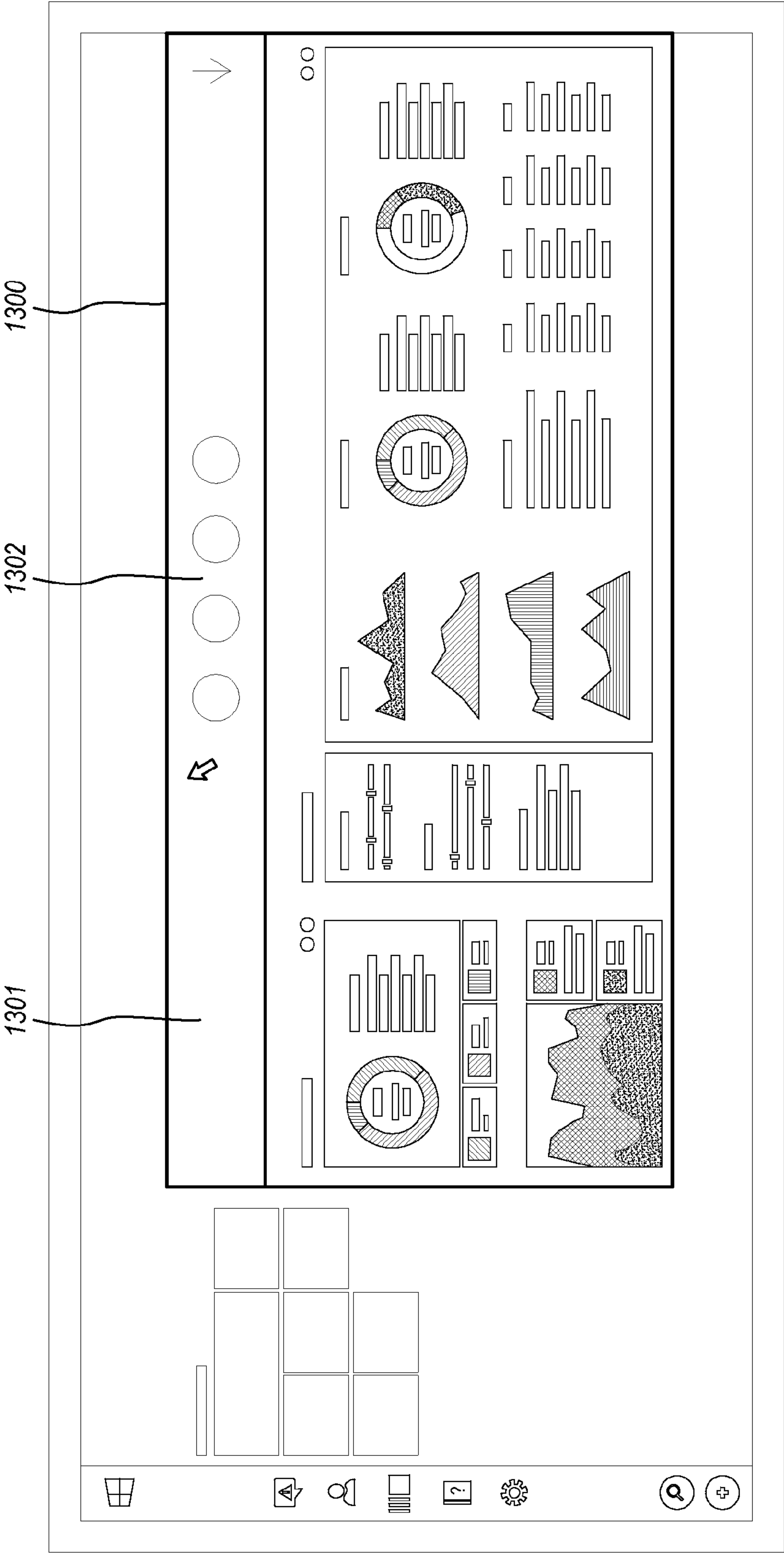
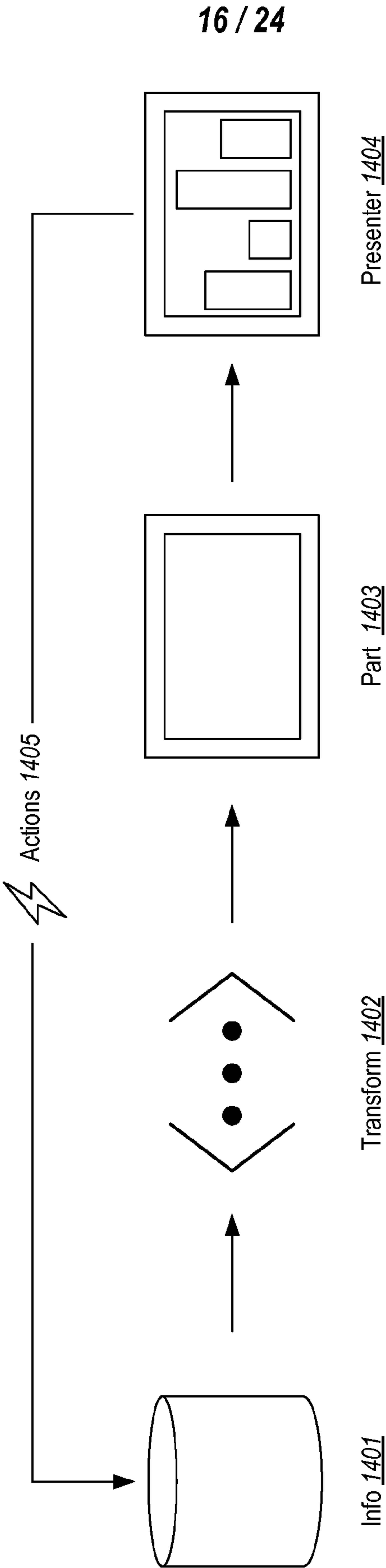


Figure 13

1400



16 / 24

Figure 14

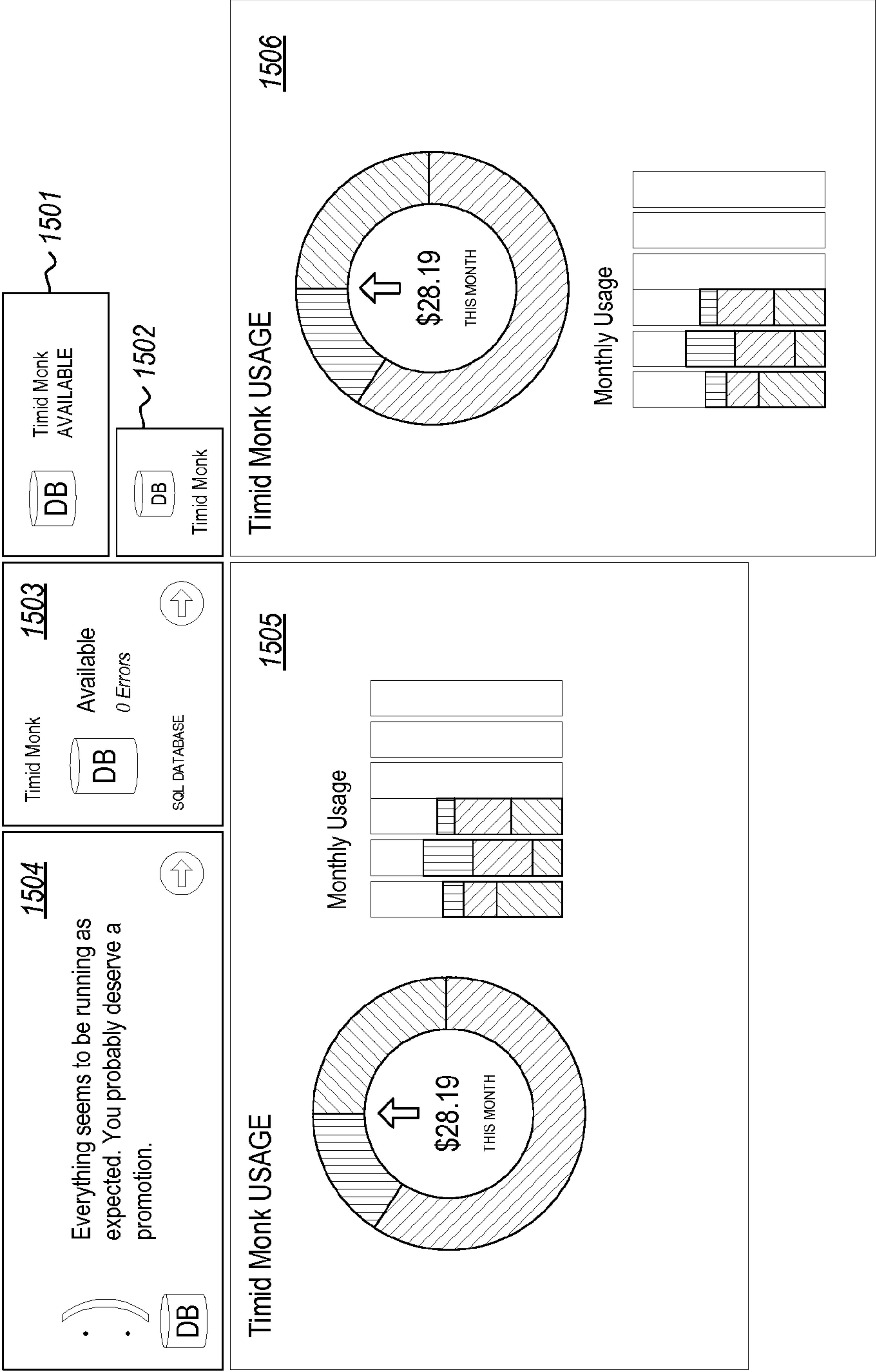


Figure 15

1600

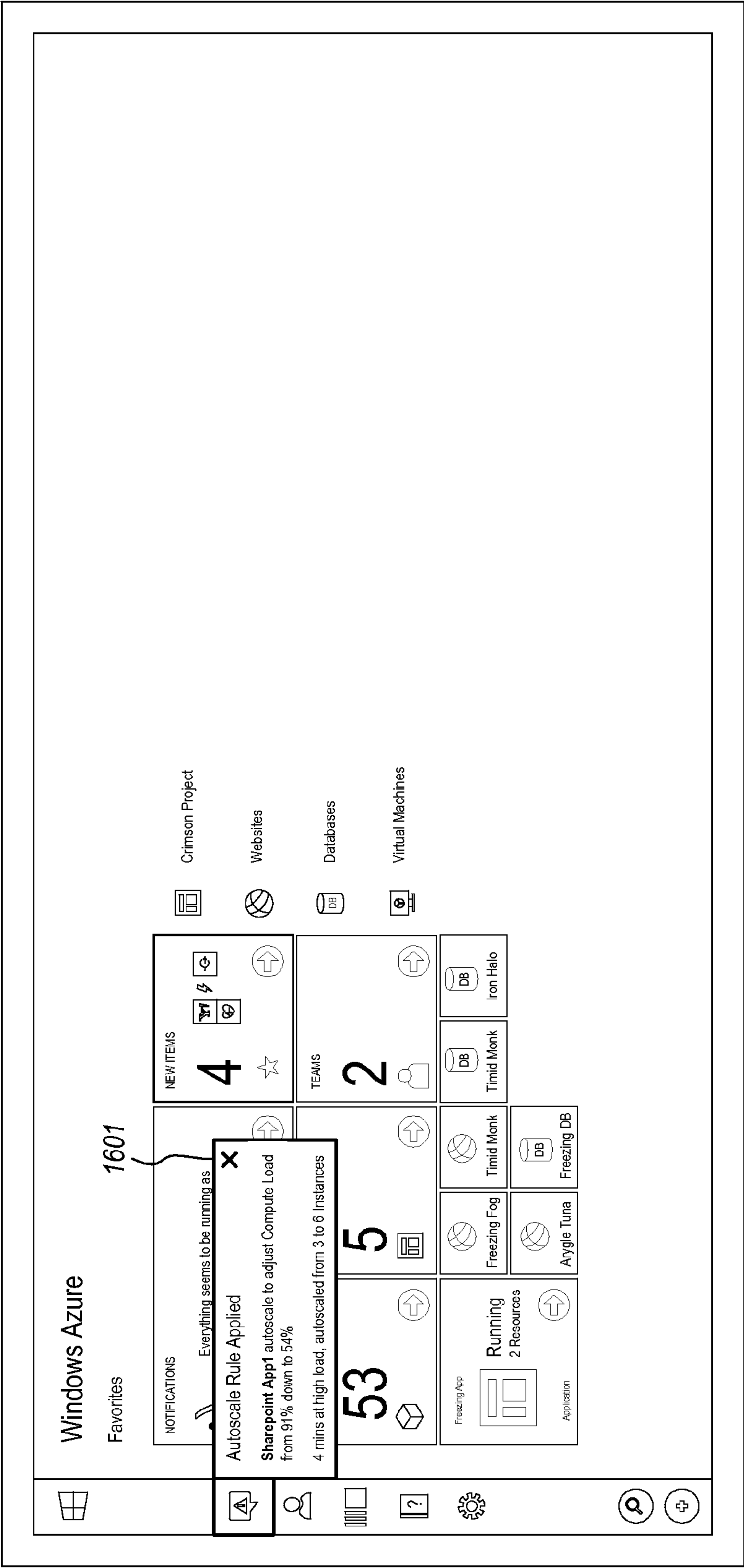


Figure 16

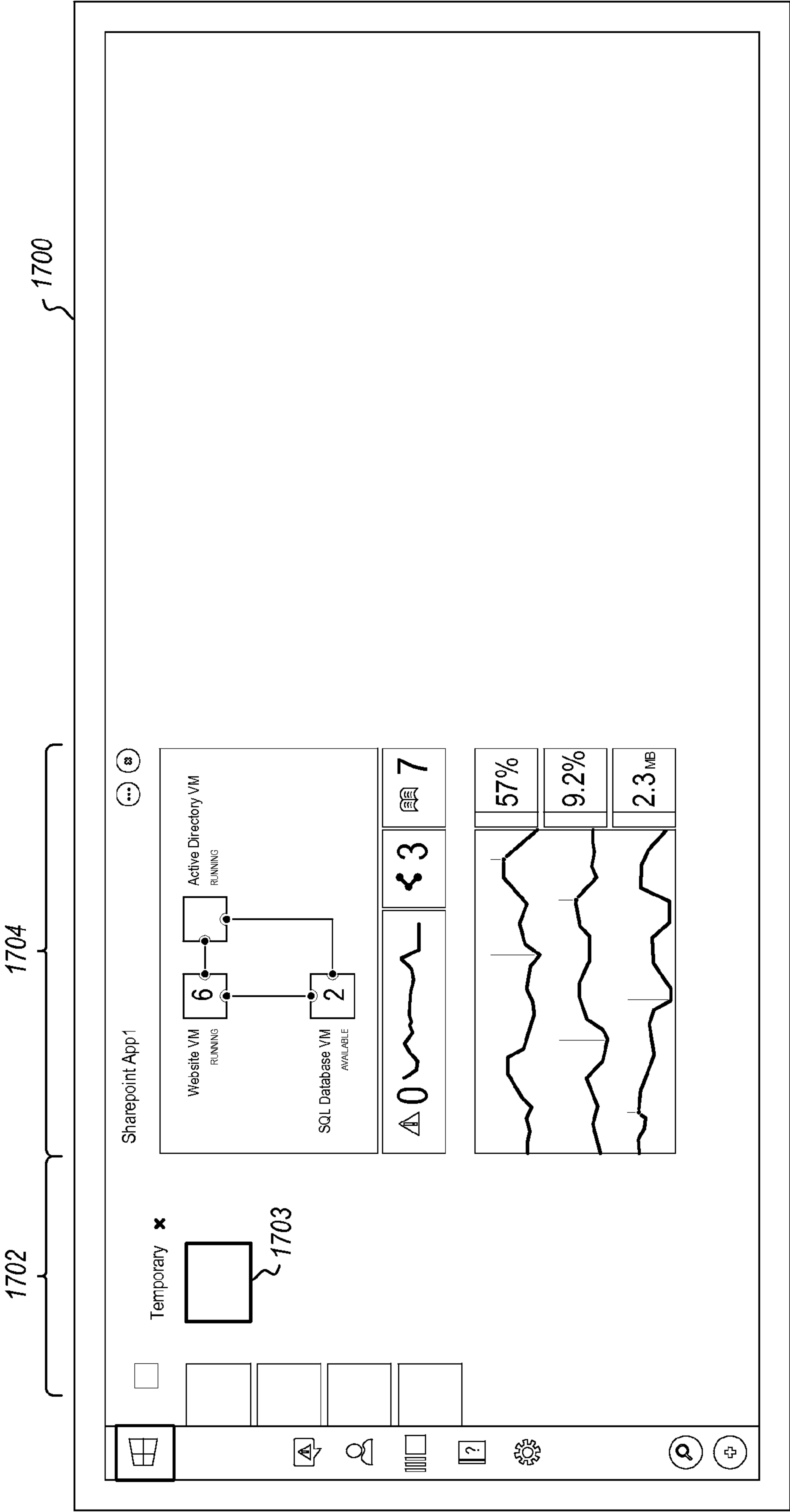


Figure 17

1800

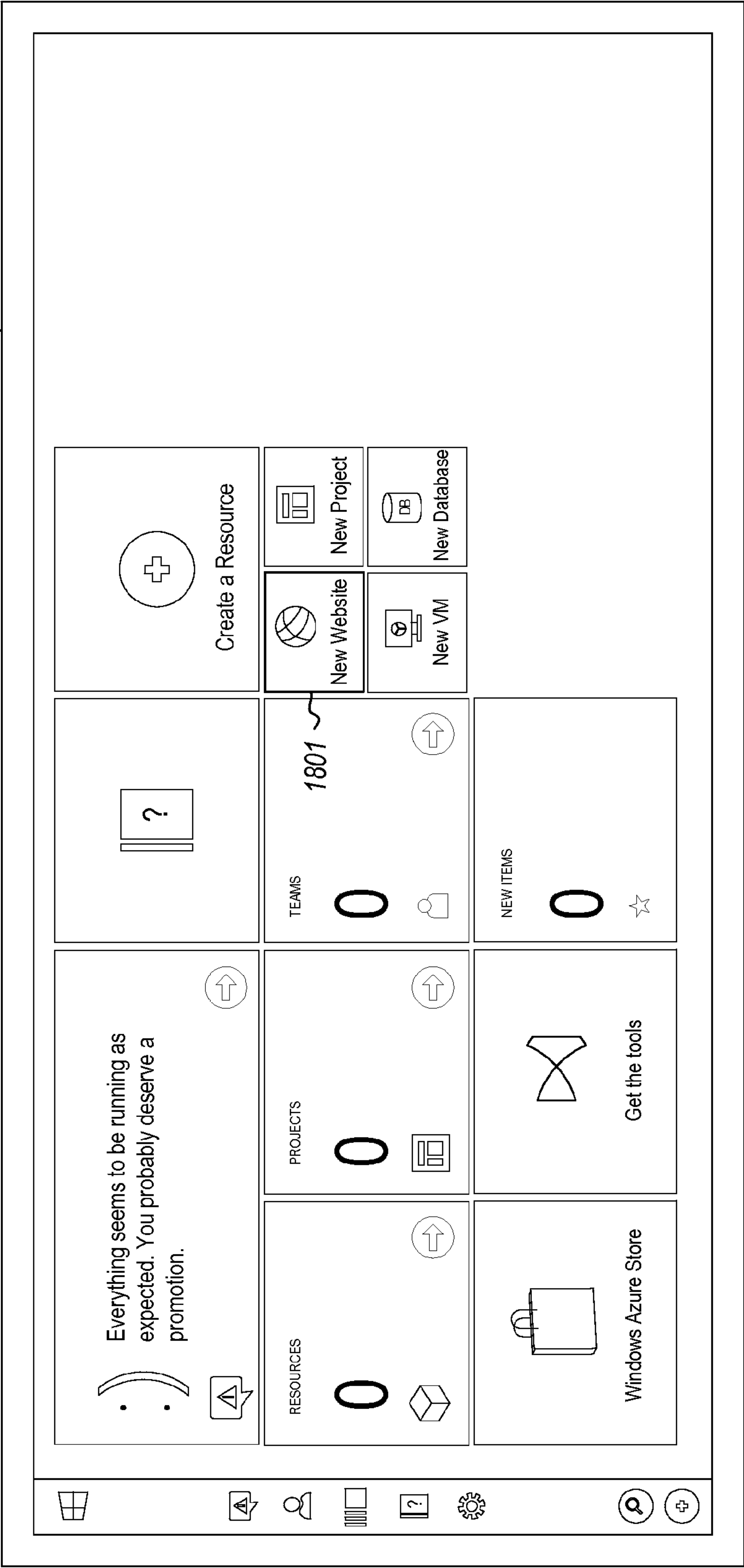


Figure 18

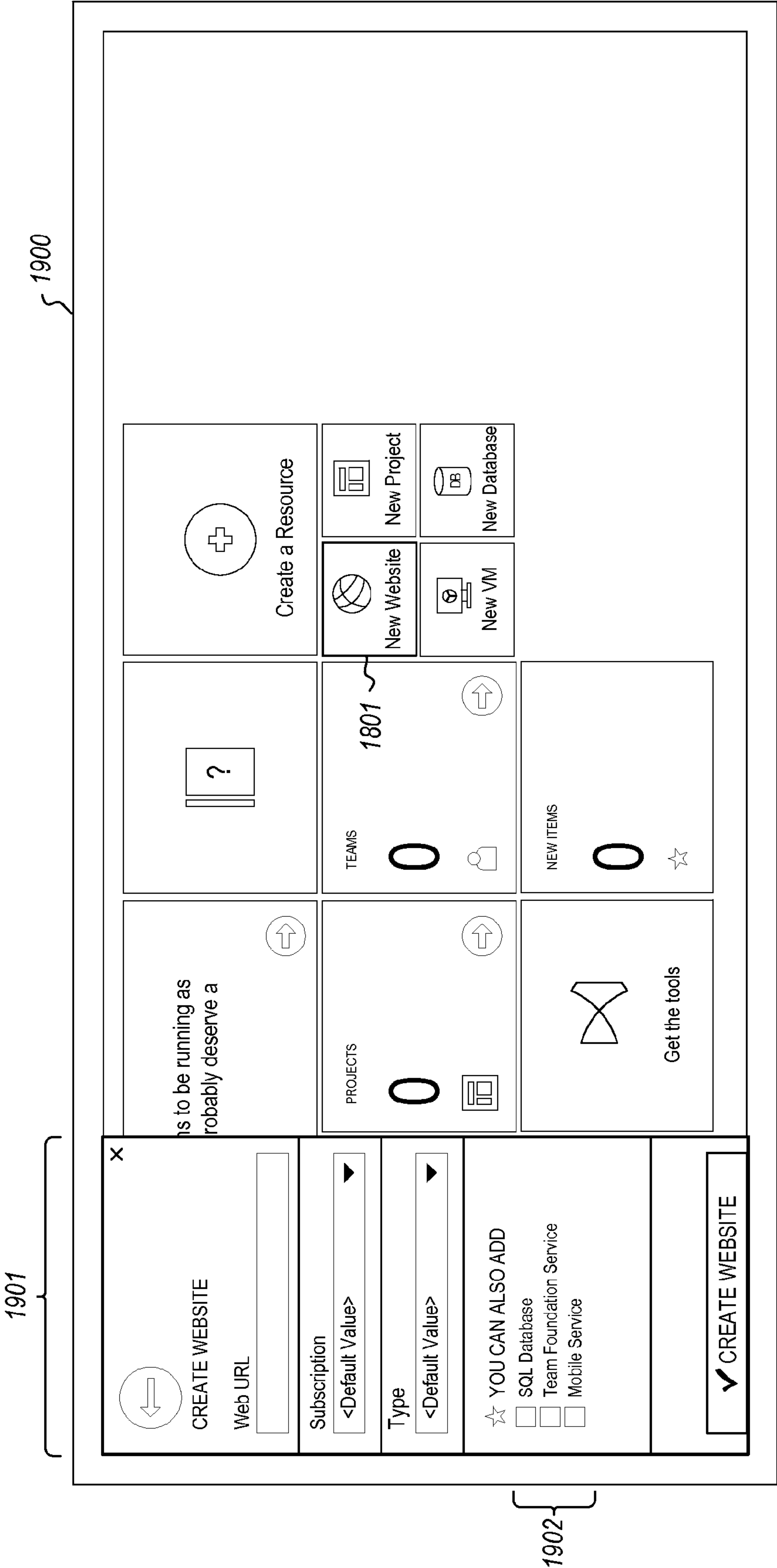


Figure 19

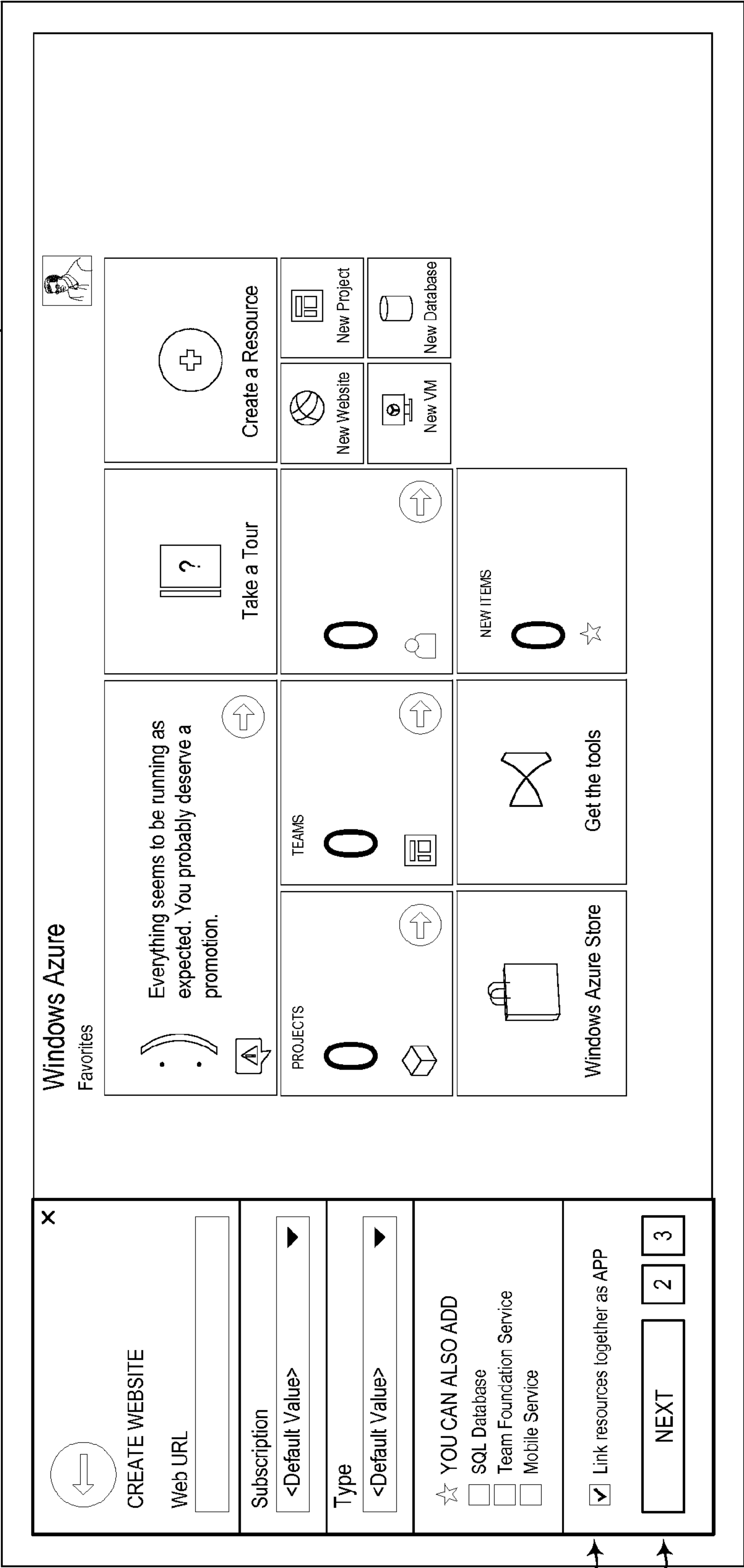


Figure 20

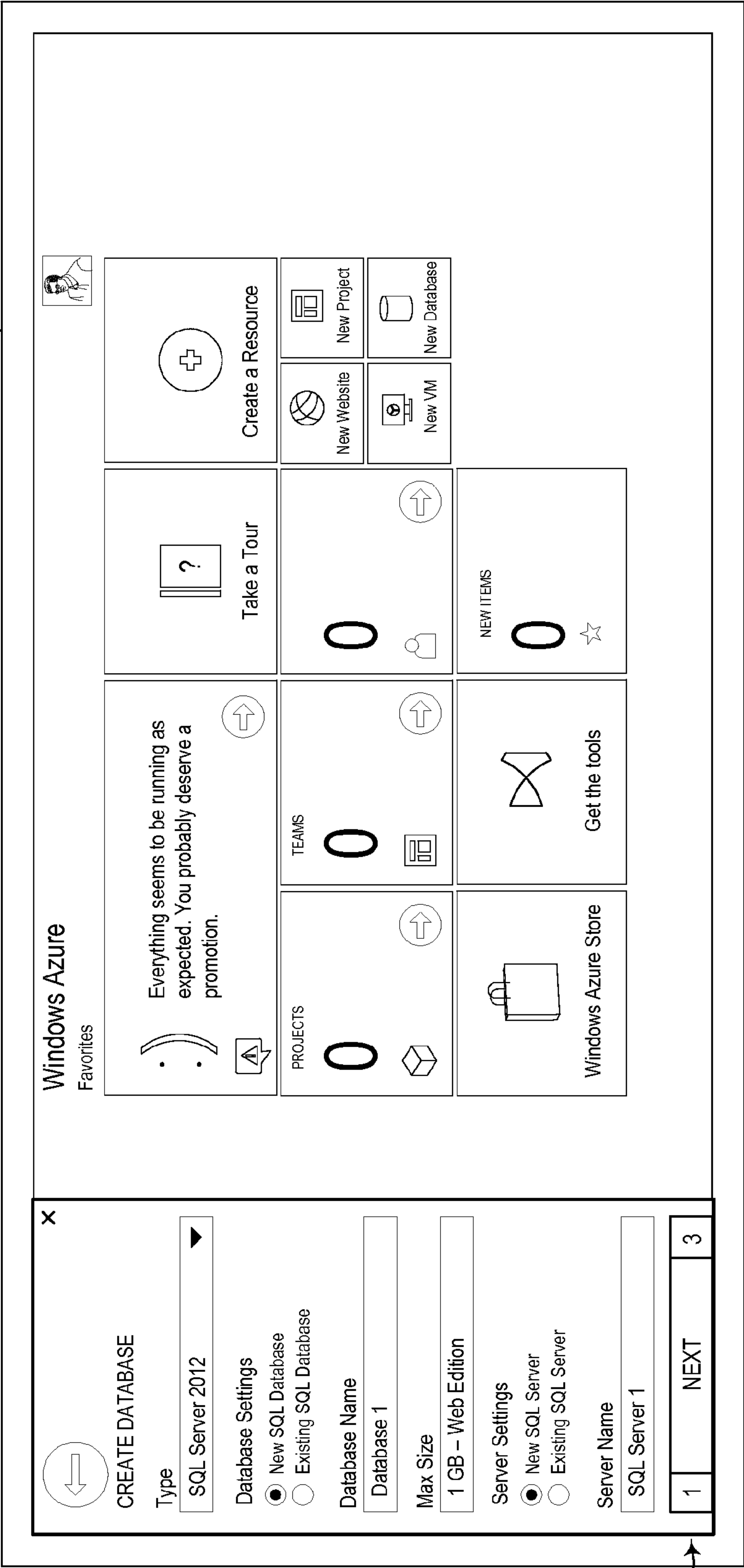


Figure 21

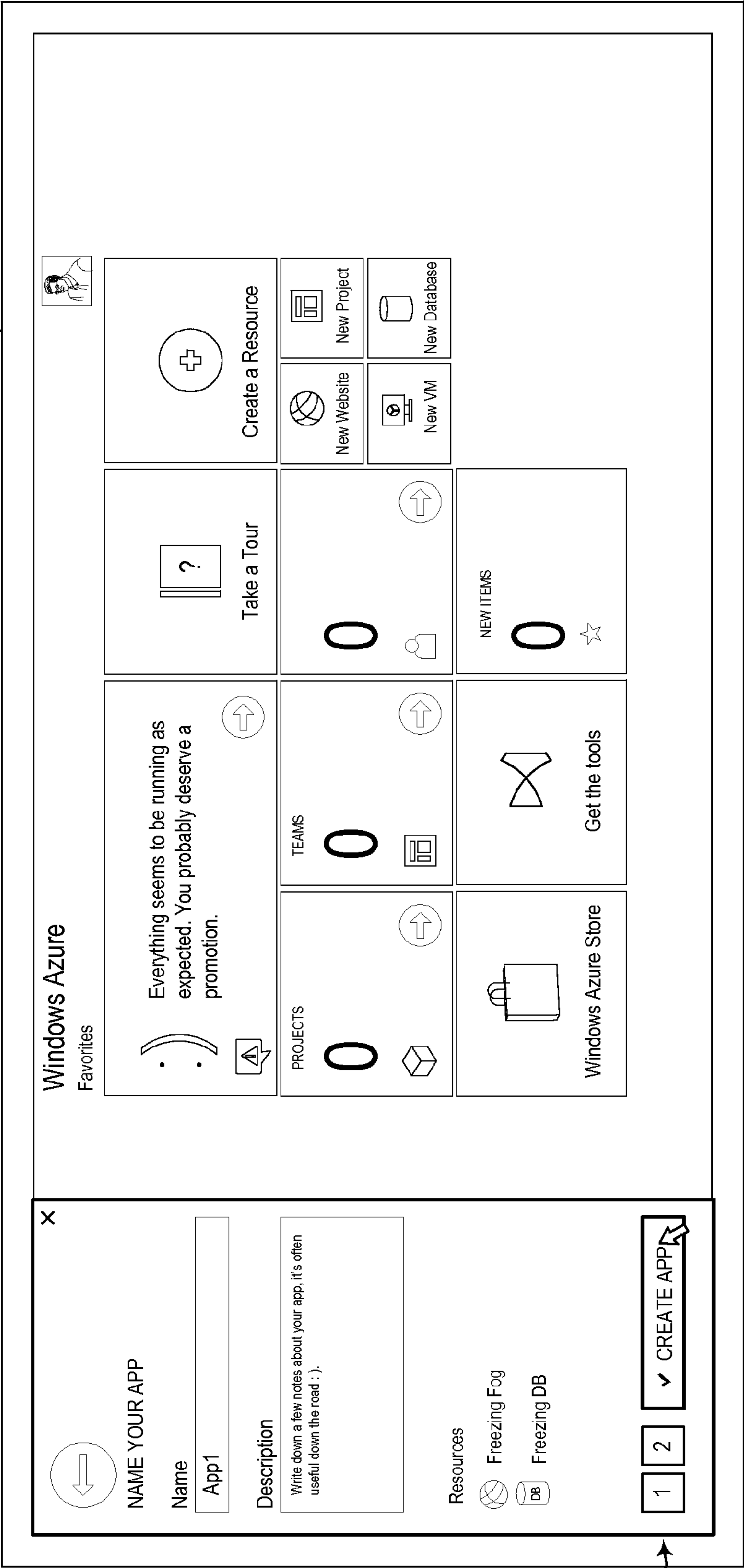


Figure 22

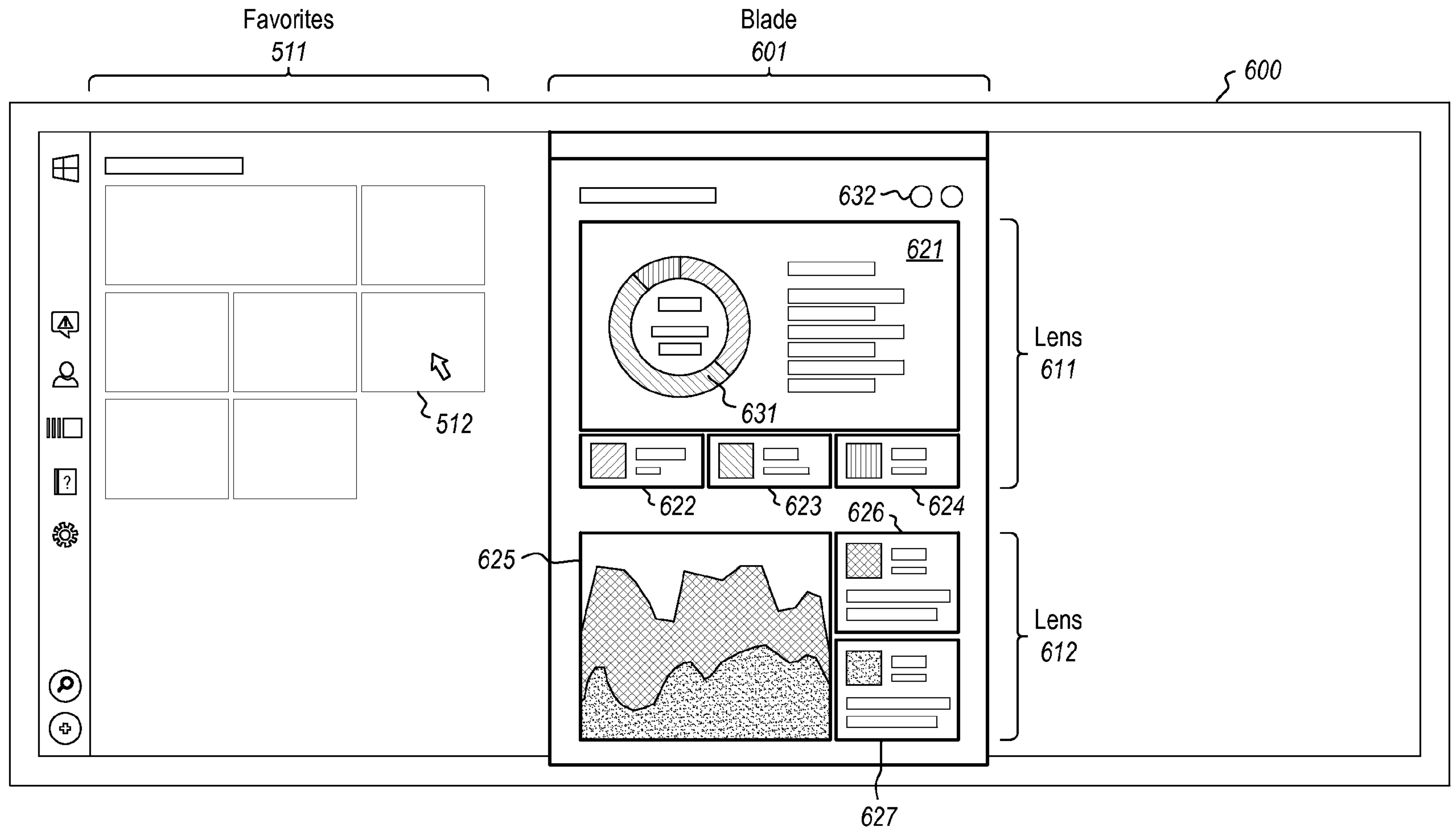


Figure 6