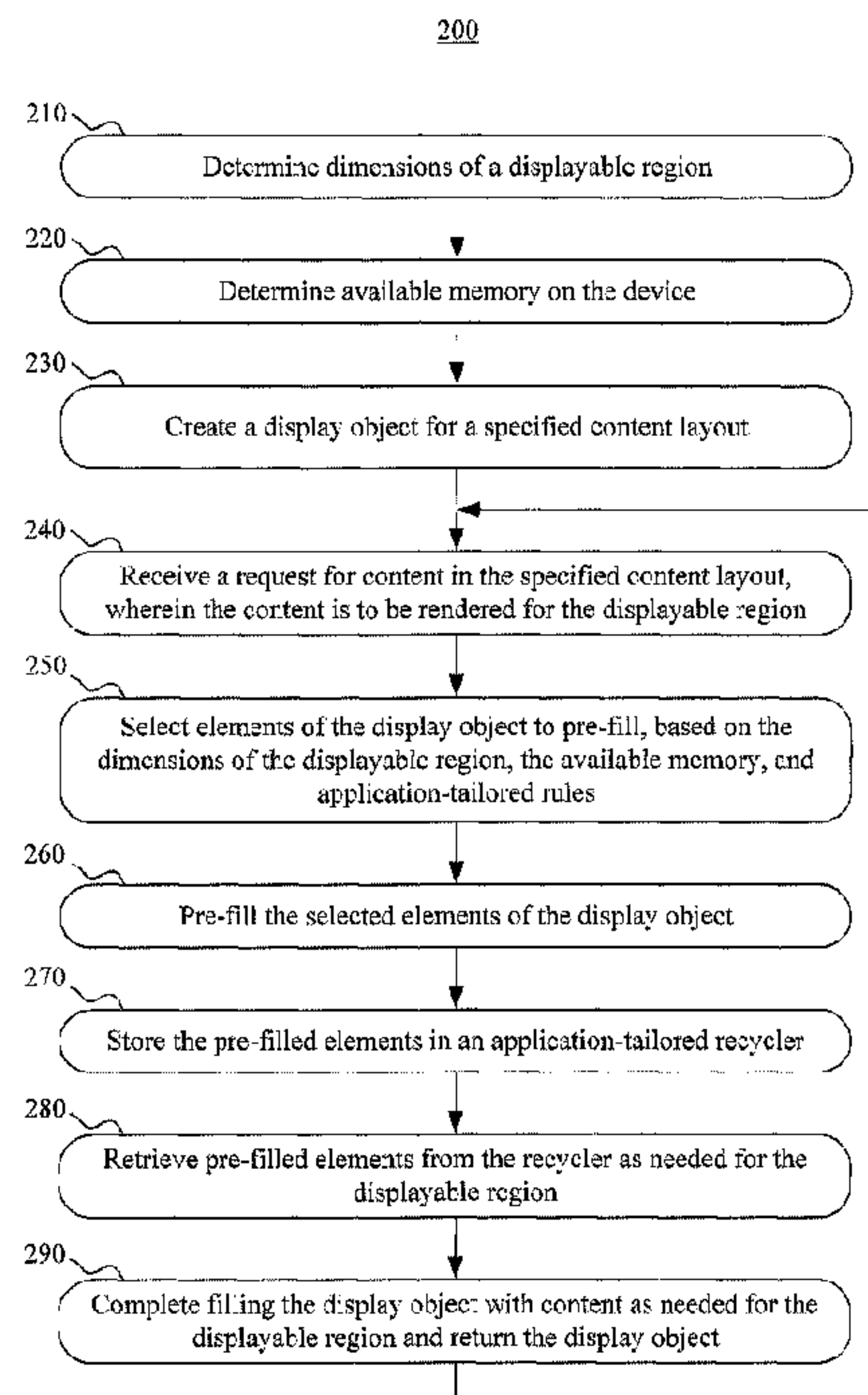




(86) Date de dépôt PCT/PCT Filing Date: 2014/04/03
 (87) Date publication PCT/PCT Publication Date: 2014/10/23
 (45) Date de délivrance/Issue Date: 2019/10/08
 (85) Entrée phase nationale/National Entry: 2015/10/07
 (86) N° demande PCT/PCT Application No.: US 2014/032823
 (87) N° publication PCT/PCT Publication No.: 2014/172114
 (30) Priorité/Priority: 2013/04/15 (US13/862,960)

(51) Cl.Int./Int.Cl. *G06F 3/14* (2006.01)
 (72) Inventeurs/Inventors:
 DU, QIXING, US;
 PENG, I. CHIEN, US;
 LI, JOSHUA, US
 (73) Propriétaire/Owner:
 FACEBOOK, INC., US
 (74) Agent: ROBIC

(54) Titre : PRE-GONFLAGE D'OBJET AJUSTE PAR APPLICATION
 (54) Title: APPLICATION-TAILORED OBJECT PRE-INFLATION



(57) **Abrégé/Abstract:**

In one embodiment, a computing device receives a request for content in a first portion of a content layout in a displayable region of a screen associated with the computing device. The device may pre-inflate at least one selected element of a display object for a second portion of the content layout, and then store the element in an application-tailored recycler. Selection of the at least one selected element may be based on dimensions of the displayable region, available memory of the computing device, or application-specific rules. The device may then retrieve, in response to a request for content in the second portion of the content layout, the element from the application-tailored recycler, update other elements of the display object as needed for the second portion of the content layout, and return the display object.

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

(19) World Intellectual Property
Organization
International Bureau(43) International Publication Date
23 October 2014 (23.10.2014)

WIPO | PCT

(10) International Publication Number
WO 2014/172114 A1

- (51) International Patent Classification:
G06F 3/14 (2006.01)
- (21) International Application Number:
PCT/US2014/032823
- (22) International Filing Date:
3 April 2014 (03.04.2014)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
13/862,960 15 April 2013 (15.04.2013) US
- (71) Applicant: FACEBOOK , INC. [US/US]; 1601 Willow Road, Menlo Park, CA 94025 (US).
- (72) Inventors: DU, Qixing; 1601 Willow Road, Menlo Park, CA 94025 (US). PENG, I, Chien; 1601 Willow Road, Menlo Park, CA 94025 (US). LI, Joshua; 1601 Willow Road, Menlo Park, CA 94025 (US).
- (74) Agent: CHOI, Hogene, L.; Baker Botts L.L.P., 2001 Ross Avenue, Suite 600, Dallas, TX 75201 (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, LT, 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, 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,

[Continued on next page]

(54) Title: APPLICATION-TAILORED OBJECT PRE-INFLATION

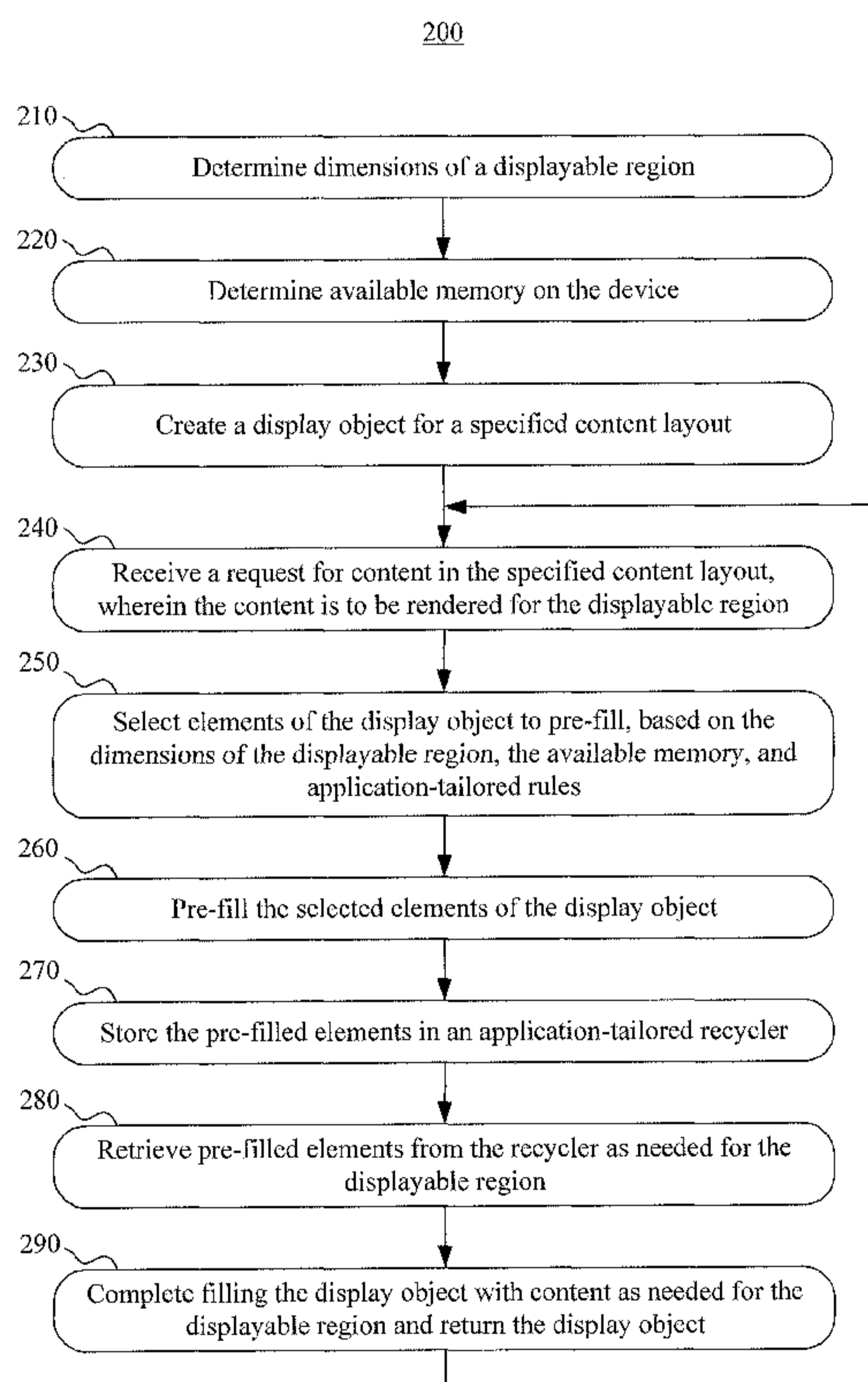


FIG. 2

(57) Abstract: In one embodiment, a computing device receives a request for content in a first portion of a content layout in a displayable region of a screen associated with the computing device. The device may pre-inflate at least one selected element of a display object for a second portion of the content layout, and then store the element in an application-tailored recycler. Selection of the at least one selected element may be based on dimensions of the displayable region, available memory of the computing device, or application-specific rules. The device may then retrieve, in response to a request for content in the second portion of the content layout, the element from the application-tailored recycler, update other elements of the display object as needed for the second portion of the content layout, and return the display object.

WO 2014/172114 A1

WO 2014/172114 A1



TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG). **Published:**

— *with international search report (Art. 21(3))*

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

APPLICATION-TAILORED OBJECT PRE-INFLATION

TECHNICAL FIELD

[1] This disclosure generally relates to object lifecycle management.

BACKGROUND

[2] Creating display objects and inflating them may constitute a significant cause of latency for content-intensive applications, particularly content layouts with variable and/or lengthy sections of content, such as a list or grid. When rendering a content layout for a particular display screen where the content extends beyond the displayable region (requiring the user to scroll the content layout to bring content into view), display objects, nested display objects, and other content elements may be created and inflated as they come into view and recycled as they disappear from view. Request-driven real-time inflation of display objects may prove insufficient for the requirements of such a content-intensive application.

SUMMARY OF PARTICULAR EMBODIMENTS

[3] In order to facilitate pre-inflation of display objects in an application-tailored manner, an application may apply application-specific policies to determine how to selectively pre-inflate only those display objects and/or elements of a display object that are most costly to inflate.

[4] Prior to receiving a request for a particular section of a content layout to be displayed, the application may select elements of the particular section of the content layout to pre-inflate, inflate those selected elements, and store the pre-inflated elements in a recycler. When the same content element is requested again, the application retrieves the display object from the recycler, encloses it in a wrapper object, and returns the wrapped display object. The embodiments described herein may also be performed with elements of a display object, some of which may comprise nested display objects.

BRIEF DESCRIPTION OF THE DRAWINGS

- [5] FIG. 1A is a wireframe of a content layout rendered using display objects.
- [6] FIG. 1B is a wireframe illustrating a detailed section of FIG. 1A.
- [7] FIG. 2 illustrates an example method for application-tailored object pre-inflation.
- [8] FIG. 3 illustrates an example network environment associated with a social-networking system.
- [9] FIG. 4 illustrates an example social graph.
- [10] FIG. 5 illustrates an example computer system.

DESCRIPTION OF EXAMPLE EMBODIMENTS

[11] In order to facilitate pre-inflation of display objects in an application-tailored manner, an application may apply application-specific policies to determine how to selectively pre-inflate only those display objects and/or elements of a display object that are most costly to inflate.

[12] Prior to receiving a request for a particular section of a content layout to be displayed, the application may select elements of the particular section of the content layout to pre-inflate, inflate those selected elements, and store the pre-inflated elements in a recycler. When that content section is requested, the application retrieves the display object from the recycler, encloses it in a wrapper object, and returns the wrapped display object. The embodiments described herein may also be performed with elements of a display object, some of which may comprise nested display objects.

[13] In connection with methods of implementing a recycler, particular embodiments may utilize one or more systems, components, elements, functions, methods, operations, or steps disclosed in U.S. Patent Application No. 13/861,250, filed 11 April 2013 and entitled “Application-Tailored Object Recycling”.

[14] FIG. 1A is a wireframe illustrating an example wireframe of a content layout 100 a newsfeed page for a user of a social-networking system that is rendered for display on a tablet device based on a number of display objects. The displayable region of content layout 100 (indicated by the dashed line) includes the portion of content layout 100 that will fit within the screen dimensions of the tablet device (above the fold).

[15] Content layout 100 includes a header region 110, a menu bar 120, a left-hand sidebar 130, a right-hand sidebar 140, and a main body area 150. Each region is associated with its own display object, each of which may include a number of elements such as, by way of example and not limitation: image(s), video, text, links, forms, and interactive elements. Each of these elements may be represented by one or more nested display objects.

[16] The display object associated with header region 110 includes an interactive profile photo 112 of the social-network user, text block 114, a background image 116, and theme music selected by the social-network user. Interactive profile photo 112 is rendered based on a display object that may include an image, text, a link, and Javascript code to render a pop-up window displaying a summary of a profile of the social-network user that appears when a pointing device, such as a mouse, hovers over the profile photo.

[17] The display object associated with menu bar 120 includes five menu options 125A-F, each of which comprises an image, text, and a link.

[18] The display object associated with left-hand sidebar 130 includes content for a mix of recommendations (including sponsored stories) (only 135A-E are shown, but there may be many more recommendations), each of which is rendered based on a display object. Each display object for a recommendation may include an interactive image, a title, text for the recommendation, and a link to obtain further information related to the recommendation. Each display object for a newsfeed item may also include one or more images or a video

[19] The display object associated with right-hand sidebar 140 includes content for one or more advertisements 145, each of which is rendered based on a display object. As illustrated in the example shown in FIG. 1A, advertisement 145A appears above the fold, whereas 145B is not yet in view. The content of the advertisement may be determined at least in part by a third-party advertiser.

[20] The display object associated with main body area 150 includes a number of newsfeed items (only 155A-E are shown, but there may be many more newsfeed items), each of which is rendered based on a display object associated with an activity related to friend (i.e., social-graph connection) of the user.

[21] FIG. 1B is a wireframe illustrating a detailed section of FIG. 1A: newsfeed items 155B-E in the main body area 150 of content layout 100. As indicated in FIG. 1A, the

displayable region of content layout 100 (indicated by the dashed line) includes the portion of content layout 100 that will fit onto the screen of the tablet device.

[22] Each display object for a newsfeed item 155 may include an interactive profile photo of the friend 160, a name of the friend 165, text for the newsfeed item 170, time-sensitive information 175 (e.g., a constantly-updating count of the number of people who have viewed/commented on/"Liked" the listing, a short listing of the most recent one or two people who viewed/commented on/"Liked" the listing, etc.), a link to "Like" the newsfeed item 180, a link to comment on the newsfeed item 185, and a link to "Share" the newsfeed item 190. Each display object for a newsfeed item may also include one or more images 195 or a video.

[23] Each interactive profile photo 160 is rendered based on a display object that may include an image, text, a link, and Javascript code to render a pop-up window displaying a summary of a profile of the social-network user that appears when a pointing device, such as a mouse, hovers over the profile photo.

[24] FIG. 2 illustrates an example method for application-controlled pre-inflation of display objects. In step 210, the application may determine dimensions of a displayable region--this information may factor into the process of selecting elements of the display object to pre-inflate, as described in step 250 below. The displayable region comprises an area in which a content layout (or a portion thereof) is to be rendered using content associated with a display object. The displayable region may be determined based on any relevant information, such as, by way of example and not limitation: dimensions of a screen, an aspect ratio of the screen, a type of the content to be rendered, application-specific factors (e.g., differing content layouts based on whether the user is a paid subscriber to whom advertisements need not be served), user-configured settings (e.g., re-sizing a window), user preferences (e.g., preventing pop-up windows), etc.

[25] In step 220, the application may determine the available memory on the device--this information may factor into the process of selecting elements of the display object to pre-inflate, as described in step 250 below.

[26] In step 230, the application may determine create a display object for a specified content layout. For example, if the user requests their social-networking information-based newsfeed, the application may create a display object for the newsfeed content layout.

[27] In step 240, the application may receive a request for particular content (e.g., the content included in the displayable region indicated in FIG. 1A, which includes header region 110, menu bar 120, the portion of left-hand sidebar 130 that includes recommendation 135A, the portion of right-hand sidebar 140 that includes advertisement 145A, and the portion of main body area 150 that includes newsfeed item 155A), to which it may respond with a display object containing the requested content. As described above, the display object may contain a number of elements, which may themselves be display objects. For example, newsfeed item 155C includes an interactive profile photo 160 that may be rendered by a display object

[28] In step 250, the application may select elements of the display object to pre-inflate, based on the dimensions of the displayable region, the available memory, and application-specific rules. In particular embodiments, the application-specific rules may prioritize selection of elements of the display object that are more costly to inflate. The cost of inflating an element may be determined on any applicable basis, e.g., size of the element, complexity of the element (e.g., is the element a nested display object?), how frequently the element must be updated, capabilities and configuration of the platform (e.g., does the device lack support for or prohibit execution of client-side scripts, which necessitates substitution with alternate, simpler content?), etc.

[29] In particular embodiments, particularly with respect to list-based or grid-based content layouts the application-specific rules may set an increment by which to pre-inflate elements, wherein the increment is a multiple of units defined by the number of list or grid items that will fit into the dimensions of the displayable region. For example, as shown in FIG. 1B, four newsfeed items may inflate a vertical dimension of the displayable region, where the content layout may extend in the vertical dimension (but not the horizontal dimension). In this example, the application-specific rules determine that the unit comprises four newsfeed items (since that is what will inflate out the displayable region), and set the multiple for the increment to three, so that when the application responds to a request to display newsfeed items 155B-E, it also pre-inflates selected elements for the next twelve newsfeed items.

[30] In particular embodiments, the multiple may be based on user-specific usage patterns. For example, if a user typically scrolls through the newsfeed fairly slowly, then the multiple may be a smaller number. Conversely, if the user typically scrolls through the newsfeed fairly quickly, then the multiple may be a larger number.

[31] In particular embodiments, the multiple may be based on attributes of the content for the list or grid elements. For example, if the content is of a type that is very rich and may likely take the user longer to browse through (e.g., thumbnails of videos that play as the user mouses over each one), then the multiple may be reduced.

[32] In particular embodiments, the application-specific rules may prioritize selection of elements of the display object based on available memory. For example, if there is not much available memory, the application may choose to pre-inflate only the most costly elements, whereas if there was more available memory, it might pre-inflate additional (somewhat less costly elements). In another example, if there is not much available memory, the application may choose to pre-inflate those elements that are likely to be re-used very often, e.g., if the next ten newsfeed items are all related to people who have posted the same complex link (which includes an image, text, a link, and other information) to a popular online article, the element representing the complex link may receive priority above other elements.

[33] In step 260, the application may pre-inflate the selected elements of the display object. As part of pre-inflating a selected element (which may itself be a display object), particular embodiments may perform steps 250-290 within the context of the selected element.

[34] In step 270, the application may store the pre-inflated elements of the display object in an application-tailored recycler. These stored elements may be retained in the recycler until the relevant portions of the content layout are requested for the displayable region.

[35] In step 280, the application may retrieve any pre-inflated elements of the display object that can be used for the content to be rendered for the displayable region in the application-tailored recycler. However, since the recycler may contain previously-used display objects and elements thereof as well as pre-inflated display objects and elements thereof, particular embodiments may apply application-tailored recycling policies to identify which display objects and elements thereof in the recycler are suitable for re-use, which may include those that were pre-inflated.

[36] In particular embodiments, a display object in the recycler may be suitable for re-use if it was inflated using the same class hierarchy as the display object that is currently being requested. In particular embodiments, a display object in the recycler may be suitable for re-use if it was inflated using the same XML specification as the display object that is currently being requested.

[37] A display object in the recycler may be suitable for recycling if it is the same type of object but is inflated with different content. In this case, the application may inflate the display object from scratch (which should not be necessary for any pre-inflated display objects or elements thereof), thereby overwriting most if not all of the existing content in the display object.

[38] The determination of which display objects and elements are suitable for re-use, recycling, or neither may be dependent upon application-specific, device-specific, and/or user-specific pre-inflation policies. In particular embodiments, an application may set pre-inflation policies in accordance with application requirements, content types, device attributes (e.g., available memory and/or dimensions of a display screen.), content statistics, usage patterns and preferences, or any other appropriate factor(s). Such policies may govern how many objects to retain in the recycler, what types of objects to store in the recycler, how to determine whether a display object is suitable for re-use and/or re-cycling, when to re-use versus recycle a display object, etc.

[39] If the recycler returns a null, the application creates a new display object of an appropriate type to display the requested content, and then inflates the display object. This is the most expensive path, since both creating the new display object and inflating the display object are costly operations. Therefore, an application may need to balance conflicting priorities when determining when to prohibit recycling of otherwise suitable display objects stored in the recycler.

[40] In step 290, the application may return the display object in response to the request.

[41] Embodiments disclosed herein are described as being performed with respect to elements of a display object—however, in particular embodiments, the techniques described herein may be applied in a recursive fashion to certain elements of a display object (which may themselves be display objects) in order to pre-inflate a portion or all of those elements.

[42] Embodiments disclosed herein are described as being performed by an application; however, as one of skill in the art would be aware, many of the steps in example method 200 and other embodiments disclosed herein may be performed by one or more pieces of software operating at one or more layers of abstraction. For example, the steps of the example method described above may be performed by software that executes separately from the application and interfaces with the application and the recycler.

[43] Particular embodiments may repeat one or more steps of the method of FIG. 2, where appropriate. Although this disclosure describes and illustrates particular steps of the method of FIG. 2 as occurring in a particular order, this disclosure contemplates any suitable steps of the method of FIG. 2 occurring in any suitable order. Moreover, although this disclosure describes and illustrates particular components, devices, or systems carrying out particular steps of the method of FIG. 2, this disclosure contemplates any suitable combination of any suitable components, devices, or systems carrying out any suitable steps of the method of FIG. 2.

[44] FIG. 3 illustrates an example network environment 300 associated with a social-networking system. Network environment 300 includes a user 301, a client system 330, a social-networking system 360, and a third-party system 370 connected to each other by a network 310. Although FIG. 3 illustrates a particular arrangement of user 301, client system 330, social-networking system 360, third-party system 370, and network 310, this disclosure contemplates any suitable arrangement of user 301, client system 330, social-networking system 360, third-party system 370, and network 310. As an example and not by way of limitation, two or more of client system 330, social-networking system 360, and third-party system 370 may be connected to each other directly, bypassing network 310. As another example, two or more of client system 330, social-networking system 360, and third-party system 370 may be physically or logically co-located with each other in whole or in part. Moreover, although FIG. 3 illustrates a particular number of users 301, client systems 330, social-networking systems 360, third-party systems 370, and networks 310, this disclosure contemplates any suitable number of users 301, client systems 330, social-networking systems 360, third-party systems 370, and networks 310. As an example and not by way of limitation, network environment 300 may include multiple users 301, client system 330, social-networking systems 360, third-party systems 370, and networks 310.

[45] In particular embodiments, user 301 may be an individual (human user), an entity (e.g., an enterprise, business, or third-party application), or a group (e.g., of individuals or entities) that interacts or communicates with or over social-networking system 360. In particular embodiments, social-networking system 360 may be a network-addressable computing system hosting an online social network. Social-networking system 360 may generate, store, receive, and send social-networking data, such as, for example, user-profile data, concept-profile data, social-graph information, or other suitable data related to the online social network. Social-networking system 360 may be accessed by the other components of network environment 300

either directly or via network 310. In particular embodiments, social-networking system 360 may include an authorization server (or other suitable component(s)) that allows users 301 to opt in to or opt out of having their actions logged by social-networking system 360 or shared with other systems (e.g., third-party systems 370), for example, by setting appropriate privacy settings. A privacy setting of a user may determine what information associated with the user may be logged, how information associated with the user may be logged, when information associated with the user may be logged, who may log information associated with the user, whom information associated with the user may be shared with, and for what purposes information associated with the user may be logged or shared. Authorization servers may be used to enforce one or more privacy settings of the users of social-networking system 30 through blocking, data hashing, anonymization, or other suitable techniques as appropriate. Third-party system 370 may be accessed by the other components of network environment 300 either directly or via network 310. In particular embodiments, one or more users 301 may use one or more client systems 330 to access, send data to, and receive data from social-networking system 360 or third-party system 370. Client system 330 may access social-networking system 360 or third-party system 370 directly, via network 310, or via a third-party system. As an example and not by way of limitation, client system 330 may access third-party system 370 via social-networking system 360. Client system 330 may be any suitable computing device, such as, for example, a personal computer, a laptop computer, a cellular telephone, a smartphone, or a tablet computer.

[46] This disclosure contemplates any suitable network 310. As an example and not by way of limitation, one or more portions of network 310 may include an ad hoc network, an intranet, an extranet, a virtual private network (VPN), a local area network (LAN), a wireless LAN (WLAN), a wide area network (WAN), a wireless WAN (WWAN), a metropolitan area network (MAN), a portion of the Internet, a portion of the Public Switched Telephone Network (PSTN), a cellular telephone network, or a combination of two or more of these. Network 310 may include one or more networks 310.

[47] Links 350 may connect client system 330, social-networking system 360, and third-party system 370 to communication network 310 or to each other. This disclosure contemplates any suitable links 350. In particular embodiments, one or more links 350 include one or more wireline (such as for example Digital Subscriber Line (DSL) or Data Over Cable Service Interface Specification (DOCSIS)), wireless (such as for example Wi-Fi or Worldwide

Interoperability for Microwave Access (WiMAX)), or optical (such as for example Synchronous Optical Network (SONET) or Synchronous Digital Hierarchy (SDH)) links. In particular embodiments, one or more links 350 each include an ad hoc network, an intranet, an extranet, a VPN, a LAN, a WLAN, a WAN, a WWAN, a MAN, a portion of the Internet, a portion of the PSTN, a cellular technology-based network, a satellite communications technology-based network, another link 350, or a combination of two or more such links 350. Links 350 need not necessarily be the same throughout network environment 300. One or more first links 350 may differ in one or more respects from one or more second links 350.

[48] FIG. 4 illustrates example social graph 400. In particular embodiments, social-networking system 360 may store one or more social graphs 400 in one or more data stores. In particular embodiments, social graph 400 may include multiple nodes—which may include multiple user nodes 402 or multiple concept nodes 404—and multiple edges 406 connecting the nodes. Example social graph 400 illustrated in FIG. 4 is shown, for didactic purposes, in a two-dimensional visual map representation. In particular embodiments, a social-networking system 360, client system 330, or third-party system 370 may access social graph 400 and related social-graph information for suitable applications. The nodes and edges of social graph 400 may be stored as data objects, for example, in a data store (such as a social-graph database). Such a data store may include one or more searchable or queryable indexes of nodes or edges of social graph 400.

[49] In particular embodiments, a user node 402 may correspond to a user of social-networking system 360. As an example and not by way of limitation, a user may be an individual (human user), an entity (e.g., an enterprise, business, or third-party application), or a group (e.g., of individuals or entities) that interacts or communicates with or over social-networking system 360. In particular embodiments, when a user registers for an account with social-networking system 360, social-networking system 360 may create a user node 402 corresponding to the user, and store the user node 402 in one or more data stores. Users and user nodes 402 described herein may, where appropriate, refer to registered users and user nodes 402 associated with registered users. In addition or as an alternative, users and user nodes 402 described herein may, where appropriate, refer to users that have not registered with social-networking system 360. In particular embodiments, a user node 402 may be associated with information provided by a user or information gathered by various systems, including social-networking system 360. As an

example and not by way of limitation, a user may provide his or her name, profile picture, contact information, birth date, sex, marital status, family status, employment, education background, preferences, interests, or other demographic information. In particular embodiments, a user node 402 may be associated with one or more data objects corresponding to information associated with a user. In particular embodiments, a user node 402 may correspond to one or more webpages.

[50] In particular embodiments, a concept node 404 may correspond to a concept. As an example and not by way of limitation, a concept may correspond to a place (such as, for example, a movie theater, restaurant, landmark, or city); a website (such as, for example, a website associated with social-network system 360 or a third-party website associated with a web-application server); an entity (such as, for example, a person, business, group, sports team, or celebrity); a resource (such as, for example, an audio file, video file, digital photo, text file, structured document, or application) which may be located within social-networking system 360 or on an external server, such as a web-application server; real or intellectual property (such as, for example, a sculpture, painting, movie, game, song, idea, photograph, or written work); a game; an activity; an idea or theory; another suitable concept; or two or more such concepts. A concept node 404 may be associated with information of a concept provided by a user or information gathered by various systems, including social-networking system 360. As an example and not by way of limitation, information of a concept may include a name or a title; one or more images (e.g., an image of the cover page of a book); a location (e.g., an address or a geographical location); a website (which may be associated with a URL); contact information (e.g., a phone number or an email address); other suitable concept information; or any suitable combination of such information. In particular embodiments, a concept node 404 may be associated with one or more data objects corresponding to information associated with concept node 404. In particular embodiments, a concept node 404 may correspond to one or more webpages.

[51] In particular embodiments, a node in social graph 400 may represent or be represented by a webpage (which may be referred to as a “profile page”). Profile pages may be hosted by or accessible to social-networking system 360. Profile pages may also be hosted on third-party websites associated with a third-party server 370. As an example and not by way of limitation, a profile page corresponding to a particular external webpage may be the particular

external webpage and the profile page may correspond to a particular concept node 404. Profile pages may be viewable by all or a selected subset of other users. As an example and not by way of limitation, a user node 402 may have a corresponding user-profile page in which the corresponding user may add content, make declarations, or otherwise express himself or herself. As another example and not by way of limitation, a concept node 404 may have a corresponding concept-profile page in which one or more users may add content, make declarations, or express themselves, particularly in relation to the concept corresponding to concept node 404.

[52] In particular embodiments, a concept node 404 may represent a third-party webpage or resource hosted by a third-party system 370. The third-party webpage or resource may include, among other elements, content, a selectable or other icon, or other inter-actable object (which may be implemented, for example, in JavaScript, AJAX, or PHP codes) representing an action or activity. As an example and not by way of limitation, a third-party webpage may include a selectable icon such as “like,” “check in,” “eat,” “recommend,” or another suitable action or activity. A user viewing the third-party webpage may perform an action by selecting one of the icons (e.g., “eat”), causing a client system 330 to send to social-networking system 360 a message indicating the user’s action. In response to the message, social-networking system 360 may create an edge (e.g., an “eat” edge) between a user node 402 corresponding to the user and a concept node 404 corresponding to the third-party webpage or resource and store edge 406 in one or more data stores.

[53] In particular embodiments, a pair of nodes in social graph 400 may be connected to each other by one or more edges 406. An edge 406 connecting a pair of nodes may represent a relationship between the pair of nodes. In particular embodiments, an edge 406 may include or represent one or more data objects or attributes corresponding to the relationship between a pair of nodes. As an example and not by way of limitation, a first user may indicate that a second user is a “friend” of the first user. In response to this indication, social-networking system 360 may send a “friend request” to the second user. If the second user confirms the “friend request,” social-networking system 360 may create an edge 406 connecting the first user’s user node 402 to the second user’s user node 402 in social graph 400 and store edge 406 as social-graph information in one or more of data stores 364. In the example of FIG. 4, social graph 400 includes an edge 406 indicating a friend relation between user nodes 402 of user “A” and user “B” and an edge indicating a friend relation between user nodes 402 of user “C” and user “B.”

Although this disclosure describes or illustrates particular edges 406 with particular attributes connecting particular user nodes 402, this disclosure contemplates any suitable edges 406 with any suitable attributes connecting user nodes 402. As an example and not by way of limitation, an edge 406 may represent a friendship, family relationship, business or employment relationship, fan relationship, follower relationship, visitor relationship, subscriber relationship, superior/subordinate relationship, reciprocal relationship, non-reciprocal relationship, another suitable type of relationship, or two or more such relationships. Moreover, although this disclosure generally describes nodes as being connected, this disclosure also describes users or concepts as being connected. Herein, references to users or concepts being connected may, where appropriate, refer to the nodes corresponding to those users or concepts being connected in social graph 400 by one or more edges 406.

[54] In particular embodiments, an edge 406 between a user node 402 and a concept node 404 may represent a particular action or activity performed by a user associated with user node 402 toward a concept associated with a concept node 404. As an example and not by way of limitation, as illustrated in FIG. 4, a user may “like,” “attended,” “played,” “listened,” “cooked,” “worked at,” or “watched” a concept, each of which may correspond to a edge type or subtype. A concept-profile page corresponding to a concept node 404 may include, for example, a selectable “check in” icon (such as, for example, a clickable “check in” icon) or a selectable “add to favorites” icon. Similarly, after a user clicks these icons, social-networking system 360 may create a “favorite” edge or a “check in” edge in response to a user’s action corresponding to a respective action. As another example and not by way of limitation, a user (user “C”) may listen to a particular song (“Ramble On”) using a particular application (SPOTIFY, which is an online music application). In this case, social-networking system 360 may create a “listened” edge 406 and a “used” edge (as illustrated in FIG. 4) between user nodes 402 corresponding to the user and concept nodes 404 corresponding to the song and application to indicate that the user listened to the song and used the application. Moreover, social-networking system 360 may create a “played” edge 406 (as illustrated in FIG. 4) between concept nodes 404 corresponding to the song and the application to indicate that the particular song was played by the particular application. In this case, “played” edge 406 corresponds to an action performed by an external application (SPOTIFY) on an external audio file (the song “Imagine”). Although this disclosure describes particular edges 406 with particular attributes connecting user nodes 402 and concept

nodes 404, this disclosure contemplates any suitable edges 406 with any suitable attributes connecting user nodes 402 and concept nodes 404. Moreover, although this disclosure describes edges between a user node 402 and a concept node 404 representing a single relationship, this disclosure contemplates edges between a user node 402 and a concept node 404 representing one or more relationships. As an example and not by way of limitation, an edge 406 may represent both that a user likes and has used at a particular concept. Alternatively, another edge 406 may represent each type of relationship (or multiples of a single relationship) between a user node 402 and a concept node 404 (as illustrated in FIG. 4 between user node 402 for user “E” and concept node 404 for “SPOTIFY”).

[55] In particular embodiments, social-networking system 360 may create an edge 406 between a user node 402 and a concept node 404 in social graph 400. As an example and not by way of limitation, a user viewing a concept-profile page (such as, for example, by using a web browser or a special-purpose application hosted by the user’s client system 330) may indicate that he or she likes the concept represented by the concept node 404 by clicking or selecting a “Like” icon, which may cause the user’s client system 330 to send to social-networking system 360 a message indicating the user’s liking of the concept associated with the concept-profile page. In response to the message, social-networking system 360 may create an edge 406 between user node 402 associated with the user and concept node 404, as illustrated by “like” edge 406 between the user and concept node 404. In particular embodiments, social-networking system 360 may store an edge 406 in one or more data stores. In particular embodiments, an edge 406 may be automatically formed by social-networking system 360 in response to a particular user action. As an example and not by way of limitation, if a first user uploads a picture, watches a movie, or listens to a song, an edge 406 may be formed between user node 402 corresponding to the first user and concept nodes 404 corresponding to those concepts. Although this disclosure describes forming particular edges 406 in particular manners, this disclosure contemplates forming any suitable edges 406 in any suitable manner.

[56] In particular embodiments, an advertisement may be text (which may be HTML-linked), one or more images (which may be HTML-linked), one or more videos, audio, one or more ADOBE FLASH files, a suitable combination of these, or any other suitable advertisement in any suitable digital format presented on one or more webpages, in one or more e-mails, or in connection with search results requested by a user. In addition or as an alternative, an

advertisement may be one or more sponsored stories (e.g., a news-feed or ticker item on social-networking system 360). A sponsored story may be a social action by a user (such as “liking” a page, “liking” or commenting on a post on a page, RSVPing to an event associated with a page, voting on a question posted on a page, checking in to a place, using an application or playing a game, or “liking” or sharing a website) that an advertiser promotes, for example, by having the social action presented within a pre-determined area of a profile page of a user or other page, presented with additional information associated with the advertiser, bumped up or otherwise highlighted within news feeds or tickers of other users, or otherwise promoted. The advertiser may pay to have the social action promoted. As an example and not by way of limitation, advertisements may be included among the search results of a search-results page, where sponsored content is promoted over non-sponsored content.

[57] In particular embodiments, an advertisement may be requested for display within social-networking-system webpages, third-party webpages, or other pages. An advertisement may be displayed in a dedicated portion of a page, such as in a banner area at the top of the page, in a column at the side of the page, in a GUI of the page, in a pop-up window, in a drop-down menu, in an input field of the page, over the top of content of the page, or elsewhere with respect to the page. In addition or as an alternative, an advertisement may be displayed within an application. An advertisement may be displayed within dedicated pages, requiring the user to interact with or watch the advertisement before the user may access a page or utilize an application. The user may, for example view the advertisement through a web browser.

[58] FIG. 5 illustrates an example computer system 500. In particular embodiments, one or more computer systems 500 perform one or more steps of one or more methods described or illustrated herein. In particular embodiments, one or more computer systems 500 provide functionality described or illustrated herein. In particular embodiments, software running on one or more computer systems 500 performs one or more steps of one or more methods described or illustrated herein or provides functionality described or illustrated herein. Particular embodiments include one or more portions of one or more computer systems 500. Herein, reference to a computer system may encompass a computing device, and vice versa, where appropriate. Moreover, reference to a computer system may encompass one or more computer systems, where appropriate.

[59] This disclosure contemplates any suitable number of computer systems 500. This disclosure contemplates computer system 500 taking any suitable physical form. As example and not by way of limitation, computer system 500 may be an embedded computer system, a system-on-chip (SOC), a single-board computer system (SBC) (such as, for example, a computer-on-module (COM) or system-on-module (SOM)), a desktop computer system, a laptop or notebook computer system, an interactive kiosk, a mainframe, a mesh of computer systems, a mobile telephone, a personal digital assistant (PDA), a server, a tablet computer system, or a combination of two or more of these. Where appropriate, computer system 500 may include one or more computer systems 500; be unitary or distributed; span multiple locations; span multiple machines; span multiple data centers; or reside in a cloud, which may include one or more cloud components in one or more networks. Where appropriate, one or more computer systems 500 may perform without substantial spatial or temporal limitation one or more steps of one or more methods described or illustrated herein. As an example and not by way of limitation, one or more computer systems 500 may perform in real time or in batch mode one or more steps of one or more methods described or illustrated herein. One or more computer systems 500 may perform at different times or at different locations one or more steps of one or more methods described or illustrated herein, where appropriate.

[60] In particular embodiments, computer system 500 includes a processor 502, memory 504, storage 506, an input/output (I/O) interface 508, a communication interface 510, and a bus 512. Although this disclosure describes and illustrates a particular computer system having a particular number of particular components in a particular arrangement, this disclosure contemplates any suitable computer system having any suitable number of any suitable components in any suitable arrangement.

[61] In particular embodiments, processor 502 includes hardware for executing instructions, such as those making up a computer program. As an example and not by way of limitation, to execute instructions, processor 502 may retrieve (or fetch) the instructions from an internal register, an internal cache, memory 504, or storage 506; decode and execute them; and then write one or more results to an internal register, an internal cache, memory 504, or storage 506. In particular embodiments, processor 502 may include one or more internal caches for data, instructions, or addresses. This disclosure contemplates processor 502 including any suitable number of any suitable internal caches, where appropriate. As an example and not by way of

limitation, processor 502 may include one or more instruction caches, one or more data caches, and one or more translation lookaside buffers (TLBs). Instructions in the instruction caches may be copies of instructions in memory 504 or storage 506, and the instruction caches may speed up retrieval of those instructions by processor 502. Data in the data caches may be copies of data in memory 504 or storage 506 for instructions executing at processor 502 to operate on; the results of previous instructions executed at processor 502 for access by subsequent instructions executing at processor 502 or for writing to memory 504 or storage 506; or other suitable data. The data caches may speed up read or write operations by processor 502. The TLBs may speed up virtual-address translation for processor 502. In particular embodiments, processor 502 may include one or more internal registers for data, instructions, or addresses. This disclosure contemplates processor 502 including any suitable number of any suitable internal registers, where appropriate. Where appropriate, processor 502 may include one or more arithmetic logic units (ALUs); be a multi-core processor; or include one or more processors 502. Although this disclosure describes and illustrates a particular processor, this disclosure contemplates any suitable processor.

[62] In particular embodiments, memory 504 includes main memory for storing instructions for processor 502 to execute or data for processor 502 to operate on. As an example and not by way of limitation, computer system 500 may load instructions from storage 506 or another source (such as, for example, another computer system 500) to memory 504. Processor 502 may then load the instructions from memory 504 to an internal register or internal cache. To execute the instructions, processor 502 may retrieve the instructions from the internal register or internal cache and decode them. During or after execution of the instructions, processor 502 may write one or more results (which may be intermediate or final results) to the internal register or internal cache. Processor 502 may then write one or more of those results to memory 504. In particular embodiments, processor 502 executes only instructions in one or more internal registers or internal caches or in memory 504 (as opposed to storage 506 or elsewhere) and operates only on data in one or more internal registers or internal caches or in memory 504 (as opposed to storage 506 or elsewhere). One or more memory buses (which may each include an address bus and a data bus) may couple processor 502 to memory 504. Bus 512 may include one or more memory buses, as described below. In particular embodiments, one or more memory management units (MMUs) reside between processor 502 and memory 504 and facilitate

accesses to memory 504 requested by processor 502. In particular embodiments, memory 504 includes random access memory (RAM). This RAM may be volatile memory, where appropriate. Where appropriate, this RAM may be dynamic RAM (DRAM) or static RAM (SRAM). Moreover, where appropriate, this RAM may be single-ported or multi-ported RAM. This disclosure contemplates any suitable RAM. Memory 504 may include one or more memories 504, where appropriate. Although this disclosure describes and illustrates particular memory, this disclosure contemplates any suitable memory.

[63] In particular embodiments, storage 506 includes mass storage for data or instructions. As an example and not by way of limitation, storage 506 may include a hard disk drive (HDD), a floppy disk drive, flash memory, an optical disc, a magneto-optical disc, magnetic tape, or a Universal Serial Bus (USB) drive or a combination of two or more of these. Storage 506 may include removable or non-removable (or fixed) media, where appropriate. Storage 506 may be internal or external to computer system 500, where appropriate. In particular embodiments, storage 506 is non-volatile, solid-state memory. In particular embodiments, storage 506 includes read-only memory (ROM). Where appropriate, this ROM may be mask-programmed ROM, programmable ROM (PROM), erasable PROM (EPROM), electrically erasable PROM (EEPROM), electrically alterable ROM (EAROM), or flash memory or a combination of two or more of these. This disclosure contemplates mass storage 506 taking any suitable physical form. Storage 506 may include one or more storage control units facilitating communication between processor 502 and storage 506, where appropriate. Where appropriate, storage 506 may include one or more storages 506. Although this disclosure describes and illustrates particular storage, this disclosure contemplates any suitable storage.

[64] In particular embodiments, I/O interface 508 includes hardware, software, or both, providing one or more interfaces for communication between computer system 500 and one or more I/O devices. Computer system 500 may include one or more of these I/O devices, where appropriate. One or more of these I/O devices may enable communication between a person and computer system 500. As an example and not by way of limitation, an I/O device may include a keyboard, keypad, microphone, monitor, mouse, printer, scanner, speaker, still camera, stylus, tablet, touch screen, trackball, video camera, another suitable I/O device or a combination of two or more of these. An I/O device may include one or more sensors. This disclosure contemplates any suitable I/O devices and any suitable I/O interfaces 508 for them. Where appropriate, I/O

interface 508 may include one or more device or software drivers enabling processor 502 to drive one or more of these I/O devices. I/O interface 508 may include one or more I/O interfaces 508, where appropriate. Although this disclosure describes and illustrates a particular I/O interface, this disclosure contemplates any suitable I/O interface.

[65] In particular embodiments, communication interface 510 includes hardware, software, or both providing one or more interfaces for communication (such as, for example, packet-based communication) between computer system 500 and one or more other computer systems 500 or one or more networks. As an example and not by way of limitation, communication interface 510 may include a network interface controller (NIC) or network adapter for communicating with an Ethernet or other wire-based network or a wireless NIC (WNIC) or wireless adapter for communicating with a wireless network, such as a WI-FI network. This disclosure contemplates any suitable network and any suitable communication interface 510 for it. As an example and not by way of limitation, computer system 500 may communicate with an ad hoc network, a personal area network (PAN), a local area network (LAN), a wide area network (WAN), a metropolitan area network (MAN), or one or more portions of the Internet or a combination of two or more of these. One or more portions of one or more of these networks may be wired or wireless. As an example, computer system 500 may communicate with a wireless PAN (WPAN) (such as, for example, a BLUETOOTH WPAN), a WI-FI network, a WI-MAX network, a cellular telephone network (such as, for example, a Global System for Mobile Communications (GSM) network), or other suitable wireless network or a combination of two or more of these. Computer system 500 may include any suitable communication interface 510 for any of these networks, where appropriate. Communication interface 510 may include one or more communication interfaces 510, where appropriate. Although this disclosure describes and illustrates a particular communication interface, this disclosure contemplates any suitable communication interface.

[66] In particular embodiments, bus 512 includes hardware, software, or both coupling components of computer system 500 to each other. As an example and not by way of limitation, bus 512 may include an Accelerated Graphics Port (AGP) or other graphics bus, an Enhanced Industry Standard Architecture (EISA) bus, a front-side bus (FSB), a HYPERTRANSPORT (HT) interconnect, an Industry Standard Architecture (ISA) bus, an INFINIBAND interconnect, a low-pin-count (LPC) bus, a memory bus, a Micro Channel Architecture (MCA) bus, a

Peripheral Component Interconnect (PCI) bus, a PCI-Express (PCIe) bus, a serial advanced technology attachment (SATA) bus, a Video Electronics Standards Association local (VLB) bus, or another suitable bus or a combination of two or more of these. Bus 512 may include one or more buses 512, where appropriate. Although this disclosure describes and illustrates a particular bus, this disclosure contemplates any suitable bus or interconnect.

[67] Herein, a computer-readable non-transitory storage medium or media may include one or more semiconductor-based or other integrated circuits (ICs) (such, as for example, field-programmable gate arrays (FPGAs) or application-specific ICs (ASICs)), hard disk drives (HDDs), hybrid hard drives (HHDs), optical discs, optical disc drives (ODDs), magneto-optical discs, magneto-optical drives, floppy diskettes, floppy disk drives (FDDs), magnetic tapes, solid-state drives (SSDs), RAM-drives, SECURE DIGITAL cards or drives, any other suitable computer-readable non-transitory storage media, or any suitable combination of two or more of these, where appropriate. A computer-readable non-transitory storage medium may be volatile, non-volatile, or a combination of volatile and non-volatile, where appropriate.

[68] Herein, “or” is inclusive and not exclusive, unless expressly indicated otherwise or indicated otherwise by context. Therefore, herein, “A or B” means “A, B, or both,” unless expressly indicated otherwise or indicated otherwise by context. Moreover, “and” is both joint and several, unless expressly indicated otherwise or indicated otherwise by context. Therefore, herein, “A and B” means “A and B, jointly or severally,” unless expressly indicated otherwise or indicated otherwise by context.

[69] The scope of this disclosure encompasses all changes, substitutions, variations, alterations, and modifications to the example embodiments described or illustrated herein that a person having ordinary skill in the art would comprehend. The scope of this disclosure is not limited to the example embodiments described or illustrated herein. Moreover, although this disclosure describes and illustrates respective embodiments herein as including particular components, elements, functions, operations, or steps, any of these embodiments may include any combination or permutation of any of the components, elements, functions, operations, or steps described or illustrated anywhere herein that a person having ordinary skill in the art would comprehend.

WHAT IS CLAIMED IS:

1. A method for pre-inflating display objects, comprising:
 - by a computing device, receiving a request for content in a first portion of a content layout in a displayable region of a screen associated with the computing device, wherein the content layout is a newsfeed in a social-networking system;
 - by the computing device, selecting at least one element of one of a plurality of display objects to pre-inflate based on application-specific rules that prioritize element selection for pre-inflation, wherein the display object is associated with a second portion of the content layout, wherein the at least one element is selected based on (1) a frequency with which the at least one element is included the plurality of display objects and (2) a complexity of the at least one element;
 - by the computing device, pre-inflating the selected at least one element of the display object;
 - by the computing device, storing the at least one pre-inflated element in an application-tailored recycler;
 - by the computing device, retrieving, in response to a request for content in the second portion of the content layout, the at least one pre-inflated element from the application-tailored recycler;
 - by the computing device, updating other elements of the display object as needed for the second portion of the content layout; and
 - by the computing device, returning the display object.
2. The method of Claim 1, wherein the displayable region is determined based on dimensions of the screen, an aspect ratio of the screen, a type of the content to be rendered, application-specific factors, user-configured settings, or user preferences.
3. The method of Claim 1, wherein the content layout comprises a list or grid, and wherein application-specific rules set an increment by which to pre-inflate elements, the

increment being a multiple of a unit defined by a number of list or grid items that fit into the displayable region.

4. The method of Claim 3, wherein the multiple is also based on user-specific usage patterns.

5. The method of Claim 3, wherein the multiple is also based on content attributes.

6. The method of Claim 1, wherein the application-specific rules that prioritize element selection for pre-inflation are further based on a cost of pre-inflating each element, wherein the cost of pre-inflating an element is based on factors comprising: a size of the element, capabilities of a platform of the computing device, or a configuration of the computing device.

7. The method of Claim 1, wherein application-specific rules adjust the number of elements to be selected for pre-inflation based on the available memory of the computing device.

8. One or more computer-readable non-transitory storage media embodying software for pre-inflating display objects, that is operable when executed to:

receive a request for content in a first portion of a content layout in a displayable region of a screen associated with a computing device, wherein the content layout is a newsfeed in a social-networking system;

select at least one element of one of a plurality of display objects to pre-inflate based on application-specific rules that prioritize element selection for pre-inflation, wherein the display object is associated with a second portion of the content layout, wherein the at least one element is selected based on (1) a frequency with which the at least one element is included the plurality of display objects and (2) a complexity of the at least one element;

pre-inflate the selected at least one element of the display object;

store the at least one pre-inflated element in an application-tailored recycler;

retrieve, in response to a request for content in the second portion of the content layout, the at least one pre-inflated element from the application-tailored recycler;

update other elements of the display object as needed for the second portion of the content layout; and

return the display object.

9. The media of Claim 8, wherein the displayable region is determined based on dimensions of the screen, an aspect ratio of the screen, a type of the content to be rendered, application-specific factors, user-configured settings, or user preferences.

10. The media of Claim 8, wherein the content layout comprises a list or grid, and wherein application-specific rules set an increment by which to pre-inflate elements, the increment being a multiple of a unit defined by a number of list or grid items that fit into the displayable region.

11. The media of Claim 10, wherein the multiple is also based on user-specific usage patterns.

12. The media of Claim 10, wherein the multiple is also based on content attributes.

13. The media of Claim 8, wherein the application-specific rules that prioritize element selection for pre-inflation are further based on a cost of pre-inflating each element, wherein the cost of pre-inflating an element is based on factors comprising: a size of the element, capabilities of a platform of the computing device, or a configuration of the computing device.

14. The media of Claim 8, wherein application-specific rules adjust the number of elements to be selected for pre-inflation based on the available memory of the computing device.

15. A computing device for pre-inflating display objects, comprising:
one or more processors; and
a memory coupled to the processors comprising instructions executable by the processors, the processors being operable when executing the instructions to:

receive a request for content in a first portion of a content layout in a displayable region of a screen associated with the computing device, wherein the content layout is a newsfeed in a social-networking system;

select at least one element of one of a plurality of display objects to pre-inflate based on application-specific rules that prioritize element selection for pre-inflation, wherein the display object is associated with a second portion of the content layout, wherein the at least one element is selected based on (1) a frequency with which the at least one element is included the plurality of display objects and (2) a complexity of the at least one element;

pre-inflate the selected at least one element of the display object;

store the at least one pre-inflated element in an application-tailored recycler;

retrieve, in response to a request for content in the second portion of the content layout, the at least one pre-inflated element from the application-tailored recycler;

update other elements of the display object as needed for the second portion of the content layout; and

return the display object.

16. The device of Claim 15, wherein the displayable region is determined based on dimensions of the screen, an aspect ratio of the screen, a type of the content to be rendered, application-specific factors, user-configured settings, or user preferences.

17. The device of Claim 15, wherein the content layout comprises a list or grid, and wherein application-specific rules set an increment by which to pre-inflate elements, the increment being a multiple of a unit defined by a number of list or grid items that fit into the displayable region.

18. The device of Claim 17, wherein the multiple is based on user-specific usage patterns or content attributes.

19. The device of Claim 15, wherein the application-specific rules that prioritize element selection for pre-inflation are further based on a cost of pre-inflating each element,

wherein the cost of pre-inflating an element is based on factors comprising: a size of the element, capabilities of a platform of the computing device, or a configuration of the computing device.

20. The device of Claim 15, wherein application-specific rules adjust the number of elements to be selected for pre-inflation based on the available memory of the computing device.

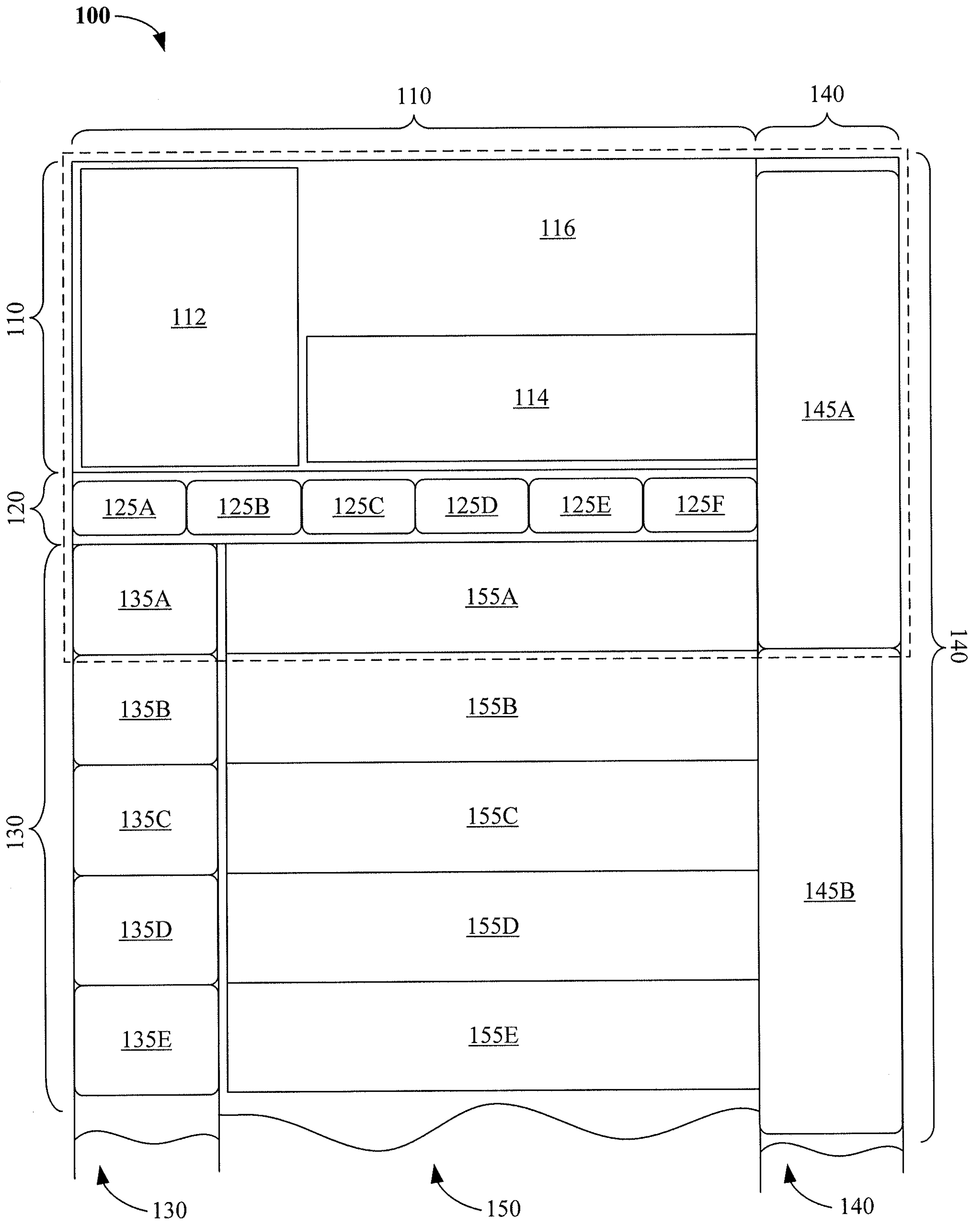


FIG. 1A

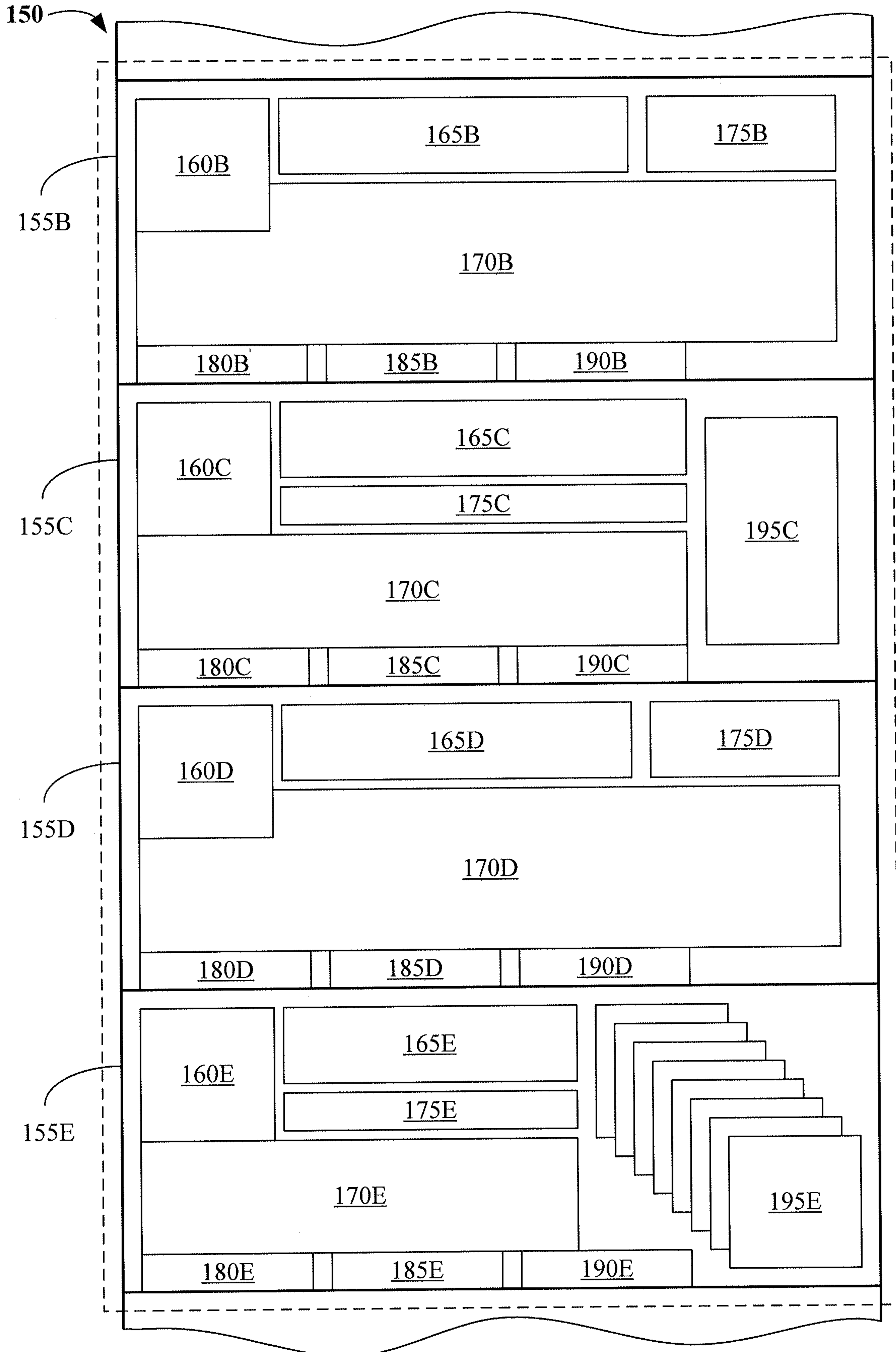
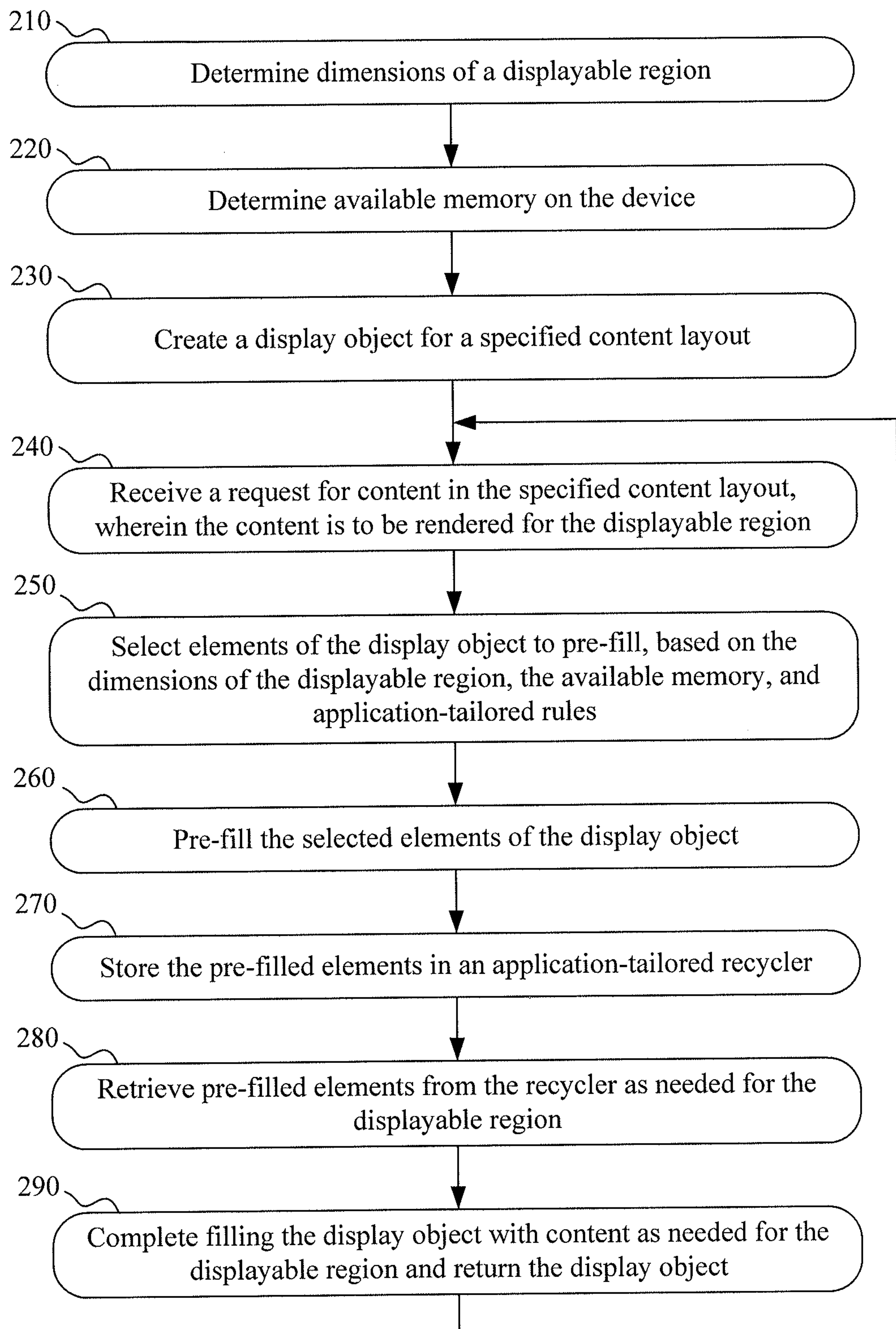


FIG. 1B

3/6

200**FIG. 2**

300

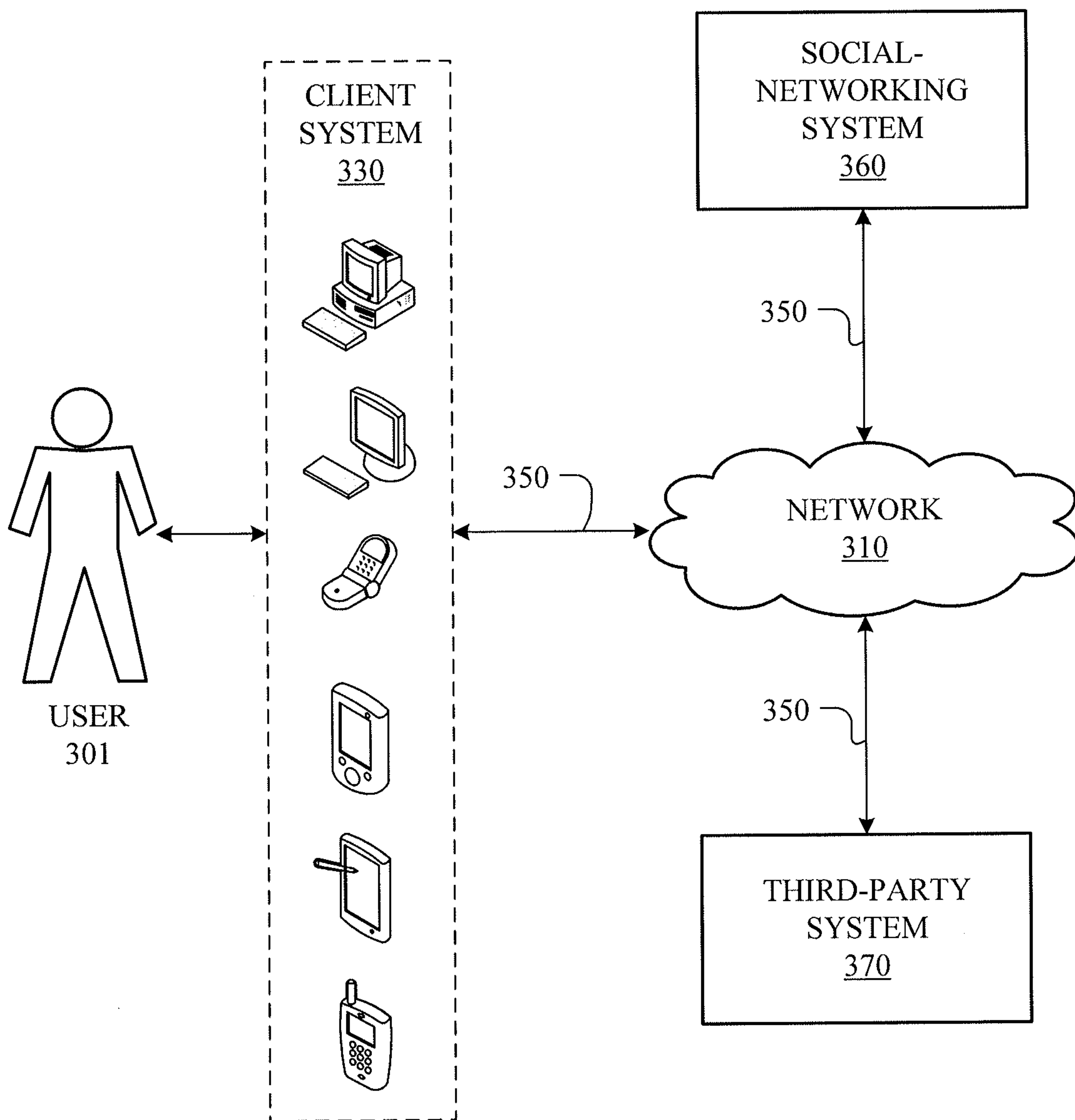


FIG. 3

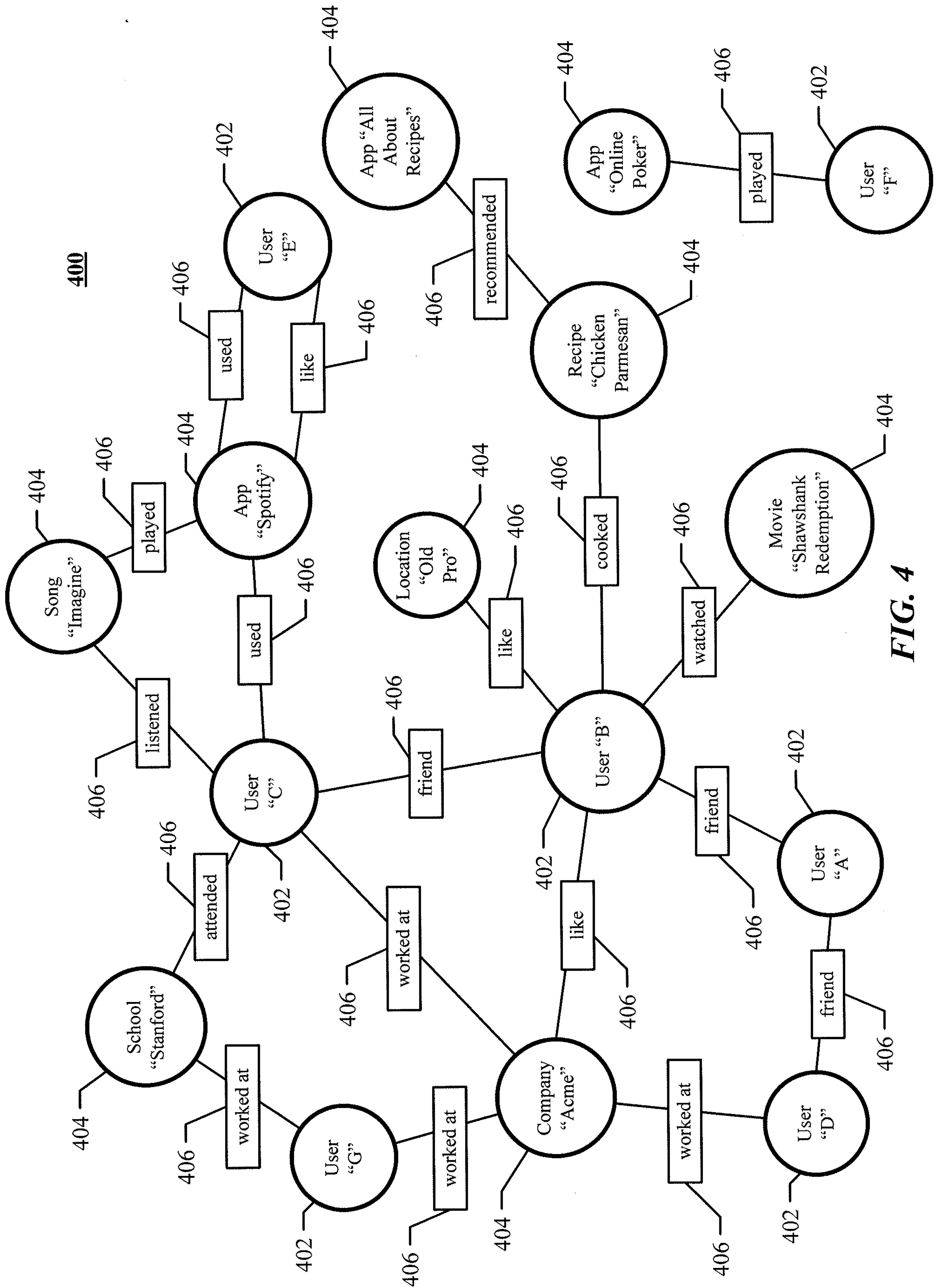


FIG. 4

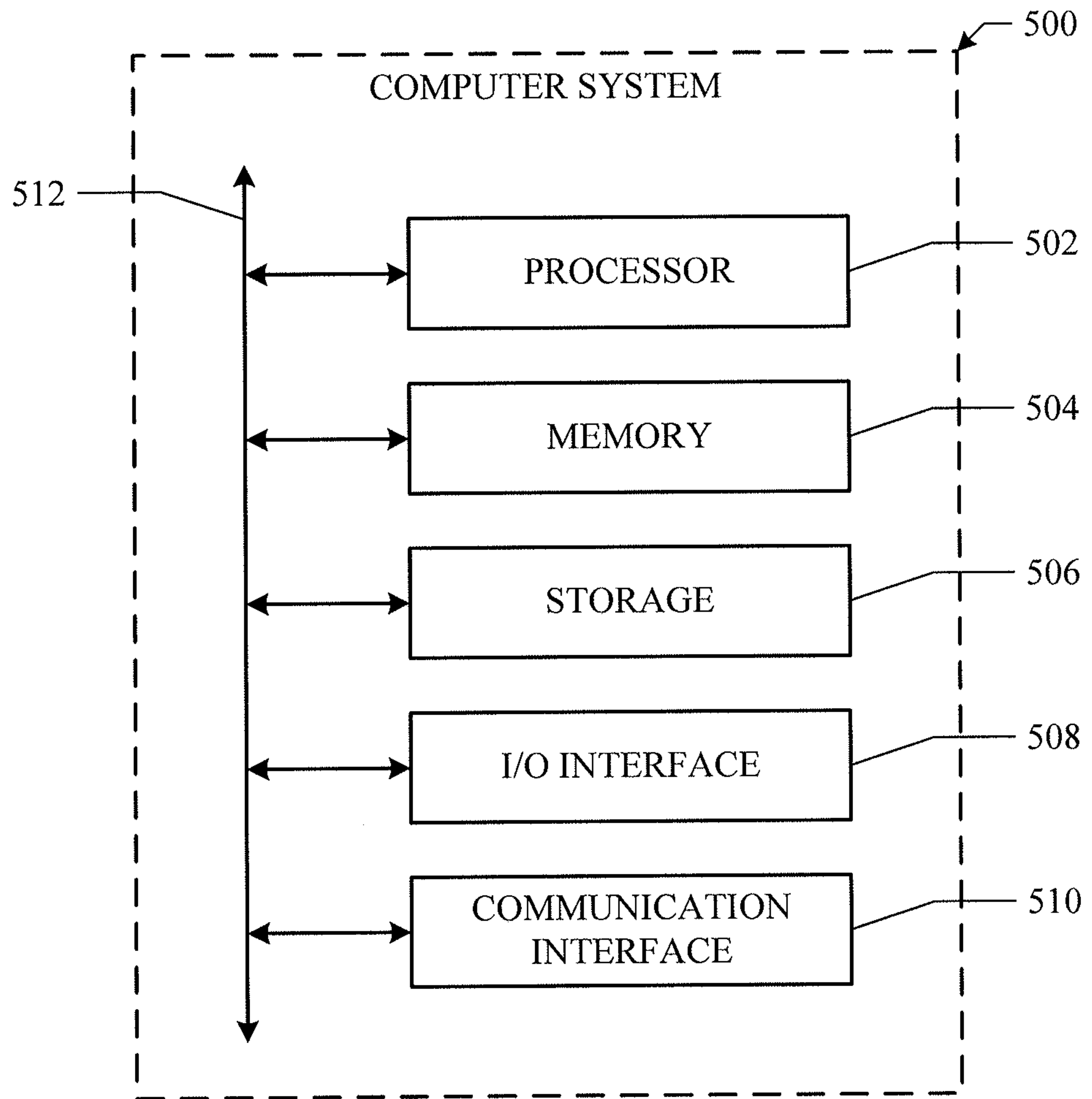


FIG. 5

