



- (51) **International Patent Classification:**
H04L 29/08 (2006.01) G06Q 50/10 (2012.01)
H04L 9/32 (2006.01)
- (21) **International Application Number:**
PCT/US2016/053164
- (22) **International Filing Date:**
22 September 2016 (22.09.2016)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
14/864,448 24 September 2015 (24.09.2015) US
- (71) **Applicant:** MCAFEE, INC. [US/US]; 2821 Mission College Blvd., Santa Clara, CA 95054 (US).
- (72) **Inventors:** LUPIAN, Nicolas; 2821 Mission College Blvd, Santa Clara, CA 95054 (US). LAKHIA, Michael; 2821 Mission College Blvd, Santa Clara, CA 95054 (US). GAGNON-LAMONDE, Hubert; 2821 Mission College Blvd, Santa Clara, CA 95054 (US).

(74) **Agent:** ABDULLAHI, Abdul; Blank Rome, LLP, 717 Texas Avenue, Suite 1400, Houston, TX 77002 (US).

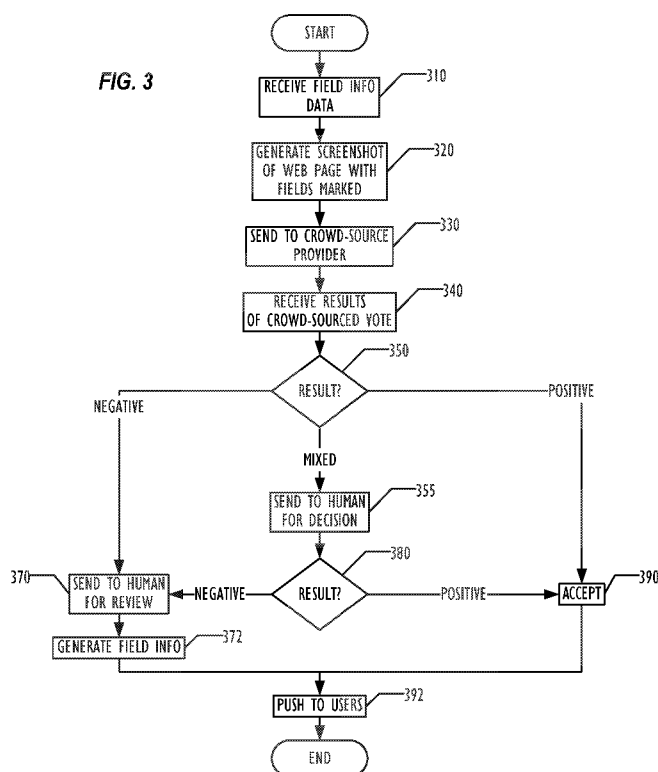
(81) **Designated States** (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KW, 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).

[Continued on next page]

(54) **Title:** CROWD-SOURCE AS A BACKUP TO ASYNCHRONOUS IDENTIFICATION OF A TYPE OF FORM AND RELEVANT FIELDS IN A CREDENTIAL-SEEKING WEB PAGE

FIG. 3



(57) **Abstract:** A technique allows a credential manager application on a client computer system to identify fields and forms on a web page. An analysis server may automatically crawl web pages and identify the fields and form, then push the information to the client computer system for use by the credential manager. If the credential manager discovers the information is not available, the credential manager may analyze the web form to discover the fields and form information, then provide the discovered information to the analysis server for providing to other client computers. The analysis server may use crowd-sourcing for asynchronous verification of field and form information discovered by the analysis server or provided by the client computer.



Published:

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

Title : **CROWD-SOURCE AS A BACKUP TO ASYNCHRONOUS IDENTIFICATION OF A
TYPE OF FORM AND RELEVANT FIELDS IN A CREDENTIAL-SEEKING WEB
PAGE**

CROWD-SOURCE AS A BACKUP TO ASYNCHRONOUS IDENTIFICATION OF A
TYPE OF FORM AND RELEVANT FIELDS IN A CREDENTIAL-SEEKING WEB PAGE
TECHNICAL FIELD

[0001] Embodiments described herein generally relate to client and server networks and, more particularly, to determining a location of enrollment fields in and the type of form, in a credential seeking web page, by using crowd-sourced information .

BACKGROUND ART

[0002] Users typically maintain a number of web-based accounts to personalize a web experience. Examples of such web-based accounts include email accounts, online shopping accounts, online banking accounts, online brokerage accounts, and the like. Most accounts may be accessed in a web browser over a personal computer, mobile device, smart device or other personal device as users may find it convenient to access these accounts on their personal devices when they are away from a desk or home computer. Each web-based account (referred to herein as a web account) requires a user to provide a username, a password, and/or other user credentials in, for example, a web browser to provide access to the web account. Each web account may present, in a web page, a web form to the user during initial login and subsequent access to the web account. This web form is a structured document that includes “form fields” for entering user identifier or credential information, such as a user ID (a user identifier), a password, or the like.

[0003] Today, applications are available such as, for example, credential manager applications that provide the ability to store user credentials and later be used for logging a user into the user’s online accounts using web pages received over the internet. These applications log the user into the online account by entering user credentials in one or more fields in a web form that is received in the web page. However, over time, web forms in web pages may be changed and location of enrollment fields in previous web pages may not be located in similar locations in the new web pages. Therefore, user credentials that are stored on a user device cannot be used in new web pages. A way of determining a location of enrollment fields in a credential-seeking web page would be desirable.

BRIEF DESCRIPTION OF DRAWINGS

[0004] FIG. 1 is a diagram illustrating a system for identifying enrollments fields using crowd-sourced data according to one embodiment.

[0005] FIG. 2 is a flowchart illustrating a technique for identifying fields and filling fields in a web page by a credential manager application according to one embodiment.

[0006] FIG. 3 is a flowchart illustrating a crowd-source assisted technique for identifying fields in a web page by a server according to one embodiment.

[0007] FIG. 4 is a diagram illustrating a computing device for use with techniques described herein according to one embodiment.

[0008] FIG. 5 is a block diagram illustrating a computing device for use with techniques described herein according to another embodiment.

[0009] FIG. 6 is a diagram illustrating a network of programmable devices according to one embodiment.

DESCRIPTION OF EMBODIMENTS

[0010] In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the invention. It will be apparent, however, to one skilled in the art that the invention may be practiced without these specific details. In other instances, structure and devices are shown in block diagram form in order to avoid obscuring the invention. References to numbers without subscripts or suffixes are understood to reference all instance of subscripts and suffixes corresponding to the referenced number. Moreover, the language used in this disclosure has been principally selected for readability and instructional purposes, and may not have been selected to delineate or circumscribe the inventive subject matter, resort to the claims being necessary to determine such inventive subject matter. Reference in the specification to “one embodiment” or to “an embodiment” means that a particular feature, structure, or characteristic described in connection with the embodiments is included in at least one embodiment of the invention, and multiple references to “one embodiment” or “an embodiment” should not be understood as necessarily all referring to the same embodiment.

[0011] As used herein, the term “computer system” can refer to a single computer or a plurality of computers working together to perform the function described as being performed on or by a computer system.

[0012] As used herein, the term “medium” can refer to a single physical medium or a plurality of media that together store the information described as being stored on the medium.

[0013] As used herein, the term “web crawler” can refer to an automated program, or script, that methodically scans or “crawls” through Internet pages to create an index of the data the web crawler is looking for. There are several uses for the program, perhaps the most popular being search engines using it to provide webs surfers with relevant websites. A web crawler can also be referred to as a web spider, a web robot, a bot, a crawler, and an automatic indexer.

[0014] As used herein, the term “headless browser” can refer to a web browser without a graphical user interface (GUI) that can access web pages over the internet but does not display them in the GUI on a client.

[0015] A technique allows a credential manager application on a client computer system to identify fields and forms on a web page. An analysis server may automatically crawl web pages and identify the fields and form, then push the information to the client computer system for use by the credential manager. If the credential manager discovers the information is not available, the credential manager may analyze the web form to discover the fields and form information, then provide the discovered information to the analysis server for providing to other client computers. The analysis server may use crowd-sourcing for asynchronous verification of field and form information discovered by the analysis server or provided by the client computer.

[0016] Referring to the figures, FIG. 1 illustrates a system **100** according to one embodiment. System **100** may include a web/content server **102**, analysis server **104**, network **106**, client **108**, and crowd-sourcing server **110**. While a single element of each type is illustrated in FIG. 1 for clarity, any number of each of the elements may be used as desired.

[0017] Web server **102** is a server that communicates one or more web pages to client **108** via network **106**. Web server **102** may transmit one or more HyperText Markup Language (HTML) web pages with web forms (also referred to as “login forms”) having enrollment fields to a web browser **112** on client **108** in response to a Hypertext Transfer Protocol (HTTP) request, for example, a HTTP GET request from client **108**. Web pages that are received from

web server **102** may also include HTML text and Cascading Style Sheets (CSS) data. The web server **102** may also transmit web pages to analysis server **104** for use by analysis server **104** in determining the location of enrollment fields and other fields of web forms in web pages, as will be described below.

[0018] Analysis server 104 may also include a database or knowledge base 105, for storing information that includes information about fields and forms associated with web pages that has been discovered either by the analysis server 104 or by the client 108. Although referred to as a database or knowledge base, no specific format or type of data storage functionality is intended by the term, and any desired or convenient way of storing the field and form information may be used. Although as illustrated the database 105 is connected directly to the analysis server 104, embodiments may provide the database 105 as a remotely hosted database connected to a database server or other computer system that is then connected via the network 106 to the analysis server 104.

[0019] The network 106 may be a single network or a collection of interconnected networks, and different elements of the system 100 may be connected to different ones of the interconnected networks. The network 106 may include the Internet as well as private or other public networks of any type, using any type of network protocol for communication, including Internet Protocol (IP) network protocols.

[0020] The crowd-sourcing server 110 provides a crowd-sourcing functionality for use by the analysis server 104. Crowd-sourcing server 110 typically provides a way for small tasks to be performed under conditions defined by a crowd-sourcing client (in this case, the analysis server 104). Individuals typically may agree to perform the tasks for a small payment. These individuals are generally not employed by the entity controlling the crowd-sourcing server 110, but may be otherwise unconnected individuals who have enrolled with the crowd-sourcing service provider. The crowd-sourcing service is generally provided by a third-party service provider, although the entity controlling the analysis server may provide similar services using in-house facilities. One example of a crowd-sourcing service provider is Amazon Technologies, Inc., which provides the Amazon Mechanical Turk® crowd-sourcing service. (AMAZON MECHANICAL TURK is a registered trademark of Amazon Technologies, Inc.) Other crowd-sourcing service providers may be used as desired. The analysis server 104 typically has no information about the individuals who perform the requested tasks. As

deployed in FIG. 1 and described in more detail below, simple questions with Yes-No answers are asked as part of the crowd-sourcing effort submitted to the crowd-sourcing server 110.

[0021] The analysis server 104 populates the database 105 with information regarding web pages and their respective forms and fields. The information stored in the database 105 may be sent from the client 108 and received by the analysis server 104 via the network 106 or may be generated by the analysis server 104 as a result of its web crawling functionality, described in more detail below.

[0022] The client 108 may be any type of programmable device, including any type of computer system, such as desktop, laptop, tablet, or other mobile device, and includes elements of typical computer systems, such as are described below in the description of FIGs. 5–6. The client 108 typically includes a web browser software 112, containing an HTML engine 114 and an HTML parser 116, which provide the functionality used for requesting, receiving, parsing, and displaying the web pages requested by the client 108. A credential manager application 118 provides secure management for credentials such as password, although other information such as credit card numbers may also be managed by the credential manager application 118. The operation of a credential manager application is generally outside the scope of the present invention and is described herein only for specific functionality of concern to the present disclosure. The client 108 may also maintain a cache 120 for caching information such as the field and form information used by the credential manager 118. The structure and implementation of the cache is not significant to the current disclosure, and any type of caching functionality may be used. The caching functionality 120 is typically non-volatile, such that the cached contents may survive a shutdown and restart of the client 108, but volatile cache storage techniques may be used as desired, such that a restart of the client 108 or the credential manager 118 may flush the cache 120.

[0023] Entries in the cache 120 may store information about the web page, the web server 102, fields and form information, and any other desired information. In one embodiment, timestamps or other fields may be provided to allow any or all of the cache entries as invalid and not to be used. For example, the credential manager may mark entries associated with a web page as invalid upon discover that the web page has changed since the entry was created. In one embodiment, the analysis server 104 may push cache entries to the client 108 for insertion into the cache 120, and may also instruct the credential manager 118 to mark one or more cache entries as invalid.

[0024] FIG. 2 is a flowchart illustrating a technique for a credential manager application 118 to use field and form information obtained from the analysis server 104 for inserting credentials into a web form of a web page served by web server 102. In block 210, the client 108 loads the web page from the web server 102. The credential manager 118 now tries to determine where the appropriate fields (if any) are on the web page for submission as a web form.

[0025] In block 220, the credential manager 118 checks the cache 120 to see if there are entries for the current web page. If any cache 120 entries are found in block 230 that are not marked invalid, the credential manager may then determine in block 260 whether the field and form information in those entries matches the actual current web page. If so, then in block 290 the credential manager 118 may use those entries to insert credentials into one or more of the fields of the web form on the web page.

[0026] If no valid cache 120 entries are found, then in block 240 the credential manager 118 may query the analysis server 104 for any field and form information the analysis server 104 may hold related to the current web page, causing the analysis server 104 to search the database 105. If the client 108 fails to receive any field or form information in the current web page from the analysis server 104, as determined in block 250, in block 270 the credential manager 118 may analyze the web page to locate fields and forms in the current web page. The credential manager may then provide the discovered fields and forms to the analysis server 104 in block 280, who may choose to push that newly discovered field and form information out to other clients 108.

[0027] Finally, the credential manager may use the field and form information to insert credentials into the web page in block 290.

[0028] If the credential manager 118 receives field and form information for the current web page from the analysis server 104 as determined in block 250, then the field and form information is checked in block 260 as described above, to determine whether the field and form information from the analysis server 104 matches the current web page. In no match exists, then the credential manager may proceed as if no field and form information was received in block 270, searching for fields in the web page. Not illustrated in FIG. 2, the cache 120 may at any time be updated by the analysis server 104 for use in the illustrated procedure.

[0029] FIG. 3 is a flowchart illustrating an asynchronous technique for the analysis server 104 to discover fields and form information in web pages. In addition to receiving field and

form information from clients 108 as described above, the analysis server 104 may include other techniques to automatically discover field and form information. In one embodiment, the analysis server employs a web crawler to crawl the web for web pages that may have forms for submitting credentials. Web crawling techniques are well known and are not further described herein. In one embodiment, the web crawler employed by the analysis server 104 may limit the crawling to web pages identified by a third party web page ranking resource as highly popular web pages. The analysis server may then prefill the cache 120 of clients 108 with information regarding such popular web pages, pushing the information via the network 106 to the clients 108. This crawling technique is asynchronous to the client 108's activity.

[0030] FIG. 3 illustrates a procedure for verifying that field and form information, whether received from client 108 or asynchronously discovered by web crawling. In block 310, the analysis server may receive the field and form information, either from the client 108 or the web crawler. In block 320, the analysis server may obtain the page and create a screenshot showing what a user would see on the screen if that web page were displayed by a browser. In one embodiment, this may be achieved by executing a headless browser to format the data that would be displayed, however without an actual display. The screen shot may then be annotated to mark the position of the fields corresponding to the field and form information, in some embodiments identifying the type of field, such as whether the field is a password field. Any type of visual marking may be used, such as surrounding the field with a border of a contrasting color.

[0031] In block 330, the annotated screenshot may be sent to the crowd-sourcing service provider via server 110, with a request for crowd-sourced validation of the fields. In one embodiment, the server 110 is requested to have 3 individuals review the annotated screenshot and respond with a simple yes/no answer to a question of whether the fields are correctly identified and marked. The question may be communicated in any desired way, including either separately from the screenshot or contained in the screenshot. Upon receiving results of the decisions by the crowd sourced individuals in block 340, in block 350 the analysis server analyses the results. In one embodiment, a positive result comprises all three of the crowd sourcers responding Yes to the question, saying that the fields and form information are correct as they are held by the analysis server 104 and sent to the crowd sourcer by the crowd sourcing server 110. Similarly a negative response comprises no more than one of the three crowd sourcers voting Yes, and a mixed response comprises two of the three voting Yes, but one of

the three voting No. Using three crowd sourcers is illustrative and by way of example only, and any number may be used.

[0032] If the result of the vote in block 350 is positive, then in block 390 the field and form information may be accepted and pushed out to the caches 120 for use by the credential managers 118 on the clients 108.

[0033] If the result of the vote is negative, the screenshot's annotation of the web page is considered incorrect and the screenshot may be presented to another human being in block 370 to allow the human being to generate a new set of field and form information in block 372. The new and presumably corrected information may then be pushed to the user caches 120 in block 392.

[0034] If the result of the vote is mixed, the screenshot may be sent to another human being in block 355 for a final decision on the validity of the field and form annotations of the web page. If the human arbiter's decision is positive, as determined in block 380, then the procedure accepts the field and form information as valid and pushes the information to clients 108 in block 392. Similarly, if the human arbiter's decision is negative, the actions of block 370, 372, and 392 may be performed as described above.

[0035] In another embodiment, instead of or in addition to sending the annotated screenshots to a crowd-sourcing service provider at server 110, other techniques may be used. For example, computer vision techniques may be used to consider the annotated screenshots and use machine learning techniques directly in either a headless browser or a GUI browser, with or without marked fields to generate a decision on whether the field and form information is correct, given the decision a confidence level. A positive decision would then be a decision with a high confidence level that the field and form information is correct; a negative decision would be a decision with a high confidence level that the field and form information is incorrect, and a mixed decision would be a decision with a lower confidence level. In one embodiment a high confidence level that the field and form information is correct may be a confidence level that exceeds a first predetermined threshold, a high confidence level that the field is incorrect may be a confidence level that is lower than a second predetermined threshold, and a mixed decision may be a confidence level between the first and second predetermined thresholds. The processing of the computer-vision guided embodiment may then follow the procedure of blocks 350–392 of FIG. 3. Other embodiments may combine computer vision and

crowd sourcing or use other automatic or semi-automatic techniques for verifying or validating the discovered field and form information.

[0036] By using these techniques, crowd-sourcing may be used as a backup to asynchronously discovering field and form information by the analysis server 104, providing a better user experience for the user of client 108, by detecting erroneous decisions about field and form information. These techniques are scalable, because any number of analysis servers may be used for performing the web crawling and backend analysis, and because any number of crowd sources may be employed by the crowd sourcing service provider without the need for the analysis server provider to hire dedicated staff to review and make decisions on field and form information at the level that would be needed to review very large numbers of web pages.

[0037] In one embodiment, if the web crawler of the analysis server 104 detects that a web page has changed since a previous crawl, the analysis server 104 may instruct the clients 108 to invalidate the corresponding entry in the cache 120. Similarly, if the analysis procedure illustrated in FIG. 3 determines that the field and form information is incorrect, the analysis server 104 may instruct the clients 108 to invalidate the corresponding entry in the cache 120 before pushing the new field and form information to the clients 108 for storing in the cache.

[0038] Referring now to FIG. 4, a block diagram illustrates a programmable device **400** that may be used as the analysis server 104 or the client **108** in accordance with one embodiment. The programmable device **400** illustrated in FIG. 4 is a multiprocessor programmable device that includes a first processing element **470** and a second processing element **480**. While two processing elements **470** and **480** are shown, an embodiment of programmable device **400** may also include only one such processing element.

[0039] Programmable device **400** is illustrated as a point-to-point interconnect system, in which the first processing element **470** and second processing element **480** are coupled via a point-to-point interconnect **450**. Any or all of the interconnects illustrated in FIG. 4 may be implemented as a multi-drop bus rather than point-to-point interconnects.

[0040] As illustrated in FIG. 4, each of processing elements **470** and **480** may be multicore processors, including first and second processor cores (i.e., processor cores **474a** and **474b** and processor cores **484a** and **484b**). Such cores **474a**, **474b**, **484a**, **484b** may be configured to execute instruction code in a manner similar to that discussed above in connection with FIGs. 1–3. However, other embodiments may use processing elements that are single core processors

as desired. In embodiments with multiple processing elements **470**, **480**, each processing element may be implemented with different numbers of cores as desired.

[0041] Each processing element **470**, **480** may include at least one shared cache **446**. The shared cache **446a**, **446b** may store data (e.g., instructions) that are utilized by one or more components of the processing element, such as the cores **474a**, **474b** and **484a**, **484b**, respectively. For example, the shared cache may locally cache data stored in a memory **432**, **434** for faster access by components of the processing elements **570**, **580**. In one or more embodiments, the shared cache **446a**, **446b** may include one or more mid-level caches, such as level 2 (L2), level 3 (L3), level 4 (L4), or other levels of cache, a last level cache (LLC), or combinations thereof.

[0042] While FIG. 4 illustrates a programmable device with two processing elements **470**, **480** for clarity of the drawing, the scope of the present invention is not so limited and any number of processing elements may be present. Alternatively, one or more of processing elements **470**, **480** may be an element other than a processor, such as a graphics processing unit (GPU), a digital signal processing (DSP) unit, a field programmable gate array, or any other programmable processing element. Processing element **480** may be heterogeneous or asymmetric to processing element **470**. There may be a variety of differences between processing elements **470**, **480** in terms of a spectrum of metrics of merit including architectural, microarchitectural, thermal, power consumption characteristics, and the like. These differences may effectively manifest themselves as asymmetry and heterogeneity amongst processing elements **470**, **480**. In some embodiments, the various processing elements **470**, **480** may reside in the same die package.

[0043] First processing element **470** may further include memory controller logic (MC) **472** and point-to-point (P-P) interconnects **476** and **478**. Similarly, second processing element **480** may include a MC **482** and P-P interconnects **486** and **488**. As illustrated in FIG. 4, MCs **472** and **482** couple processing elements **470**, **480** to respective memories, namely a memory **432** and a memory **434**, which may be portions of main memory locally attached to the respective processors. While MC logic **472** and **482** is illustrated as integrated into processing elements **470**, **480**, in some embodiments the memory controller logic may be discrete logic outside processing elements **470**, **480** rather than integrated therein.

[0044] Processing element **470** and processing element **480** may be coupled to an I/O subsystem **490** via respective P-P interconnects **476** and **486** through links **452** and **454**. As

illustrated in FIG. 4, I/O subsystem 490 includes P-P interconnects 494 and 498. Furthermore, I/O subsystem 490 includes an interface 492 to couple I/O subsystem 490 with a high performance graphics engine 438. In one embodiment, a bus (not shown) may be used to couple graphics engine 438 to I/O subsystem 490. Alternately, a point-to-point interconnect 439 may couple these components.

[0045] In turn, I/O subsystem 490 may be coupled to a first link 416 via an interface 496. In one embodiment, first link 416 may be a Peripheral Component Interconnect (PCI) bus, or a bus such as a PCI Express bus or another I/O interconnect bus, although the scope of the present invention is not so limited.

[0046] As illustrated in FIG. 4, various I/O devices 414, 424 may be coupled to first link 416, along with a bridge 418 that may couple first link 416 to a second link 410. In one embodiment, second link 420 may be a low pin count (LPC) bus. Various devices may be coupled to second link 420 including, for example, a keyboard/mouse 412, communication device(s) 426 (which may in turn be in communication with the computer network 403), and a data storage unit 428 such as a disk drive or other mass storage device which may include code 430, in one embodiment. The code 430 may include instructions for performing embodiments of one or more of the techniques described above. Further, an audio I/O 424 may be coupled to second link 420.

[0047] Note that other embodiments are contemplated. For example, instead of the point-to-point architecture of FIG. 4, a system may implement a multi-drop bus or another such communication topology. Although links 416 and 420 are illustrated as busses in FIG. 4, any desired type of link may be used. In addition, the elements of FIG. 4 may alternatively be partitioned using more or fewer integrated chips than illustrated in FIG. 4.

[0048] Referring now to FIG. 5, a block diagram illustrates a programmable device 500 according to another embodiment. Certain aspects of FIG. 5 have been omitted from FIG. 5 in order to avoid obscuring other aspects of FIG. 5.

[0049] FIG. 5 illustrates that processing elements 570, 580 may include integrated memory and I/O control logic ("CL") 572 and 582, respectively. In some embodiments, the 572, 582 may include memory control logic (MC) such as that described above in connection with FIG. 5. In addition, CL 572, 582 may also include I/O control logic. FIG. 5 illustrates that not only may the memories 532, 534 be coupled to the 572, 582, but also that I/O devices 544 may also be coupled to the control logic 572, 582. Legacy I/O devices 515 may be coupled to the I/O

subsystem **590** by interface **596**. Each processing element **570**, **580** may include multiple processor cores, illustrated in FIG. **5** as processor cores **574A**, **574B**, **584A** and **584B**. As illustrated in FIG. **5**, I/O subsystem **590** includes point-to-point (P-P) interconnects **594** and **598** that connect to P-P interconnects **576** and **586** of the processing elements **570** and **580** with links **552** and **554**. Processing elements **570** and **580** may also be interconnected by link **550** and interconnects **578** and **588**, respectively.

[0050] The programmable devices depicted in FIGs. **4** and **5** are schematic illustrations of embodiments of programmable devices that may be utilized to implement various embodiments discussed herein. Various components of the programmable devices depicted in FIGs. **4** and **5** may be combined in a system-on-a-chip (SoC) architecture.

[0051] Referring now to FIG. **6**, an example infrastructure **600** in which the techniques described above may be implemented is illustrated schematically. Infrastructure **600** contains computer networks **602**. Computer networks **602** may include many different types of computer networks available today, such as the Internet, a corporate network or a Local Area Network (LAN). Each of these networks can contain wired or wireless programmable devices and operate using any number of network protocols (*e.g.*, TCP/IP). Networks **602** may be connected to gateways and routers (represented by **608**), end user computers **606**, and computer servers **604**. Infrastructure **600** also includes cellular network **603** for use with mobile communication devices. Mobile cellular networks support mobile phones and many other types of mobile devices. Mobile devices in the infrastructure **600** are illustrated as mobile phones **610**, laptops **612** and tablets **614**. A mobile device such as mobile phone **610** may interact with one or more mobile provider networks as the mobile device moves, typically interacting with a plurality of mobile network towers **620**, **630**, and **640** for connecting to the cellular network **603**. Although referred to as a cellular network in FIG. **6**, a mobile device may interact with towers of more than one provider network, as well as with multiple non-cellular devices such as wireless access points and routers **608**. In addition, the mobile devices **610**, **612** and **614** may interact with non-mobile devices such as computers **604** and **606** for desired services, which may include determining web forms and field information in web pages that is provided to client **108** described above.

[0052] The following examples pertain to further embodiments.

[0053] Example 1 is a machine readable medium, on which are stored instructions, comprising instructions that when executed cause a machine to: receive a web page for a web

site over a network by an analysis server; discover field and form information for the web page; validate the field and form information using a crowd-sourcing service; accept the field and form information as validated field and form information responsive to a positive result from the crowd-sourcing service; receive a corrected field and form information from a human reviewer responsive to a negative result from the crowd-sourcing service; and send the validated or corrected field and form information from the analysis server to a credential manager application.

[0054] In Example 2 the subject matter of Example 1 optionally includes wherein the instructions further comprise instructions that when executed cause the machine to instruct the credential manager application to invalidate some or all of a cache maintained by the credential manager application.

[0055] In Example 3 the subject matter of Example 1 optionally includes wherein the instructions further comprise instructions that when executed cause the machine to send the field and form information to a human arbiter responsive to a mixed result from the crowd-sourcing service.

[0056] In Example 4 the subject matter of Examples 1–3 optionally includes wherein the instructions to validate the field and form information using a crowd-sourcing service comprise instructions that when executed cause the machine to: generate a screenshot of the web page; mark the fields on the screenshot; and send the screenshot to the crowd-sourcing service for validation.

[0057] In Example 5 the subject matter of Examples 1–3 optionally includes wherein the instructions to validate the field and form information using a crowd-sourcing service comprise instructions that when executed cause the machine to: request at least three crowd-sources responses from the crowd-sourcing service; and consider the at least three crowd-source responses as a voting result.

[0058] In Example 6 the subject matter of Examples 1–3 optionally includes wherein the instructions further comprise instructions that when executed cause the machine to: use computer vision to view a screenshot of the web page annotated corresponding to the field and form information; and return a decision based on computer vision that indicates a confidence level that the field and form information is correct.

[0059] In Example 7 the subject matter of Example 6 optionally includes wherein the instructions further comprise instructions that when executed cause the machine to: pass the screenshot to a human reviewer responsive to the confidence level being lower than a predetermined threshold.

[0060] In Example 8 the subject matter of Examples 1–3 optionally includes wherein the instructions that when executed cause the machine to receive the web page comprise instructions that when executed cause the machine to: employ a web crawler for examining web pages.

[0061] In Example 9 the subject matter of Example 8 optionally includes wherein the web crawler is configured to crawl a predetermined set of popular web pages.

[0062] Example 10 is a computer system for determining web form information in a web page for a web site comprising: one or more processors; and a memory coupled to the one or more processors, on which are stored instructions, comprising instructions that when executed cause at least some of the one or more of the processors to: receive a web page for a web site over a network by an analysis server; discover field and form information for the web page; validate the field and form information using a crowd-sourcing service; accept the field and form information as validated field and form information responsive to a positive result from the crowd-sourcing service; receive a corrected field and form information from a human reviewer responsive to a negative result from the crowd-sourcing service; and send the validated or corrected field and form information from the analysis server to a credential manager application.

[0063] In Example 11 the subject matter of Example 10 optionally includes wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to instruct the credential manager application to invalidate some or all of a cache maintained by the credential manager application.

[0064] In Example 12 the subject matter of Example 10 optionally includes wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to send the field and form information to a human arbiter responsive to a mixed result from the crowd-sourcing service.

[0065] In Example 13 the subject matter of Examples 10–12 optionally includes wherein the instructions to validate the field and form information using a crowd-sourcing service

comprise instructions that when executed cause at least some of the one or more processors to: generate a screenshot of the web page; mark the fields on the screenshot; and send the screenshot to the crowd-sourcing service for validation.

[0066] In Example 14 the subject matter of Examples 10–12 optionally includes wherein the instructions to validate the field and form information using a crowd-sourcing service comprise instructions that when executed cause at least some of the one or more processors to: request at least three crowd-sources responses from the crowd-sourcing service; and consider the at least three crowd-source responses as a voting result.

[0067] In Example 15 the subject matter of Examples 10–12 optionally includes wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to: use computer vision to view a screenshot of the web page annotated corresponding to the field and form information; and return a decision based on computer vision that indicates a confidence level that the field and form information is correct.

[0068] In Example 16 the subject matter of Example 15 optionally includes wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to: pass the screenshot to a human reviewer responsive to the confidence level being lower than a predetermined threshold.

[0069] In Example 17 the subject matter of Examples 10–12 optionally includes wherein the instructions that when executed cause at least some of the one or more processors to receive the web page comprise instructions that when executed cause at least some of the one or more processors to: employ a web crawler for examining web pages.

[0070] In Example 18 the subject matter of Example 17 optionally includes wherein the web crawler is configured to crawl a predetermined set of popular web pages.

[0071] Example 19 is a method for determining web form information in a web page for a web site, comprising: receiving a web page for a web site over a network by an analysis server; discovering field and form information for the web page; validating the field and form information using a crowd-sourcing service; accepting the field and form information as validated field and form information responsive to a positive result from the crowd-sourcing service; receiving a corrected field and form information from a human reviewer responsive to a negative result from the crowd-sourcing service; and sending the validated or corrected field and form information from the analysis server to a credential manager application.

[0072] In Example 20 the subject matter of Example 19 optionally includes further comprising instructing the credential manager application to invalidate some or all of a cache maintained by the credential manager application.

[0073] In Example 21 the subject matter of Example 19 optionally includes further comprising sending the field and form information to a human arbiter responsive to a mixed result from the crowd-sourcing service.

[0074] In Example 22 the subject matter of Examples 19–21 optionally includes wherein validating the field and form information using a crowd-sourcing service comprises: generating a screenshot of the web page; marking the fields on the screenshot; and sending the screenshot to the crowd-sourcing service for validation.

[0075] In Example 23 the subject matter of Examples 19–21 optionally includes wherein validating the field and form information using a crowd-sourcing service comprises: requesting at least three crowd-sources responses from the crowd-sourcing service; and considering the at least three crowd-source responses as a voting result.

[0076] In Example 24 the subject matter of Examples 19–21 optionally includes further comprising: using computer vision to view a screenshot of the web page annotated corresponding to the field and form information; and return a decision based on computer vision that indicates a confidence level that the field and form information is correct.

[0077] In Example 25 the subject matter of Examples 19–21 optionally includes wherein receiving the web page comprise: employing a web crawler for examining web pages.

[0078] Example 26 is a computer system, comprising: means for receiving a web page for a web site over a network by an analysis server; means for discovering field and form information for the web page; means for validating the field and form information using a crowd-sourcing service; means for accepting the field and form information as validated field and form information responsive to a positive result from the crowd-sourcing service; means for receiving a corrected field and form information from a human reviewer responsive to a negative result from the crowd-sourcing service; and means for sending the validated or corrected field and form information from the analysis server to a credential manager application.

[0079] In Example 27 the subject matter of Example 26 optionally includes further means for instructing the credential manager application to invalidate some or all of a cache maintained by the credential manager application.

[0080] In Example 28 the subject matter of Example 26 optionally includes further comprising sending the field and form information to a human arbiter responsive to a mixed result from the crowd-sourcing service.

[0081] In Example 29 the subject matter of Examples 26–28 optionally includes wherein the means for validating the field and form information using a crowd-sourcing service comprise: means for generating a screenshot of the web page; means for marking the fields on the screenshot; and means for sending the screenshot to the crowd-sourcing service for validation.

[0082] In Example 30 the subject matter of Examples 26–28 optionally includes wherein the means for validating the field and form information using a crowd-sourcing service comprise: means for requesting at least three crowd-sources responses from the crowd-sourcing service; and means for considering the at least three crowd-source responses as a voting result.

[0083] In Example 31 the subject matter of Examples 26–28 optionally includes further comprising: means for using computer vision to view a screenshot of the web page annotated corresponding to the field and form information; and means for returning a decision based on computer vision that indicates a confidence level that the field and form information is correct.

[0084] In Example 32 the subject matter of Example 31 optionally includes further comprising: means for passing the screenshot to a human reviewer responsive to the confidence level being lower than a predetermined threshold.

[0085] In Example 33 the subject matter of Examples 26–28 optionally includes wherein the means for receiving the web page comprise: means for employing a web crawler for examining web pages.

[0086] In Example 34 the subject matter of Example 33 optionally includes wherein the web crawler is configured to crawl a predetermined set of popular web pages.

[0087] It is to be understood that the above description is intended to be illustrative, and not restrictive. For example, the above-described embodiments may be used in combination with each other. Many other embodiments will be apparent to those of skill in the art upon reviewing the above description. The scope of the invention therefore should be determined

with reference to the appended claims, along with the full scope of equivalents to which such claims are entitled.

CLAIMS

What is claimed is:

1. A machine readable medium, on which are stored instructions, comprising instructions that when executed cause a machine to:
 - receive a web page for a web site over a network by an analysis server;
 - discover field and form information for the web page;
 - validate the field and form information using a crowd-sourcing service;
 - accept the field and form information as validated field and form information responsive to a positive result from the crowd-sourcing service;
 - receive a corrected field and form information from a human reviewer responsive to a negative result from the crowd-sourcing service; and
 - send the validated or corrected field and form information from the analysis server to a credential manager application.
2. The machine readable medium of claim 1, wherein the instructions further comprise instructions that when executed cause the machine to instruct the credential manager application to invalidate some or all of a cache maintained by the credential manager application.
3. The machine readable medium of claim 1, wherein the instructions further comprise instructions that when executed cause the machine to send the field and form information to a human arbiter responsive to a mixed result from the crowd-sourcing service.
4. The machine readable medium of any of claims 1-3, wherein the instructions to validate the field and form information using a crowd-sourcing service comprise instructions that when executed cause the machine to:
 - generate a screenshot of the web page;
 - mark the fields on the screenshot; and
 - send the screenshot to the crowd-sourcing service for validation.

5. The machine readable medium of any of claims 1-3, wherein the instructions to validate the field and form information using a crowd-sourcing service comprise instructions that when executed cause the machine to:

request at least three crowd-sources responses from the crowd-sourcing service;

and

consider the at least three crowd-source responses as a voting result.

6. The machine readable medium of any of claims 1-3, wherein the instructions further comprise instructions that when executed cause the machine to:

use computer vision to view a screenshot of the web page annotated corresponding to the field and form information; and

return a decision based on computer vision that indicates a confidence level that the field and form information is correct.

7. The machine readable medium of claim 6, wherein the instructions further comprise instructions that when executed cause the machine to:

pass the screenshot to a human reviewer responsive to the confidence level being lower than a predetermined threshold.

8. The machine readable medium of any of claims 1-3, wherein the instructions that when executed cause the machine to receive the web page comprise instructions that when executed cause the machine to:

employ a web crawler for examining web pages.

9. The machine readable medium of claim 8, wherein the web crawler is configured to crawl a predetermined set of popular web pages.

10. A computer system for determining web form information in a web page for a web site comprising:

one or more processors; and

a memory coupled to the one or more processors, on which are stored instructions, comprising instructions that when executed cause at least some of the one or more of the processors to:

receive a web page for a web site over a network by an analysis server;

discover field and form information for the web page;
validate the field and form information using a crowd-sourcing service;
accept the field and form information as validated field and form information responsive to a positive result from the crowd-sourcing service;
receive a corrected field and form information from a human reviewer responsive to a negative result from the crowd-sourcing service; and
send the validated or corrected field and form information from the analysis server to a credential manager application.

11. The computer system of claim 10, wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to instruct the credential manager application to invalidate some or all of a cache maintained by the credential manager application.

12. The computer system of claim 10, wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to send the field and form information to a human arbiter responsive to a mixed result from the crowd-sourcing service.

13. The computer system of any of claims 10-12, wherein the instructions to validate the field and form information using a crowd-sourcing service comprise instructions that when executed cause at least some of the one or more processors to:

generate a screenshot of the web page;
mark the fields on the screenshot; and
send the screenshot to the crowd-sourcing service for validation.

14. The computer system of any of claims 10-12, wherein the instructions to validate the field and form information using a crowd-sourcing service comprise instructions that when executed cause at least some of the one or more processors to:

request at least three crowd-sources responses from the crowd-sourcing service;
and
consider the at least three crowd-source responses as a voting result.

15. The computer system of any of claims 10-12, wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to:
- use computer vision to view a screenshot of the web page annotated corresponding to the field and form information; and
 - return a decision based on computer vision that indicates a confidence level that the field and form information is correct.
16. The computer system of claim 15, wherein the instructions further comprise instructions that when executed cause at least some of the one or more processors to:
- pass the screenshot to a human reviewer responsive to the confidence level being lower than a predetermined threshold.
17. The computer system of any of claims 10-12, wherein the instructions that when executed cause at least some of the one or more processors to receive the web page comprise instructions that when executed cause at least some of the one or more processors to:
- employ a web crawler for examining web pages.
18. The computer system of claim 17, wherein the web crawler is configured to crawl a predetermined set of popular web pages.
19. A method for determining web form information in a web page for a web site, comprising:
- receiving a web page for a web site over a network by an analysis server;
 - discovering field and form information for the web page;
 - validating the field and form information using a crowd-sourcing service;
 - accepting the field and form information as validated field and form information responsive to a positive result from the crowd-sourcing service;
 - receiving a corrected field and form information from a human reviewer responsive to a negative result from the crowd-sourcing service; and
 - sending the validated or corrected field and form information from the analysis server to a credential manager application.

20. The method of claim 19, further comprising instructing the credential manager application to invalidate some or all of a cache maintained by the credential manager application.

21. The method of claim 19, further comprising sending the field and form information to a human arbiter responsive to a mixed result from the crowd-sourcing service.

22. The method of any of claims 19-21, wherein validating the field and form information using a crowd-sourcing service comprises:

- generating a screenshot of the web page;
- marking the fields on the screenshot; and
- sending the screenshot to the crowd-sourcing service for validation.

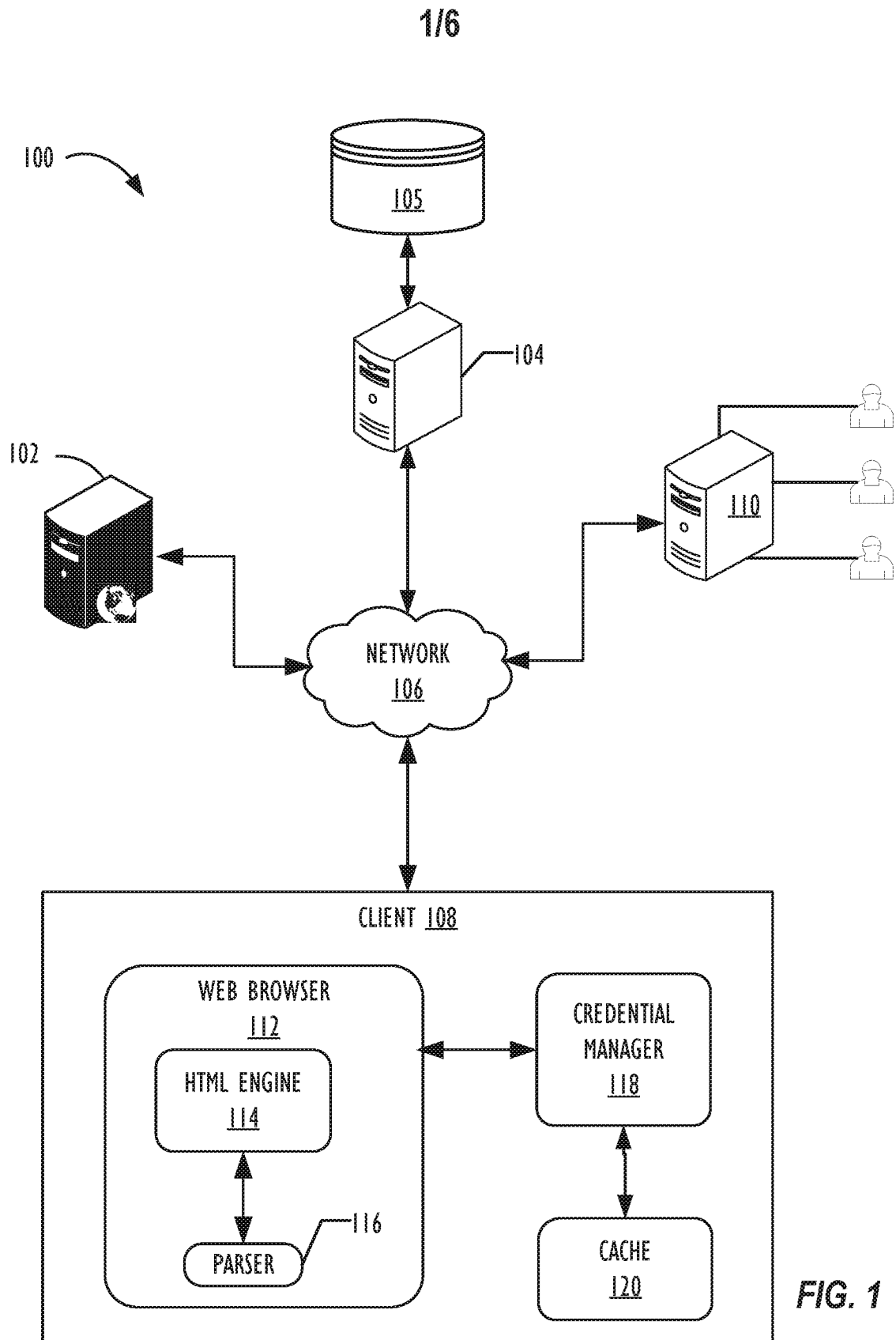
23. The method of any of claims 19-21, wherein validating the field and form information using a crowd-sourcing service comprises:

- requesting at least three crowd-sources responses from the crowd-sourcing service; and
- considering the at least three crowd-source responses as a voting result.

24. The method of any of claims 19-21, further comprising:

- using computer vision to view a screenshot of the web page annotated corresponding to the field and form information; and
- return a decision based on computer vision that indicates a confidence level that the field and form information is correct.

25. The method of any of claims 19-21, wherein receiving the web page comprise: employing a web crawler for examining web pages.



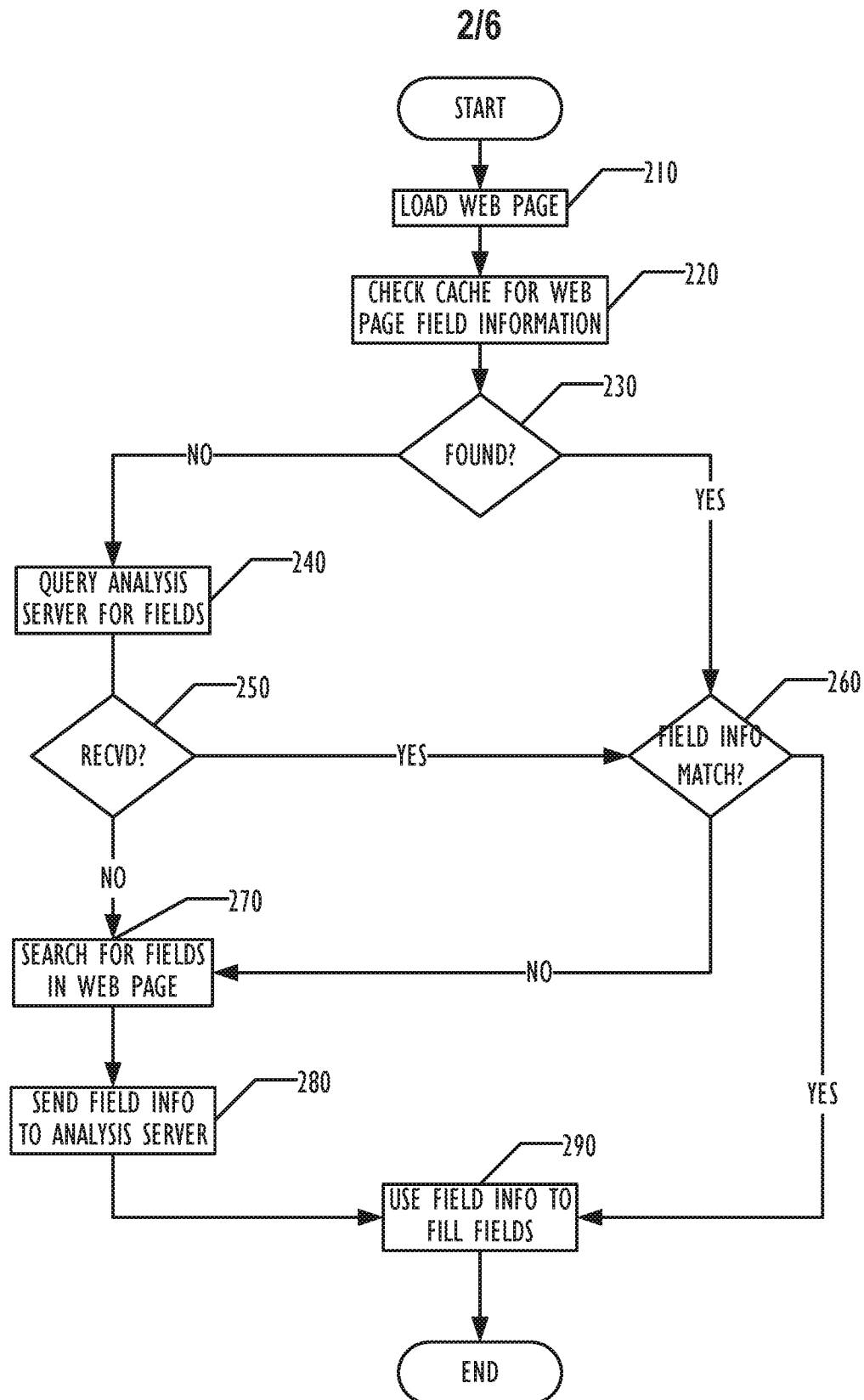


FIG. 2

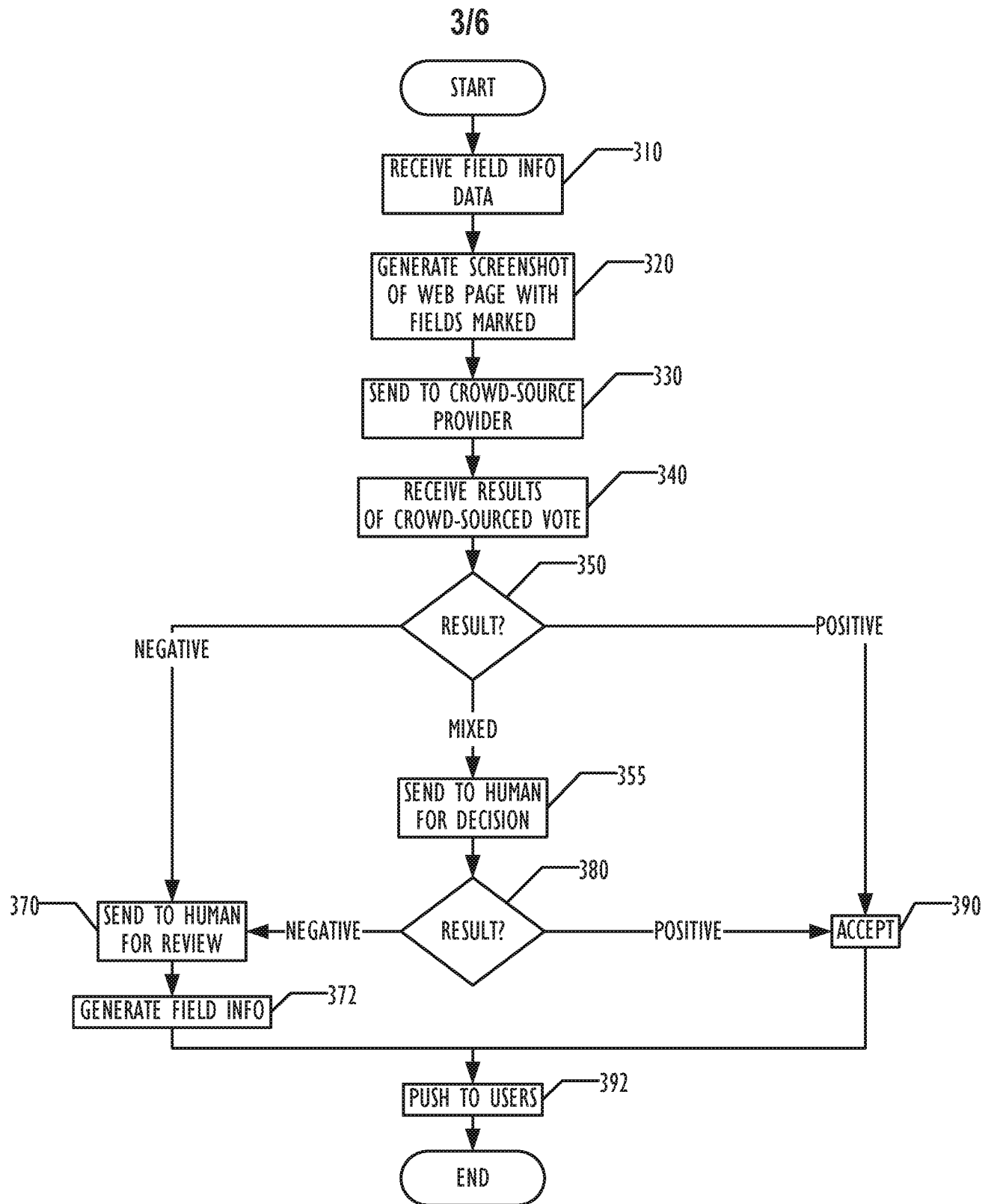


FIG. 3

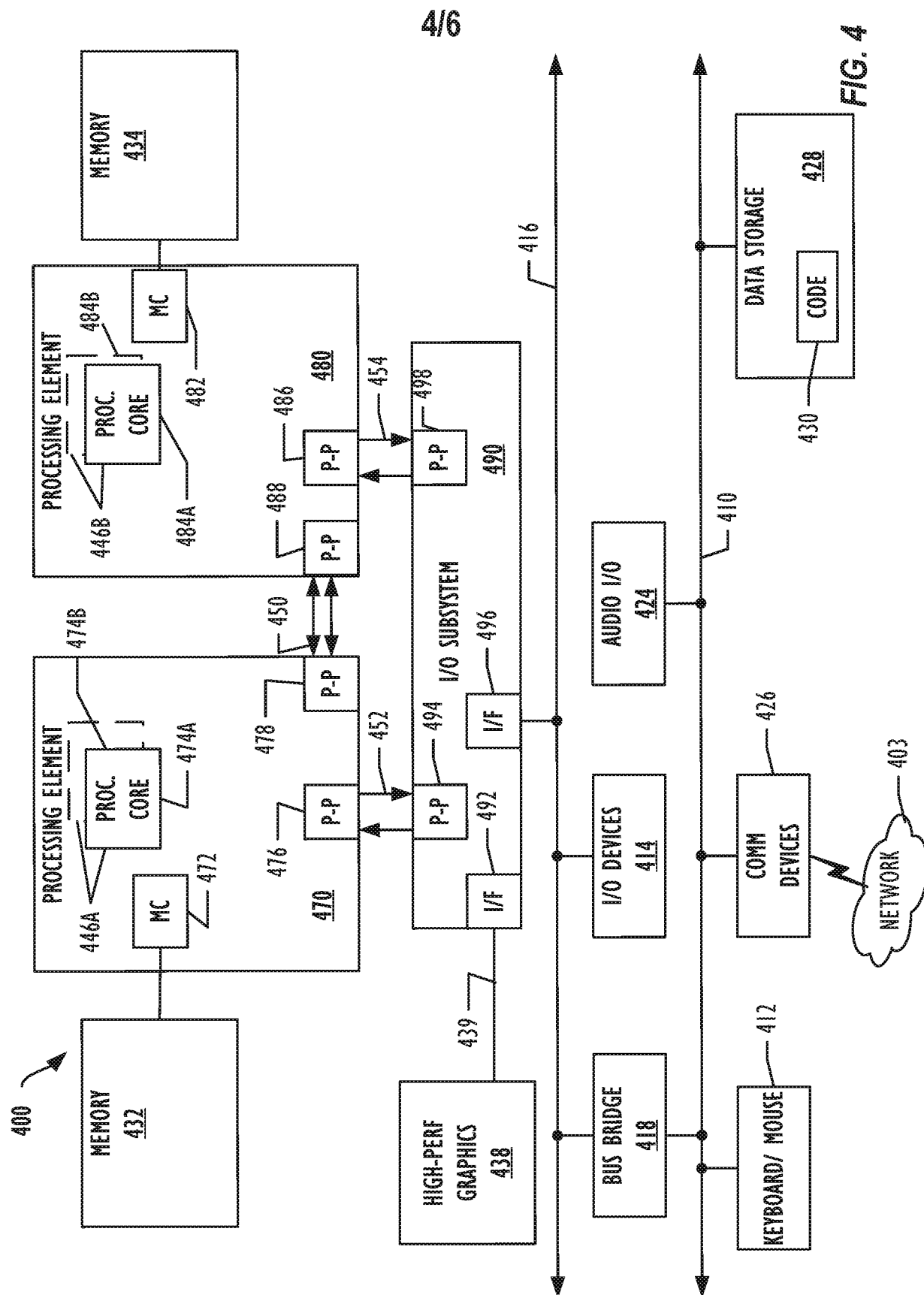


FIG. 4

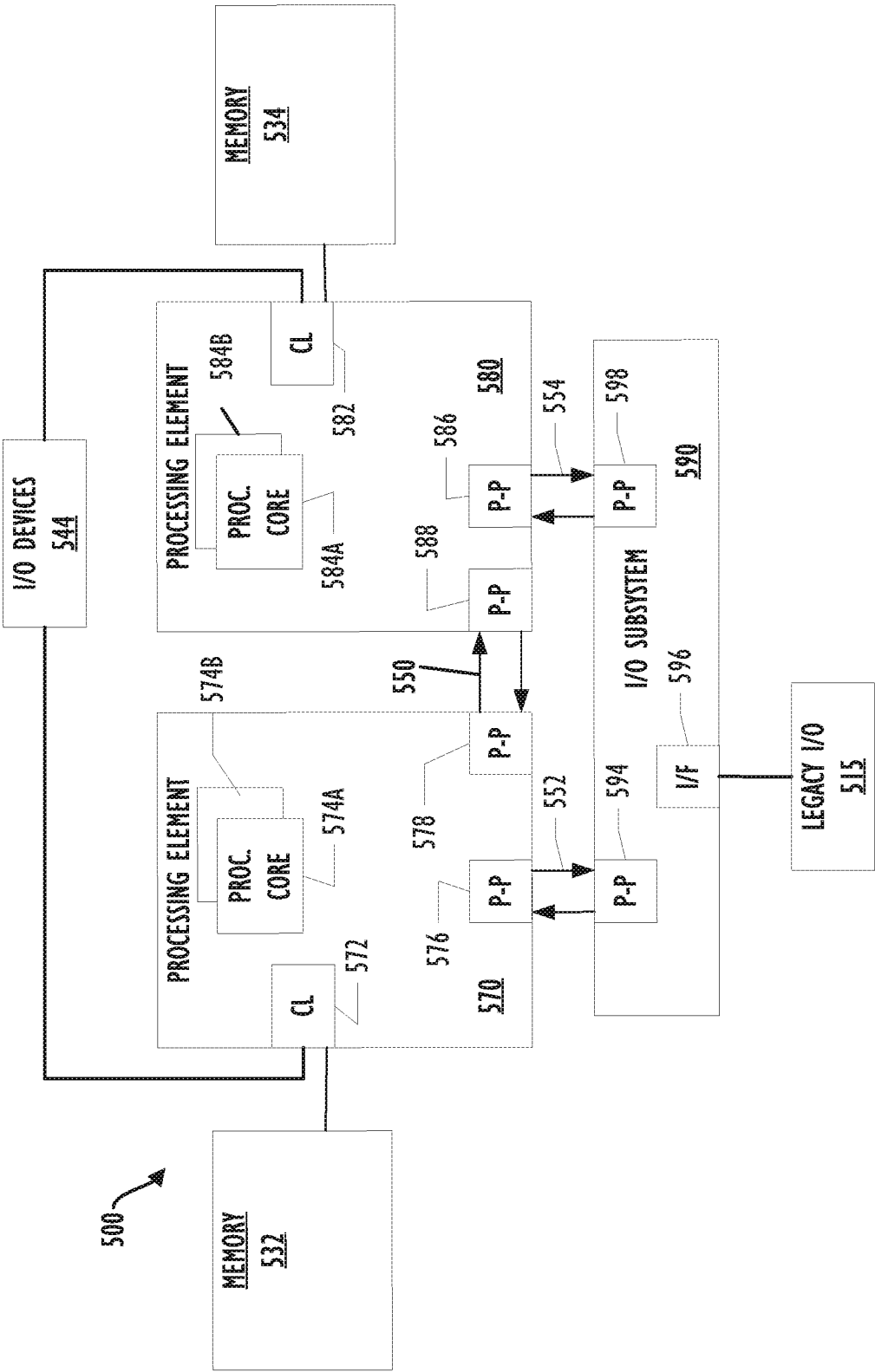


FIG. 5

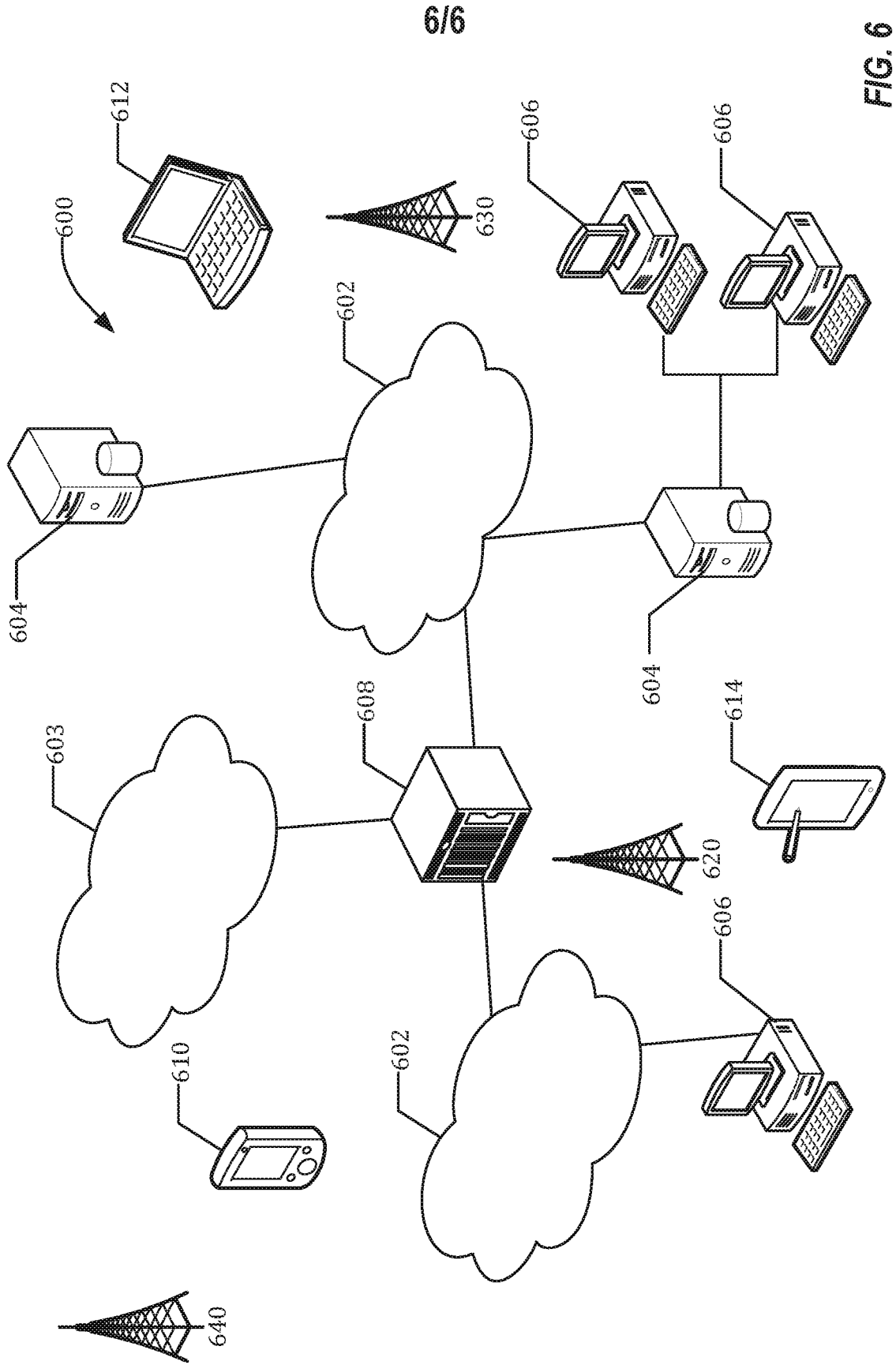


FIG. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2016/053164**A. CLASSIFICATION OF SUBJECT MATTER****H04L 29/08(2006.01)i, H04L 9/32(2006.01)i, G06Q 50/10(2012.01)i**

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L 29/08; G06F 17/24; G06F 17/00; G06F 17/30; G06F 17/21; G06F 17/27; H04L 9/32; G06Q 50/10

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) & Keywords: crowd-sourcing, validate, webpage, field, form, location

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2014-0380141 A1 (JASON CRAVEN et al.) 25 December 2014 See paragraphs [0049]-[0050], [0090], [0098]-[0099], claim 21 and figures 1, 10.	1-25
Y	US 2014-0173405 A1 (GOOGLE INC.) 19 June 2014 See claims 8-9 and figures 6-7.	1-25
Y	US 2012-0166464 A1 (JUKKA HONKOLA et al.) 28 June 2012 See paragraph [0054] and figure 4.	3, 5, 12, 14, 21, 23
Y	US 2008-0184100 A1 (ROY A. SELIG) 31 July 2008 See paragraphs [0035], [0040]-[0043] and figure 2.	4, 6-7, 13, 15-16, 22, 24
Y	US 8682674 B1 (GLENBROOK NETWORKS) 25 March 2014 See column 9, lines 58-63 and figure 3.	8-9, 17-18, 25
A	US 2008-0313529 A1 (DANIEL GWOZDZ et al.) 18 December 2008 See paragraphs [0025]-[0034], claims 1-7 and figures 3-4b.	1-25



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

21 December 2016 (21.12.2016)

Date of mailing of the international search report

27 December 2016 (27.12.2016)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

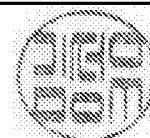
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

KIM, Seong Woo

Telephone No. +82-42-481-3348



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/053164

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2014-0380141 A1	25/12/2014	US 2014-0372860 A1 WO 2014-0153156 A1	18/12/2014 25/09/2014
US 2014-0173405 A1	19/06/2014	WO 2014-100145 A1	26/06/2014
US 2012-0166464 A1	28/06/2012	CN 103299300 A EP 2659348 A1 WO 2012-089898 A1	11/09/2013 06/11/2013 05/07/2012
US 2008-0184100 A1	31/07/2008	US 2008-0184102 A1 US 2008-0184141 A1 US 2008-0184148 A1 US 2008-0184157 A1 US 2008-0184158 A1 US 2008-0184159 A1 US 2009-0024953 A1 US 8209622 B2 US 8245151 B2 US 8255824 B2 US 8370766 B2 US 8527881 B2	31/07/2008 31/07/2008 31/07/2008 31/07/2008 31/07/2008 31/07/2008 22/01/2009 26/06/2012 14/08/2012 28/08/2012 05/02/2013 03/09/2013
US 8682674 B1	25/03/2014	US 7454430 B1 US 7756807 B1 US 8244661 B1 US 8423495 B1 US 8620848 B1	18/11/2008 13/07/2010 14/08/2012 16/04/2013 31/12/2013
US 2008-0313529 A1	18/12/2008	US 9098481 B2	04/08/2015