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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
23 December 2015 (23.12.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/195671 A1

(51) International Patent Classification:
H04W 4/02 (2009.01)

(21) International Application Number:
PCT/US2015/036039

(22) International Filing Date:
16 June 2015 (16.06.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/012,641 16 June 2014 (16.06.2014) US

(72) Inventor; and
(71) Applicant : **BAKER, Richard, L.** [US/US]; 1809 Meeting St., #1212, Lexington, KY 40509 (US).

(74) Agents: **TORRE, Patrick, M.** et al.; KING & SCHICKLI, PLLC, 247 N. Broadway, Lexington, KY 40507 (US).

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

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

Published:

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

(54) Title: DYNAMIC MOBILE PLATFORM FUNCTIONALITIES EMPLOYING PROXIMAL VARIANTS AND ADVANCED PERSONALIZATION METHODS FOR STRUCTURE, NAVIGATION, THEME, CONTENT, AND FUNCTIONALITY

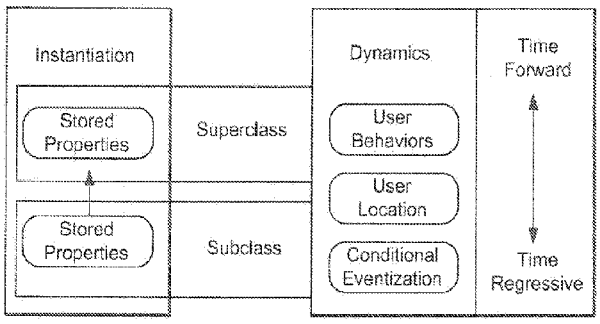

FIG. 1

(57) Abstract: A method for selecting most relevant information for display on a display screen of a mobile computing device includes comparing a the listing of past user activities and preferences to a stored informational content according to a detected proximity to a geographical location, and selecting a most relevant content of the informational content according to the comparing. The most relevant content is made available as a configured application. The most relevant content may be selected according to additional parameters, including a stored predetermined activity category, a stored geographical location of a creator of the informational content, and a time period at which the mobile device is determined to be proximal to the geographical location. The most relevant content may be displayed on the mobile device as an icon configured according to one or both of the mobile computing device geographical location and the content of the configured application.

WO 2015/195671 A1

DYNAMIC MOBILE PLATFORM FUNCTIONALITIES EMPLOYING PROXIMAL VARIANTS, EVENTIZATION AND ADVANCED PERSONALIZATION METHODS FOR STRUCTURE, NAVIGATION, THEME, CONTENT, AND FUNCTIONALITY

[0001] This application claims the benefit of U.S. Provisional Patent Application Serial No. 62/012,641 filed on June 16, 2014, the disclosure of which is incorporated herein by reference in its entirety.

TECHNICAL FIELD OF THE INVENTION

[0002] Generally, the present invention relates to computing devices and environments involving mobile computing devices. Particularly, although not exclusively, it relates to computer-implemented methods for controlling information display on a mobile computing device display screen using measures of user proximity and content relevance. Other embodiments contemplate computing systems, to name a few.

BACKGROUND OF THE INVENTION

[0003] On the web, search engines simplify locating millions of websites and apps (as used herein, “app” is a shorthand description for a software application). Many websites dynamically customize the home screen of a particular user based on user location and on the characteristics (interests, social category, context, etc.) of an individual user. This type of personalization is founded upon the premise that these changes are based on implicit data, such as items purchased or pages viewed. The term customization is used instead when the site only uses explicit data such as ratings or preferences.

[0004] Three essential categories of personalization are known:

1. Profile / Group based
2. Behavior based (also known as Wisdom of the Crowds)
3. Collaboration based

[0005] Conventional web personalization models include rules-based filtering, or “if-then” statements, based on "if this, then that" rules processing, and collaborative filtering, which serves

relevant material to customers by combining their own personal preferences with the preferences of like-minded others.

[0006] Three broadly used methods of web personalization are known:

1. Implicit
2. Explicit
3. Hybrid

[0007] With implicit personalization the personalization is performed by the web page (or information system) based on the different categories mentioned above. With explicit personalization, the web page (or information system) is changed by the user using the features provided by the system. Hybrid personalization combines the above two approaches to leverage the best of both worlds. Many companies offer services for web recommendation and email recommendation that are based on personalization or anonymously collected user behaviors.

[0008] In statistics, Bayesian inference (or probability) is a method of inference in which Bayes Rule is used to update the probability estimate for a hypothesis as additional evidence is acquired. Bayesian updating is an important technique throughout statistics, and especially in mathematical statistics. For some cases, exhibiting a Bayesian derivation for a statistical method automatically ensures that the method works as well as any competing method.

[0009] In turn, the skilled artisan is familiar with content described as “dynamic,” “adaptive” or “smart” web and mobile content that is customized based on user preference, history and behavior, such as by the methods summarized above. Dynamic Content is a term for the aspects of a website, mobile screen, ad, or email body that change based on the interests or past behavior of the viewer. It creates an experience that's customized specifically for the visitor or reader at that moment. One of the most well-known examples of smart content is the recommendation engines utilized by various online retailers. Other forms, however, range from personalization fields in emails to entire images or offers on a webpage that change based on who is looking at them.

[0010] The key to dynamic content’s effectiveness is its relevancy. Research has repeatedly demonstrated that marketing that is targeted and more relevant to the end-recipient tends to see

better results. Relevancy is born from information. Not just demographic and contact information, but saved insights about the materials that have *mattered* to a particular customer across their relationship with a company. That data then fuels the technology and set of rules that assigns the right content to the right person at the right place at the right time. Structures and methods vary but technology involved commonly includes: 1) Centralized Database — information repository on a network or cloud based; 2) Centralized Application Layer — that holds the content and application rules; 3) Mobile, network or cloud service — including location services; 4) Viewing device — computer, mobile device, etc.; and 5) Application and / or client — that resides on the viewing device.

[0011] Such personalization is likewise useful in the context of portable computing devices. However, such devices present unique problems of storage capacity and available area to display information. Portable communication devices, which as is known are mobile computing devices, are typically capable of supporting wireless communication. Typical examples of portable communication devices include, although are not limited to, mobile telephones, cellular phones, wireless-enabled tablet computers, “smart” phones, laptop computing devices, personal digital assistants ("PDA’s") and other such similar devices. Currently, portable communication devices including smart phones utilize a wide variety of different operating systems depending on the manufacturer to execute different functions. Most of these devices have the ability to determine proximity, either via GPS, or the more recently released BLUETOOTH Low Energy (BLE) technologies. A major predominant problem on all mobile devices is the explosion in the number of apps available on various operating systems and platforms, and conversely, the very limited space available on the home screens of small and medium sized mobile devices. As previously stated, there are millions of apps and only a few inches of viable screen real estate upon which to display them. Even if the real estate were infinite, the user would encounter great difficulties locating apps visually on this extended screen real estate. The more apps installed on a particular device, the more difficult it is to find the app a user is looking for. The current solutions are, to display sets of scrolling home screens with a large number of apps on each screen, or to place apps related to a particular heading into a folder. But this is not a long term solution to the problem.

[0012] Therefore, for mobile devices a need exists in the art for better methods for searching and display of information. Any improvements along such lines should also contemplate good engineering practices, such as simplicity, ease of implementation, unobtrusiveness, stability, etc. Advantageously, by the herein described methods and systems, location and personalization tools are used not only to determine content displayed on a mobile computing device, but also content display (app) structure, theme, functionality, and others.

SUMMARY OF THE INVENTION

[0013] By applying the principles and teachings described herein, the foregoing and other problems become solved. Using the described methods and systems, in the small amount of display screen real estate available to a mobile computing device —for example, the area of a single app icon—many dozens, or even thousands of pages of informational content, can be more easily accessed by a user based not only on proximity to a geographical location, but also based on settings, preferences and predictive algorithms analyzing user patterns designed to configure an app (interchangeably referred to in this application as a Dynamic Mobile Application Framework or OMVERIA app), the most useful to a particular user, in a particular location, at a particular time. The OMVERIA icon is like a chameleon. It changes not only content but also appearance wherever the user goes based on location and other factors.

[0014] In one aspect, a method for selecting most relevant information for display on a display screen of a mobile computing device is described. The method includes providing a user mobile computing device having at least one processor and at least one memory, and also providing a remote computing system comprising one or more computing devices and also non-transitory storage including at least a stored database comprising a listing of past user activities and preferences and a stored database comprising informational content relevant to a plurality of geographical locations. The remote computing system may include one or more of a server hosted in a cloud computing environment, one or more computing devices hosting an app database; one or more administrative computing devices hosting an administrative database, a router or network wi-fi transmitter, and combinations thereof. The remote computing system is also configured at least to determine a proximity of the mobile computing device to a geographical location of the plurality of geographical locations. A stored unique identifier of the

mobile computing device may be communicated to the remote computing system. The method in accordance with the present disclosure is implementable on, and is compatible with, any portable communication device that supports wireless communication such as BLUETOOTH or WLAN technology, and is in operable connection with wireless communication networks, or BLUETOOTH stations, WLAN stations etc. Furthermore, the disclosure is not limited merely to only smart phones, but works equally well with other portable communication devices/mobile computing devices as summarized above.

[0015] In embodiments, the remote computing system is configured to compare the listing of past user activities and preferences to the informational content according to the unique identifier and detected proximity, and to select a most relevant content of the informational content according to the comparing. The remote computing system is also configured to make available the selected most relevant content to the mobile computing device as a configured mobile computing device-readable application. In embodiments, the configured application is made available in a format compatible with a determined operating system of the mobile computing device. In embodiments, the step of making available the selected most relevant content to the mobile computing device comprises displaying an icon operatively connected to the configured application on a display screen of the mobile computing device.

[0016] The method may further include steps of selecting, by the remote computing system, most relevant content by factoring other parameters. In embodiments, the other parameters may include determining or refining a most relevant content according to a stored predetermined activity category, to a stored geographical location of a creator of the informational content, and/or to a determined date and/or time at which the mobile computing device is determined to be proximal to the geographical location.

[0017] In another aspect, a computing network is described for performing the above-described method for displaying most relevant information on a display screen of a mobile computing device. As described, the network includes at least a mobile computing device having at least one processor and one memory and a display screen and configured for communicating a unique identifier of the mobile computing device to a remote computing system and a remote computing system comprising one or more computing devices each having at least one processor

and at least one memory, the remote computing system further comprising non-transitory storage including at least a stored database comprising a listing of past user activities and preferences and a stored database comprising informational content relevant to a plurality of geographical locations. As described, the remote computing system is configured at least to determine a proximity of the mobile computing device to a geographical location of the plurality of geographical locations, to compare the listing of past user activities and preferences to the plurality of applications according to said unique identifier and said detected proximity, to select a most relevant content of the informational content according to the comparing, and to make available the selected most relevant content of the informational content to the mobile computing device as a configured mobile computing device-readable application.

[0018] On a mobile device, these icons generally correspond to mobile application programs, Web sites (aka "web apps"), and others. The icons displayed by OMVERIA can be retrieved from a cloud database and service that identifies a particular app, and its corresponding icon, as relating to a particular location but also to certain user behaviors/preferences, and still further to particular temporal metrics. So this method is focused on the art of using sensors to confirm location of a user, then automatically searching to identify apps and their icons providing a predicted most relevant content to the user, for example apps registered in a cloud service. In the cloud service database these icons will have been associated with the most relevant content. The described OMVERIA method dynamically retrieves and presents the associated app icon, making it easier for the user to access and use a more relevant app to their location without having to search for that app on their various home screens or an "app store" provided by their particular device OS.

[0019] Essentially, the concept is; "Why search when you are already there?" When we consider the location based technologies and sensors available to confirm location, and the sophisticated methods of attributing behavior to action, it makes sense that mobile search should be fundamentally changed. From a desktop computer, we are essentially searching the world wide web to bring information back to us at our fixed location. But now just the opposite is true. In this age of mobility where mobile search outweighs desktop search, searching strategies should be reversed, that is, searching should be automatic and should be pre-filtered based on a user's location, preferences and condition. Users want to find locations, but it is also true that

locations (businesses) need to find users, who are moving through geographical space, and so delivered content must be contextual to be more useful. So the presently described methods provide a system that makes both function and content dynamic to a context, such as location. In addition, we must move beyond touch, to intuition, allowing the algorithms to do the work for us, while optimizing space on the mobile device.

[0020] These and other embodiments of the present invention will be set forth in the description which follows, and in part will become apparent to those of ordinary skill in the art by reference to the following description of the invention and referenced drawings or by practice of the invention. The claims, however, indicate the particularities of the invention.

BRIEF DESCRIPTION OF THE DRAWINGS

[0021] The accompanying drawings incorporated in and forming a part of the specification, illustrate several aspects of the present invention, and together with the description serve to explain the principles of the invention. In the drawings:

[0022] Figure 1 schematically illustrates a representative programming structure for implementing a dynamic mobile computing framework according to the present disclosure;

[0023] Figure 2 illustrates a determination of distance detection for a mobile computing device using BLUETOOTH Low Energy (BLE) beacons;

[0024] Figure 3 illustrates a system for providing a dynamic icon on a mobile computing device according to proximity to a content provider according to the present disclosure;

[0025] Figure 4 illustrates the system for providing a dynamic icon on a mobile computing device by a push notification according to proximity to a specific content provider such as a retail store as shown in Figure 2;

[0026] Figure 5 illustrates a representative content/informational flow through the dynamic mobile computing framework according to the present disclosure;

[0027] Figure 6 illustrates a representative computing architecture for a dynamic mobile application framework to provide the content/informational flow according to Figure 5; and

[0028] Figure 7 illustrates a representative hive structure defined by the dynamic mobile application framework.

[0029] Reference will now be made in detail to the present preferred embodiments of the described dynamic mobile computing framework, examples of which are illustrated in the accompanying drawing figures.

DETAILED DESCRIPTION OF THE ILLUSTRATED EMBODIMENTS

[0030] In the following detailed description of the illustrated embodiments, reference is made to the accompanying drawings that form a part hereof, and in which is shown by way of illustration, specific embodiments in which the invention may be practiced. These embodiments are described in sufficient detail to enable those skilled in the art to practice the invention and like numerals represent like details in the various figures. Also, it is to be understood that other embodiments may be utilized and that process, mechanical, electrical, arrangement, software and/or other changes may be made without departing from the scope of the present invention.

[0031] The high level idea underpinning the presently described technology is application of situational awareness to provide a goal-directed task analysis (GDTA) in order to select and present most relevant information from a pool of informational content to a mobile device user. As is known, situational awareness is a field of research within cognitive science defined as "the perception of elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future." Stated differently, situational awareness is knowing what is going on around a user and what is most important to the user's current goals. By the systems and methods described infra, a particular class of user can draw from significant quantities of content and data in order to quickly comprehend the information and arrive at a suitable decision. As a non-limiting example, consider a class of users (travelers) whose goal is to search for and reserve a hotel room for the night. Particular decisions to be made might include: 1) is the hotel in a good, safe location? 2) do the hotel amenities meet my needs? and 3) can I afford the hotel? User task flow analysis is

nothing new, however using GDTA can help us display the right content at the right time for the right situational awareness. GDTA methods combine content and data to help users comprehend quickly–and then make a decision.

[0032] Particular metrics and informational content can be considered in a goal-directed framework such as the presently described dynamic mobile framework, including location, recent crime statistics, and closeness in time to other trip-related events to determine an answer to the question "is this hotel in a good location." The ultimate question is how to turn this simple data flow into awareness and generate predictive analytics to anticipate and provide the right functionality and content on a mobile device at the right time for a particular user. The presently described dynamic mobile framework provides a different approach to data, analytics and the application of Bayesian inference on behalf of the mobile user, utilizing protocol-oriented programming and first class value semantics for benefits in predictability, performance, and productivity.

[0033] For the presently described methods and systems, program design constructs were built using Classes, Structures, and Protocols to create core Object-Oriented programs. Classes and structures are general-purpose, flexible constructs that become the building blocks of OMVERIA and its sense of app awareness and app inference. Classes and Structures can both: 1) define properties to store values; 2) define methods to provide functionality; 3) define initializers to set up an initial state; and 4) be extended to expand functionality beyond a default implementation. These methods are a model of Object Oriented (OO) Programming Language. We can model problems using objects which can encapsulate data and functionality, and build complex relationships between these objects in a modular fashion.

[0034] Classes provide functionality for two important pieces of OO functionality: 1) Inheritance, where one class can inherit the characteristics of another; and 2) Type casting, where you can check and interpret the type of a class instance at runtime of an app. On a mobile platform this kind of thinking is useful because the method helps manage application memory which optimizes performance, an important criteria when creating complex predictive behaviors. Different memory semantics is not the only advantage of having these distinct types. Because static structures are simpler than classes, and cannot be as heavily modified after declaration,

they provide an opportunity to create value objects which represent pieces of data independent from their behavior.

[0035] Protocols are a valuable addition to classes and structures. They provide a way to define behavior separately from the classes which implement them. A protocol defines a blueprint of methods, properties, and other requirements that suit a particular task or piece of functionality. Protocols embrace flexible design by encapsulating the necessary data and behavior for a domain idea outside the scope of a Class or Structure definition. This means that a concept can be represented separately from its implementation, allowing for more creative reuse, composition, and inclusion of dynamic data from an internal or external source.

[0036] Other concepts are relevant to the described dynamic mobile framework. In the context of the described framework, instantiation is the creation of a real instance or particular realization of an abstraction or template such as a class of objects. To instantiate is to create such an instance by, for example, defining one particular variation of object within a class, giving it a name, and locating it in some physical place within the application code.

[0037] Eventization is a variant of location that can be related to a particular User's behavior patterns to create Inference. For example, the location of an entertainment venue is known and very likely will not change. On one night, the venue holds a sporting event. On another night a music concert is held. Expressed in Superclass and Subclass terminology, Eventization applies time-contextual Subclasses (types of events which may vary) to a Superclass (Location which will remain fixed). Stated differently, Location alone may not determine relevance. Eventization represents a time variable layered on a location, i.e. the location parameter further refined by the time parameter, or the eventization of the Superclass, provides relevant information of interest to a user – a restaurant location (Superclass) by itself may be relevant information to a decision-making process from Monday to Saturday, but on Sunday the restaurant is closed and so the restaurant location is not relevant to the query "where is a good place to eat," and/or information about the restaurant location should not be pushed to a user on Sunday. Likewise, information regarding the Subclass may be relevant in the future (for example, "the restaurant is having a special next Thursday"), and so that information could be pushed to the user.

[0038] Another variant of Eventization is a time-contextual variable used to determine an event's potential relevance at a particular location based on applying practical limits to the User's available time to create a "true" or "false" related to Inference or Suggestion. For example, if a User's available time while visiting a particular city is limited, Eventization would infer relevance of a particular event occurring during the time period that the user is visiting the city ("true"). The information of the event would be determined to be relevant to decision-making and would be pushed to the user. On the other hand, a time available Subclass would change this Inference to "false" if the event was occurring outside of the time period that the user was visiting, the information of the event would be determined not to be relevant, and the suggestion would not be delivered to the user.

[0039] In one embodiment, the described dynamic mobile framework combines protocol programming to Instantiate stored properties using a Superclass and Subclass structure within these Object Oriented programming principles. We apply Time Regressive variants to create Map Reduce constructs based on all data available. These Time Regressive Map Reduce constructs are used to make Inferred suggestions for known overlaps between future events of the user and future events of nearby venues, to create suggestions. Each Superclass is modified with a Subclass of dynamic variants, i.e. Superclass Location, Subclass Eventization, etc. Each Superclass and Subclass can be cross-referenced using time forward and time regressive data for a particular user to build personalization models.

[0040] Another advantage of the described dynamic mobile framework is information sharing between apps. Conventional mobile operating systems do not allow one app to access data from another app, and the mobile OS shares very little, if any, of the data about a particular user on the OS. However, since the present dynamic mobile framework acts like a colony of apps that are presented dynamically under certain conditions, certain behavioral data is received, stored and analyzed by the OMVERIA app framework and can be used to build awareness of the user's preferences and use predictive methods to present the right app or function at the right moment for a particular user.

[0041] This is depicted schematically in Figure 1. Taking each of the three dynamic variants on the right: User Behaviors, User Location and Conditional Eventization — and running them

backwards in time to the beginning of our association with a particular user, we can begin to build patterns of movement and behaviors. We can then compare these two variants to Eventization information such as: the User visited a particular Venue and attended a concert by Artist A at a particular date. This results in a stored database including essentially a listing of past user activities and preferences.

[0042] From this information, running this pattern forward in time looking for time forward Eventization patterns that are similar, Bayesian Inference yields the following: If a Venue calendar indicates Artist A will be playing at the same venue or a nearby at a future date, we could provide the following notice to the User: "Would you like add this event to your calendar, or purchase tickets?" These types of notices can likewise be used to present the most relevant application for a given User's current location, or can suggest the most relevant OMNIA app based on these conditions. Expressed in Bayesian math the formula for adding probability to a particular Subclass variable of a Superclass could be expressed as follows:

[0043]

$$P(h|d) = \frac{P(d|h)P(h)}{P(d)},$$

[0044] In another example, a particular Subclass variable of a Superclass could be defined as follows, again using Bayesian mathematics:

[0045] A = P(A/E, L, CT, TR) = [P(A/E, L, CT) * P(A/T)]/P(A), where A= the display provided by the described system and app, E = a variable representing an eventization string, CT = a variable representing the current moment in time, TR = stored data from a past moment in time, L = a variable representing a location, and P = a calculated probability of relevancy of information. Thus, a calculated probability based on past events is then used to define a subclass which can then be stored and applied as predictive of future behaviors/events, and which will evolve in the future.

[0046] For the OMVERIA system and app to be effective in adding user behavioral data in determining which app configuration to display, we first need to identify probabilistic

relationships of business locations visited behavior, and especially to understand the probabilistic relationship between business types visited and time of day and related factors. Several analytical methods may be utilized in achieving a more targeted result. In particular embodiments, the algorithms are based on Bayesian Networks, and can include: 1) Visual analysis of Bayesian Networks to find initially interesting patterns, variables and their relationships, 2) user segmentation analysis, 3) node force analysis and 4) a combination of expert-based service clustering and machine learning for usage diversity vs. intensity analysis. All the analyses will involve handset–based data collected from the OMVERIA app. The accuracy of our predictions increases when the number of users increases and as each user increases the frequency of use of the app. In addition, probabilistic relationships can be found within certain business types cluster pairs in their diversity and intensity values. Based on these relationships, similar mediation type of behavior can be found for the kinds of places a person visits as and when and where this happens. As is known, a Bayesian Network is a straightforward way to express model data on a high level. Moreover, Node Force, Direct and Total effect are useful metrics to measure the mediation effects. The clustering implemented as a hybrid of machine learning and expert-based clustering process is also a useful way to calculate relationships between clusters of more than a hundred individual users.

[0047] Handset-based measurements are a data collection method utilizing smartphones' ability to respond to the OMVERIA application software. These measurements are implemented by installing a data collection application to the mobile phones of opt-in participants and by collecting data in the cloud. With these measurements rich contextual user level data on business locations visited can be collected. Handset-based measurements have increasingly been used for a number of purposes in the recent years, applications ranging from sociology to consumer behavior. Business locations visited in particular will be an important data set.

[0048] A Bayesian Network (BN) (also called Bayes Belief Network (BBN)) method can be used for analyzing handset-based data from a holistic business locations visited perspective. This method may use BN to find business types and to cluster locations, using handset-based measurement data and GPS/Bluetooth low energy sensors or other location services. BN is an analytical method that we can use for inferential analysis, e.g., to make "what if" simulations, to predict behavior patterns and future trends, to understand why something most probably

happened, and to understand which data correlate with other data. It is challenging to analyze the relationships between business locations visited patterns as the number of possible places to visit is very high in the used dataset. Although a BN procedure will offer easier methods to study this data the results could be further qualified against other methods like Regression analysis or Neural Networks.

[0049] A BN can be created in three ways, namely manually by using expert knowledge, by using machine learning, or a combination of them. As said, a BN is used for inferential analysis (often called predictive analytics), e.g., to predict behavior patterns and future trends, to make "what if" simulations, to understand why something most probably happened and to understand which data correlate with other data. In the present disclosure, user behavioral data analyzed by a Bayesian network in combination with ascertained location data — either stored in the cloud or in memory in the mobile device — allows a precise determination of which app icon configuration to display in the OMVERIA app at a particular time for a particular user.

[0050] Certain other known technologies utilized in the present methods and systems merit discussion. Many service discovery protocols are available in the mobile communication technology, which allow mobile devices to detect a user's location and to connect that user to businesses or other sites of interest located in their proximity. Global Positioning Satellite (GPS) technology has long been included in smart phone technology and works reliably outdoors. But tall buildings and other obstructions make triangulation difficult. And GPS has even less usefulness indoors. Recently, BLUETOOTH Beacons, which enable devices to accurately triangulate locations with pinpoint accuracy, have achieved more widespread use. A related disclosure employing, among other features, location services relying on BLUETOOTH beacons to provide location services for determining content for display on a mobile computing device is U.S. Patent Appl. S.N. 14/579,637 for Mobile Platform Functionalities Employing Proximal Variants and Advanced Personalization Methods to Control Dynamic Icon Display on a Mobile Computing Device Display Screen filed on December 22, 2014, the entire disclosure of which is incorporated herein by reference as if fully restated. However, it will be appreciated that the system can use other methods of determining location by using other sensors like RFID tags, photo matching, sound wave reception, UF signals from routers or other broadcast devices, and others.

[0051] Portable communication devices, including mobile devices, cellular phones, smartphones, personal laptop computers, tablets, personal digital assistants (PDA), etc. are predominantly used in the art, for communication and other purposes, and such devices are often equipped with the feature of supporting wireless communication, including the Wireless Local Area Networks (WLAN) and BLUETOOTH technology etc., through suitable applications/modules installed within the devices. BLUETOOTH technology facilitates short range wireless communication between such devices. Using short wavelength radio transmission, the BLUETOOTH technology enables voice and data exchange between the devices and between the devices and BLE Beacons. To support the Blue-tooth technology, the communication devices, including mobile devices, generally have a Radio Frequency Blue-tooth Transceiver that lies at their physical layer, and an adapter which may be in-built, or can be in the form of a card that connects to the device.

[0052] BLUETOOTH is a wireless technology standard for exchanging data over short distances (using short-wavelength radio waves in the ISM (Industrial, Scientific, Medical) band from 2.4 to 2.485 GHz) from fixed and mobile devices, building personal area networks (PANs). BLUETOOTH is managed by the BLUETOOTH Special Interest Group aka, “the SIG”), which has more than 19,000 member companies in the areas of telecommunication, computing, networking, and consumer electronics. BLUETOOTH was standardized as IEEE 802.15.1, but the standard is no longer maintained. The SIG oversees the development of the spec for BLUETOOTH and BLE (BLUETOOTH Low Energy), manages the qualification program, and protects the trademarks. To be marketed as a BLUETOOTH device, it must be qualified to standards defined by the SIG.

[0053] BLUETOOTH low energy (BLE) is a wireless area network technology designed and marketed by the nonprofit, nonstock corporation BLUETOOTH Special Interest Group aimed at novel applications in the healthcare, fitness, security, and home entertainment industries. Compared to "Classic" BLUETOOTH, BLE is intended to provide considerably reduced power consumption and cost while maintaining a similar broadcast range of about 20 meters. BLE was merged into the main BLUETOOTH standard in 2010 with the adoption of the BLUETOOTH Core Specification Version 4.0. Many modern mobile operating systems natively support BLE.

The BLUETOOTH SIG predicts more than 90 percent of BLUETOOTH-enabled smartphones will support the low energy standard by 2018.

[0054] In a BLE Beacon system, Received Signal Strength Indication (RSSI) is the relative received signal strength in a wireless environment, in arbitrary units. RSSI is an indication of the power level being received by the antenna. Therefore, the higher the RSSI number, the stronger the signal. To date, all BLE beacons are omnidirectional (broadcasting in a 360 degree pattern) in nature and the primary method used to make BLE beacons commercially useful is the RSSI method. In general, the greater the distance between the device and the beacon, the lesser the strength of the received signal. This is illustrated in Figure 2, showing a BLE beacon 10 determining various proximity indices ("immediate," "near," "far," "unknown") according to a detected distance from a BLUETOOTH-equipped smartphone 12.

[0055] This inverse relation between the distance and RSSI is used to estimate the approximate distance between the device and the beacon using another value generally referred to as Measured Power. Measured Power is a factory-calibrated, read-only constant which indicates the expected RSSI at a distance of 1 meter to the beacon. Combined with RSSI, this allows a method of estimating the actual distance between the device and the beacon.

[0056] Note that, due to external factors which influence the BLUETOOTH radio wave broadcasted by beacons—such as absorption, interference or diffraction—the RSSI value tends to fluctuate. The further away the device is from the beacon, the more unstable the RSSI readings will be. And, since distance approximation is based on RSSI, this directly translates to less accurate estimates at greater distances. By this method, when a sending device and a receiving device (such as a smartphone carried by a user) are within range of each other distance can be determined with some degree of accuracy.

[0057] Broadcasting Power is the power with which the beacon broadcasts its signal, i.e. the power with which the signal leaves the beacon's antenna. These Broadcast Power settings can be varied. The value ranges between -30 dBm and +4 dBm, lowest to highest power settings respectively. The higher the power, the bigger the beacon's range and the more stable the signal, but if the beacon is battery powered, high power may shorten the battery life.

[0058] Due to constraints imposed by original equipment manufacturers, there is typically an incompatibility factor between the different mobile device operating systems, which obstructs these communication devices from detecting each other, when mutually coupled in a network. For example, a smartphone running a proprietary operating system of a first manufacturer can only detect another smartphone of that manufacturer, and no other device, within its near-field wireless communication network. A smartphone running a proprietary operating system of a different manufacturer will not be detected.

[0059] With reference to Figure 3, the mobile device 10 is situated between and in range of 2 different BLE beacons 10, 10'. As will be described, the presently described methods and systems apply rules based on movement, signal strength and other factors to determine which icon 20, 20' is most relevant to the user and so which to display. The selected icon is operatively connected to informational content such as an application, web page, service, etc. of relevance to the user based on the applied rules. The application, web page, service, etc. may be hosted on a remote computing system which may be maintained in a cloud computing environment, depicted nebulously in Figure 3 as cloud service 22.

[0060] Figure 4 depicts schematically the various determinations of proximity for mobile computing devices (not shown in this figure, but see Figure 2), a service infrastructure such as an application server hosted in a cloud computing environment (nebulously, ref. num. 30) and/or a third party app store 32 from which apps and/or web pages may be retrieved according to measures of relevance including proximity ("immediate," "near," etc.) and user-based criteria. As yet another embodiment, the icons may be displayed on a map graphic 34 indicating locations proximal to the mobile computing device position.

[0061] As explained above, the presently described methods/systems do not rely exclusively on location services, but rather also incorporate determining most relevant content of a database of informational content according to certain user behavior based metrics, including without intending any limitation user past preferences and activities, user context, and others. While it is possible to individually search a particular mobile device OS for apps created by third party application developers, as described at a high level above OMVERIA is an integrated, context-aware mobile framework. Therefore, in embodiments the presently described methods and

systems are powered by a cloud-based publishing system that allows a virtually infinite number of users to publish apps stored within and retrieved from a framework referred to variously as the OMVERIA framework and a dynamic mobile application framework. The cloud based publishing system allows users to create and style content online. Application Program Interfaces (API's) are created by the system and the OMVERIA mobile apps on a mobile device OS respond to the system API's to call functionality within the app and to push new content updates to the app. Using the concepts presented the system learns user behaviors and presents apps when they need them. Rather than asking mobile users to download more apps — users such as businesses can seamlessly detect, connect and communicate with mobile users *in context* through the described dynamic mobile app framework.

[0062] At a high level, the dynamic mobile app framework is built on the following integrated technologies: 1) A cloud-based CMS that businesses can use to design and deliver HTML and multimedia content to consumer's mobile devices; 2) A rich reporting system with time, location and other reports; 3) Classic Bluetooth Low Emission and iBeacon micro-location sensors or other location services. Placed in a client's physical space, iBeacons or other locator services allow businesses to detect, connect and track mobile devices in their physical space and to deliver contextual content; and 4) a dynamic app (the OMVERIA app) that delivers app content and functionality for millions of businesses from a single mobile application icon.

[0063] Figure 5 shows a representative content/informational flow through the dynamic mobile computing framework. At step 50, a user such as a business or a venue hosting a number of businesses designs an app using a suitable editor, for example a WYSIWYG editor. The app may include as much or as little and as varied information as desired, for example a "bare bones" app providing only the business' address, phone number, and web site. Alternatively, the app may be content and information rich, including the above information plus sale offers, sale information, coupons, loyalty reward programs, etc. In turn, the user may update, alter, or replace the app content and/or layout and/or design and/or appearance at will using the editor.

[0064] At step 52, a parsing editor parses the designed app in a suitable programming language. The parsed app layout may be stored in a database 54, and updated or replaced as needed as the app evolves to include new and/or updated content. The parsing editor 52 also

provides the newly formatted (parsed) app layout to an app publishing module (step 56). At step 58, the app publishing system pushes the formatted app(s) at to user mobile computing devices 60. As described, this occurs when the mobile device is determined to be in proximity (by one or more of the location service technologies described above) to a geographical location such as the location of the business or venue. As the app evolves, including considering evolving user preferences, activities, and contextual information as described above, the app layout likewise evolves and the mobile device user sees a different layout on the device, potentially each time she passes a particular geographical location.

[0065] A representative computing architecture 62 for providing the information flow shown in Figure 5 and described above is presented in Figure 6. Included in the architecture is an online "dashboard" 64 comprising a user interface such as the WYSIWYG interface described above, whereby a user may create and manage the design (styles, themes), layout (navigation, functions), and informational content of an app as described above. The editor may be similar in form and function to known editors, for example a web-based CMS system using a conventional programming language such as javascript. The dashboard 64 may communicate via API interfaces with a parsing engine 66 which authenticates and validates submitted code and parses the app into a suitable computing language such as Ruby or other language. In the depicted embodiment the app is parsed in Ruby, in order to provide compatibility with an iOS system as manufactured by APPLE. Of course, parsing for compatibility with other OS for different types of mobile computing device is contemplated.

[0066] As shown, the dashboard 64 includes a service layer 66, a business layer 68, and a data layer 70. The parsing engine service layer 66 pushes the parsed app layout to an app publishing module 72, which in turn pushes the formatted app to a compatible mobile computing device 74, or alternatively may communicate directly with the mobile computing device 74. As described, this step of pushing a formatted app to a compatible mobile computing device 74 is conditional on one or more of a mobile device location/proximity to a geographical location, stored user behaviors, and stored user preferences such as visiting a particular venue multiple times in a short time frame, attending a function featuring a particular entertainer with a frequency that exceeds a particular threshold, or the like.

[0067] The parsing engine data layer 70 communicates with one or more databases 76 comprising, for example, business information such as business listings provided by a data puller module 78 which in turn collects such business information from multiple data providers (generically designated as 80). The business information may be formatted in an appropriate computing language for storage, such as Postgresql. Other information stored in databases 76 may include metrics of user preferences, user behavior, etc.

[0068] In another aspect, the above-described systems and methods are adaptable to a number of other uses, for example a hybrid business/social application function referred to herein as "hives" for its collective nature. At a high level, hives of informational content allow users such as businesses to form groups and present themselves both individually and in group fashion to users of the described mobile framework. A number of metrics can be employed to define particular hives. Hives may be created/defined as to membership using the online dashboard described above. A representative hive structure is shown in Figure 7, with different hives being represented by differently shaded points.

[0069] A hive can be based on a number of metrics, such as geographical location. For example, businesses in a shopping center or mall can join together and act as a hive and cross promote to users. Similarly, businesses in a particular city can form a hive. Still more, a zip code can be a hive (i.e., "show all the locations or deals in my home zip code," or "show all the locations or deals in my current zip code."

[0070] Alternatively, hives can be defined as "affinity groups," i.e. by shared common interests. For example, if a user enjoys the sport of surfing she can join a surfing hive and get content, coupons or other things related to that topic

[0071] Still more, categories of hives may be established, for example relating to broad topics such as entertainment, food, travel, shopping, cooking, etc. This is similar in principle to how conventional news feeds are selected. However, a distinguishing feature is that apps participating in the above-described dynamic mobile framework and grouped hives provide informational content to a user as a collective, rather than as individual websites, apps, or pages of information that must be separately searched.

[0072] A particular hive (or even group of hives) may be governed by rules established by a particularly influential member of the hive (for example, a large venue such as a shopping mall or an entertainment venue hosting numerous smaller businesses), or may be governed by rules established by all or substantially all members of the hive participating as equals. As examples, rules may be created that affect how deals, newsletters, ads and other content are presented within their individual member-created mobile apps stored within the mobile application framework. As users move about a city or geographical area, they are in a mode of exploration —not checked into a particular business. In this mode of exploring, tiles or other icons may be presented to users showing businesses nearby. Also navigational tabs may appear showing deals, offers, loyalty and other incentives. These offers and content types can be published by individual businesses or hive partners, co-promoting business traffic and increasing points of brand activation. Because the described dynamic mobile framework is a framework allowing mobile applications to be published and displayed from a single icon based on dynamic variables, these additional relationship constructs are possible based on the creation of rules by the businesses that chose to associate themselves together. The system functions as a collective: when a master application created by the particularly influential member of the hive is connected to partner locations, visitors get branding from the master location via the dynamic app, while also extending fully customizable app functionality to an unlimited number of partner locations that are members of the particular hive.

[0073] As a result, the foregoing scheme ensures that the user always receives the most relevant content according to a mobile computing device location, i.e. proximity to a particular content provider, and also according to predetermined user metrics. Furthermore, the protocol ensures that the most relevant content of a database of informational content is displayed within a display screen of a mobile computing device, but evolves as the user alters the geographical location of the mobile computing device (and so alters what is the most relevant content to be made accessible via the icon).

[0074] By presenting mobile apps within a proprietary mobile application framework, the described methods and systems do not require users to install and launch third party apps. Essentially the displayed apps are configured from a colony of apps stored within the framework. Each app is called up based on a determined relevance to the user. Once a user anywhere

downloads and installs the app for any particular participant such as a business, all the other businesses that have published a mobile app within the described mobile application framework are already present. Thus, the system allows the user to download once, but use the mobile framework everywhere and anywhere an OMVERIA app exists that is relevant to that location. Particularly in the context of business owners, this solves a significant problem: the average user only downloads a few apps from the millions of available apps. So small businesses miss out. Even if small businesses create and publish a mobile application using conventional methods, statistics indicate that less than 20% of these are opened more than once and most are deleted within 90 days. To solve this problem, the disclosed methods and systems use machine learning algorithms to dynamically present users with choices nearby or based on behavior, logic or other conditions. Uniquely, the described mobile framework in an embodiment presents a "colony" or "hive" of published apps with full content and function, solving a significant business problem not addressed by current technologies.

[0075] In turn, methods and apparatus of the invention further contemplate computer executable instructions, e.g., code or software, as part of computer program products on readable media, e.g., disks for insertion in a drive of computing device, or available as downloads or direct use from an upstream computing device. When described in the context of such computer program products, it is denoted that items thereof, such as modules, routines, programs, objects, components, data structures, etc., perform particular tasks or implement particular abstract data types within various structures of the computing system which cause a certain function or group of function, and such are well known in the art.

[0076] The disclosed embodiments may also include software and computer programs embodying the process steps and instructions described above. In one embodiment, the programs incorporating the process described herein can be stored as part of a computer program product and executed in one or more computers in one or more of the devices or systems. The computers can each include computer readable program code means stored on a non-transitory computer readable storage medium for carrying out and executing the process steps described herein. In one embodiment, the computer readable program code is stored in a memory. In one embodiment, one or more of the devices and systems include or are comprised of machine-readable instructions that are executable by a processor of a computing device.

[0077] The systems and devices shown in the embodiments disclosed herein are configured to utilize program storage devices embodying machine-readable program source code that is adapted to cause the devices to perform the method steps and processes disclosed herein. The program storage devices incorporating aspects of the disclosed embodiments may be devised, made and used as a component of a machine utilizing optics, magnetic properties and/or electronics to perform the procedures and methods disclosed herein. In alternate embodiments, the program storage devices may include magnetic media, such as a diskette, disk, memory stick or computer hard drive, which is readable and executable by a computer. In other alternate embodiments, the program storage devices could include optical disks, read only-memory ("ROM") floppy disks and semiconductor materials and chips.

[0078] The systems and devices may also include one or more processors or processor devices for executing stored programs, and may include a data storage or memory device on its program storage device for the storage of information and data. The computer program or software incorporating the processes and method steps incorporating aspects of the disclosed embodiments may be stored in one or more computer systems or on an otherwise conventional program storage device. For example, in one embodiment, the devices and systems, can include one or more controllers that are comprised of, or include, machine-readable instructions that are executable by a processing device. The method and the system of the present disclosure can be used for various purposes, including, though not limited to, plain device discovery, facilitating multiplayer online gaming between users of different communication devices operating through different incompatible operating systems which are generally incompatible, or to exchange data or enable short range communication between such devices.

[0079] As will be appreciated, various embodiments may take the form of an entirely hardware embodiment, an entirely software embodiment or an embodiment combining software and hardware aspects. The following detailed description is, therefore, not to be taken in a limiting sense. References to coding methods such as “Ruby,” “Javascript,” “iOS,” “X Code,” “Angular,” “Backbone,” “Postgresql” are representative only and may change in accordance with the particular mobile framework embodiment, mobile computing device OS, etc. under consideration.

[0080] The phrase "in one embodiment" as used herein does not necessarily refer to the same embodiment, though it may. Furthermore, the phrase "in another embodiment" as used herein does not necessarily refer to a different embodiment, although it may. Thus, as described below, various embodiments may be readily combined, without departing from the scope or spirit of the invention. In addition, as used herein, the term "or" is an inclusive "or" operator, and is equivalent to the term "and/or," unless the context clearly dictates otherwise. The term "based on" is not exclusive and allows for being based on additional factors not described, unless the context clearly dictates otherwise. In addition, throughout the specification, the meaning of "a," "an," and "the" include plural references. The meaning of "in" includes "in" and "on."

[0081] The terms "cloud platform," "cloud dashboard" or "web dashboard" as used herein refer to an application platform that may provide services that may be used by mobile application developers to simplify the development of their mobile applications. The cloud platform may be accessible over a network by mobile applications operating on mobile devices. The services provided by a cloud platform may vary, but they may provide, online storage, user databases, support for payment transactions, or other significant functions delivered dynamically using our methods. While a cloud platform may be operative in a cloud-based environment, it is not so limited; other well-known operational architectures may be employed.

[0082] The terms "cloud code," "cloud code module," OR "OMVERIA platform code" as used herein refer to selection of software configurations that may be provided to the cloud platform by an OMVERIA developer or user. Cloud code is a set of software that may be customized and configured by the OMVERIA mobile application user in one or more mobile application. The cloud code may be deployed to a cloud platform and may be integrated into the cloud platform services and made available one or more mobile applications. Cloud code may be developed using a variety of computer programming languages including but not limited to the ones mentioned herein.

[0083] The foregoing has been described in terms of specific embodiments, but one of ordinary skill in the art will recognize that additional embodiments are possible without departing from its teachings. This detailed description, therefore, and particularly the specific details of the exemplary embodiments disclosed, is given primarily for clarity of understanding,

and no unnecessary limitations are to be implied, for modifications will become evident to those skilled in the art upon reading this disclosure and may be made without departing from the spirit or scope of the invention. Relatively apparent modifications, of course, include combining the various features of one or more figures with the features of one or more of the other figures.

IN THE CLAIMS:

1. In a computing system network environment, a method for selecting most relevant information for display on a display screen of a mobile computing device, comprising:

providing a user mobile computing device having at least one processor and at least one memory;

providing a remote computing system comprising one or more computing devices each having at least one processor and at least one memory, the remote computing system further comprising non-transitory storage including at least a stored database comprising a listing of past user activities and preferences and a stored database comprising informational content relevant to a plurality of geographical locations, the remote computing system further being configured at least to determine a proximity of the mobile computing device to a geographical location of the plurality of geographical locations;

communicating a unique identifier of the mobile computing device to the remote computing system;

by the remote computing system, comparing the listing of past user activities and preferences to the informational content according to said unique identifier and said detected proximity and selecting a most relevant content of the informational content according to the comparing; and

by the remote computing system, making available the selected most relevant content to the mobile computing device as a configured mobile computing device-readable application.

2. The method of claim 1, wherein the step of making available the selected most relevant content to the mobile computing device comprises displaying an icon operatively connected to the configured application on a display screen of the mobile computing device.

3. The method of claim 1, further including selecting, by the remote computing system, the most relevant content according to a stored predetermined activity category.

4. The method of claim 1, further including selecting, by the remote computing system, the most relevant content according to a stored geographical location of a creator of the informational content.

5. The method of claim 1, further including selecting, by the remote computing system, the most relevant content according to a determined date and/or time at which the mobile computing device is determined to be proximal to the geographical location.

6. The method of claim 1, wherein the unique identifier is selected from the group consisting of an identifier assigned to a Network Interface Controller (NIC) of the mobile computing device, a Media Access Control (MAC) identifier of the mobile computing device, user registration information, and combinations thereof.

7. The method of claim 1, further including, by the remote computing system, storing the unique identifier in a database of unique identifiers.

8. The method of claim 1, including making available the configured application in a format compatible with a determined operating system of the mobile computing device.

9. The method of claim 1, wherein the mobile computing device proximity is determined by one or both of a module of the mobile computing device configured for determining a geographical location by global positioning satellite (GPS) technology and module of the mobile computing device configured for determining a Relative Received Signal Strength (RSSI) of a BLUETOOTH Low Energy (BLE) beacon.

10. The method of claim 1, wherein the remote computing system comprises one or more of a server hosted in a cloud computing environment, one or more computing devices hosting an app database; one or more administrative computing devices hosting an administrative database, a router or network wi-fi transmitter, and combinations thereof.

11. The method of claim 2, further including configuring an appearance of the displayed icon according to one or both of the mobile computing device geographical location and the content of the configured application.

12. A computing network for displaying most relevant information on a display screen of a mobile computing device, comprising:

a mobile computing device having at least one processor and one memory and a display screen and configured for communicating a unique identifier of the mobile computing device to a remote computing system; and

a remote computing system comprising one or more computing devices each having at least one processor and at least one memory, the remote computing system further comprising non-transitory storage including at least a stored database comprising a listing of past user activities and preferences and a stored database comprising informational content relevant to a plurality of geographical locations;

wherein the remote computing system is configured at least to determine a proximity of the mobile computing device to a geographical location of the plurality of geographical locations, to compare the listing of past user activities and preferences to the plurality of applications according to said unique identifier and said detected proximity, to select a most relevant content of the informational content according to the comparing, and to make available the selected most relevant content of the informational content to the mobile computing device as a configured mobile computing device-readable application.

13. The system of claim 12, further wherein the selected most relevant content is made available by way of an icon operatively connected to the configured application, the icon being displayed on a display screen of the mobile computing device.

14. The system of claim 12, further wherein the remote computing system is configured to select the most relevant content according to a stored predetermined activity category.

15. The system of claim 12, further wherein the remote computing system is configured to select the most relevant content according to a stored geographical location of a creator of the informational content.

16. The system of claim 12, further wherein the remote computing system is configured to select the most relevant content according to a determined date and/or time at which the mobile computing device is determined to be proximal to the geographical location.

17. The system of claim 12, wherein the mobile computing device includes one or more modules for determining proximity by one or both of a global positioning satellite (GPS)

technology and a Relative Received Signal Strength (RSSI) of a BLUETOOTH Low Energy (BLE) beacon.

18. The system of claim 12, wherein the remote computing system comprises one or more of a server hosted in a cloud computing environment, one or more computing devices hosting an app database; one or more administrative computing devices hosting an administrative database, a router or network wi-fi transmitter, and combinations thereof.

19. The system of claim 12, wherein the unique identifier is selected from the group consisting of an identifier assigned to a Network Interface Controller (NIC) of the mobile computing device, a Media Access Control (MAC) identifier of the mobile computing device, user registration information, and combinations thereof.

20. The system of claim 12, further wherein the remote computing system is configured to alter an appearance of the displayed icon according to one or both of the geographical location and the content of the configured application.

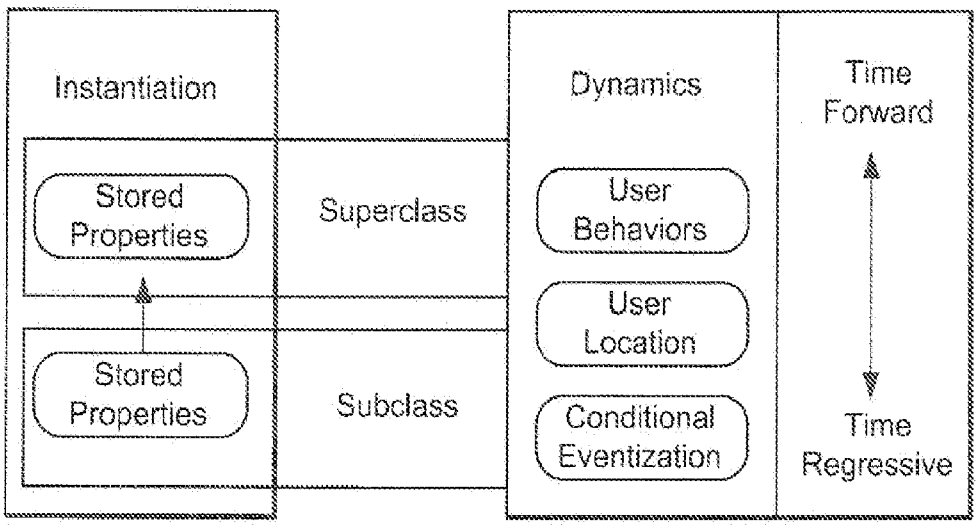
Instantiation
Stored Properties
Stored Properties
Superclass
Subclass
Dynamics
User Behaviors
User Location
Conditional Eventization
Time Forward
Time Regressive

FIG. 1

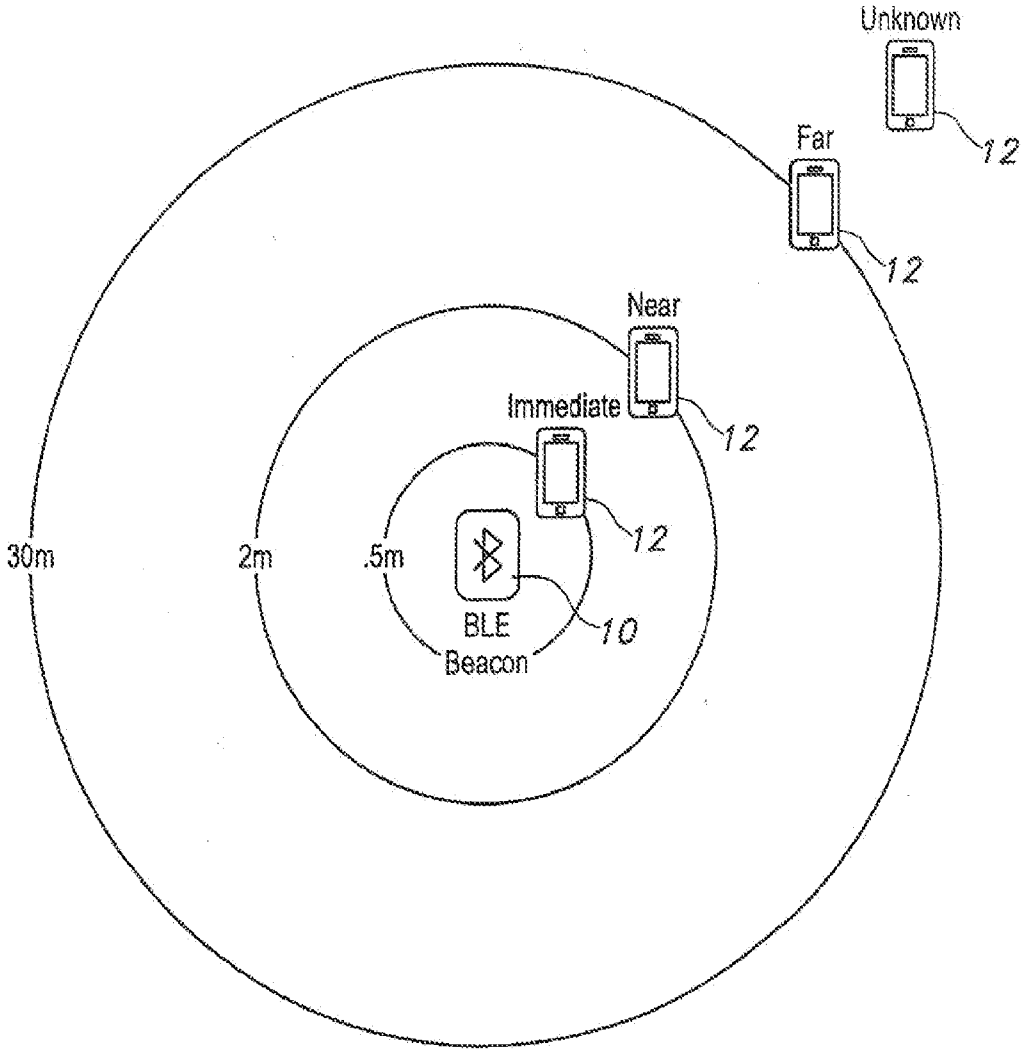

Fig.2

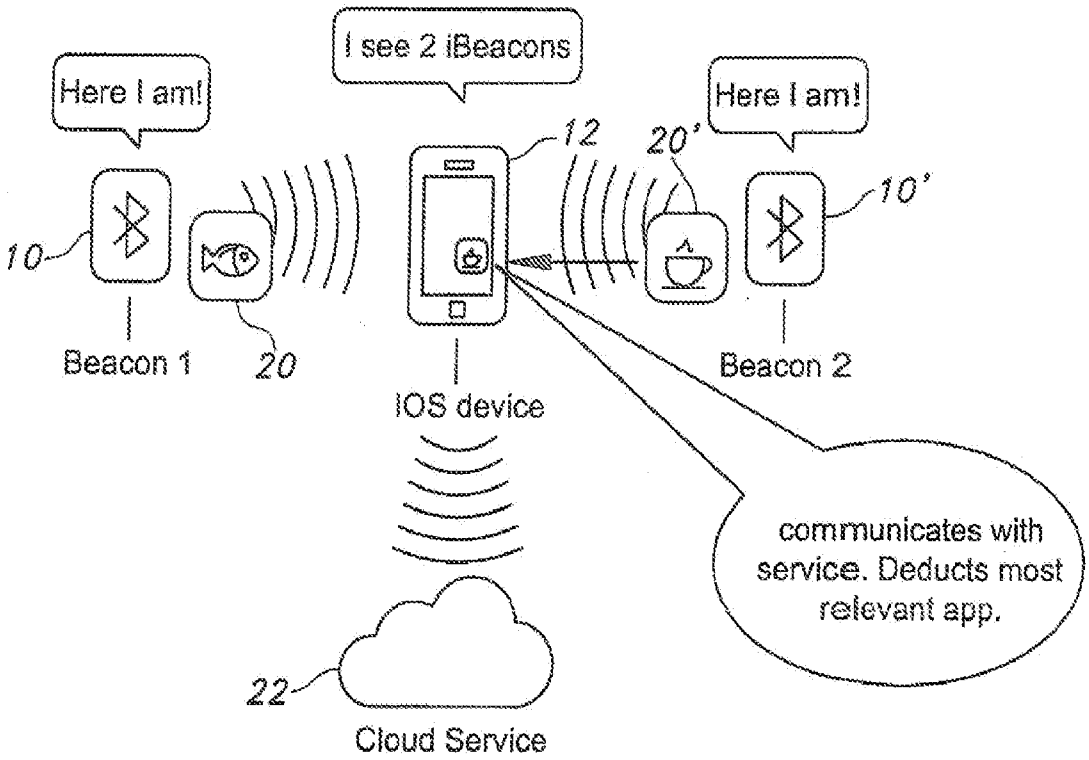
I see 2 iBeacons
Here I am!
Here I am!
12
20'
10'
10
Beacon 1
20
IOS device
Beacon 2
communicates with service. Deducts most relevant app.
22
Cloud Service

Fig. 3

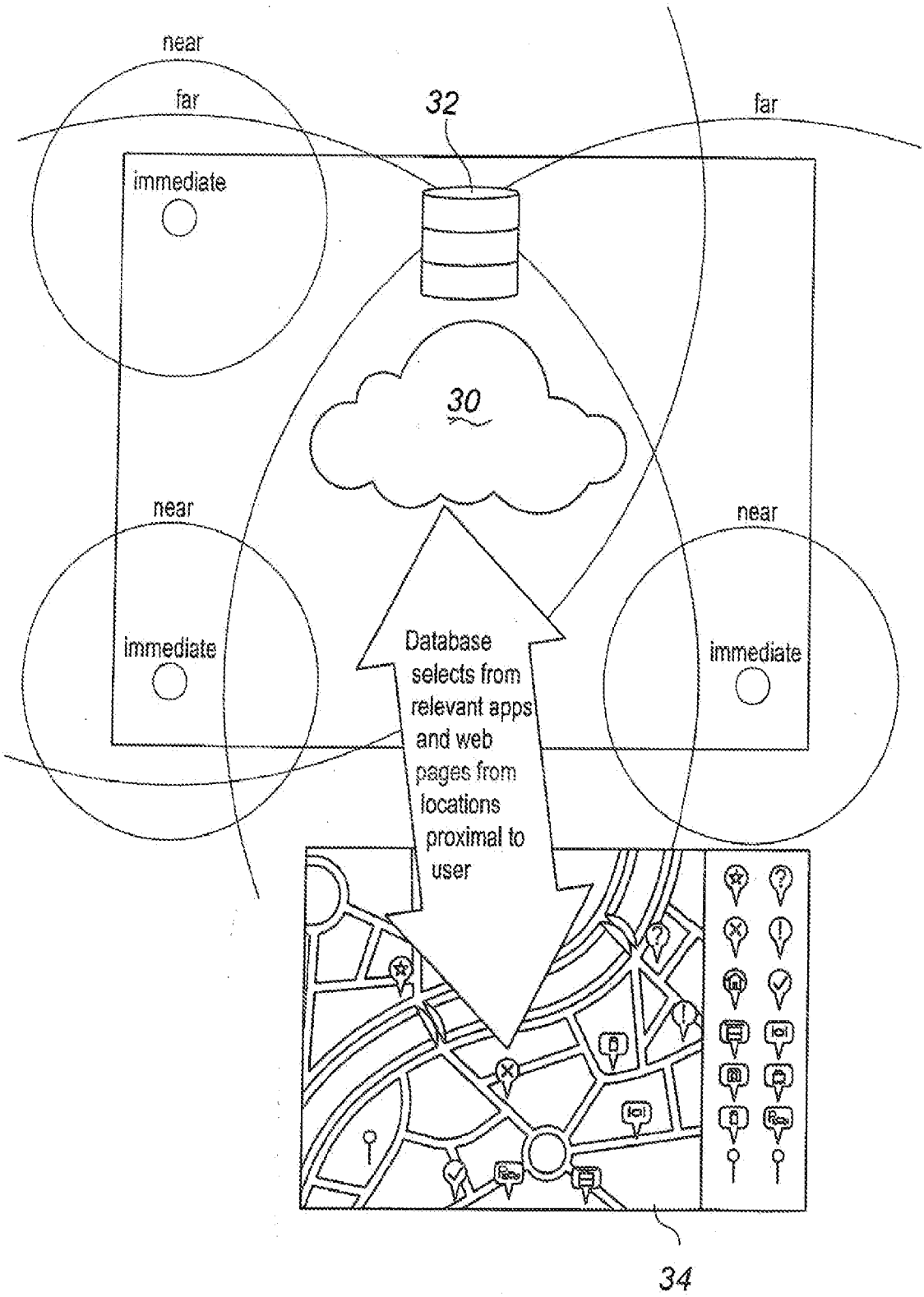
near
far
32
far
immediate
30
near
near
immediate
immediate
Database selects from relevant apps and web pages from locations proximal to user
34

Fig. 4

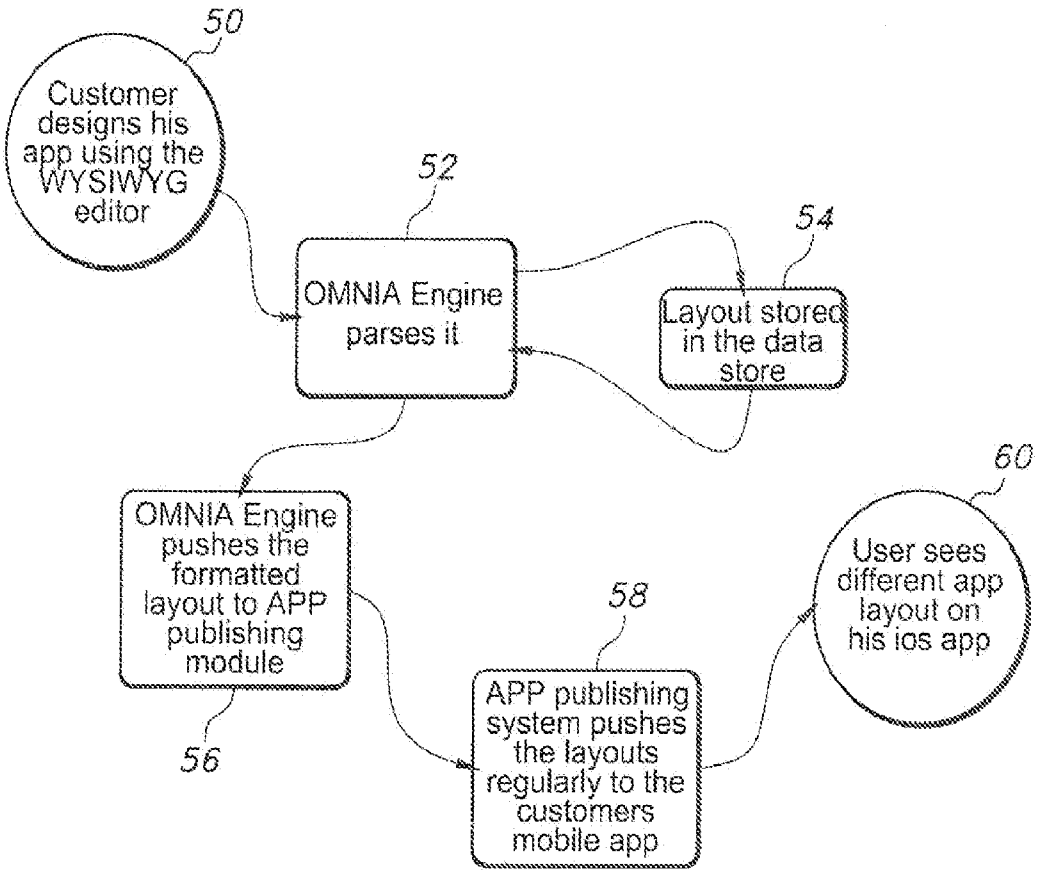
50
Customer designs his app using the WYSIWYG editor
52
OMNIA Engine parses it
54
Layout stored in the data store
OMNIA Engine pushes the formatted layout to APP publishing module
56
58
APP publishing system pushes the layouts regularly to the customers mobile app
60
User sees different app layout on his ios app

FIG. 5

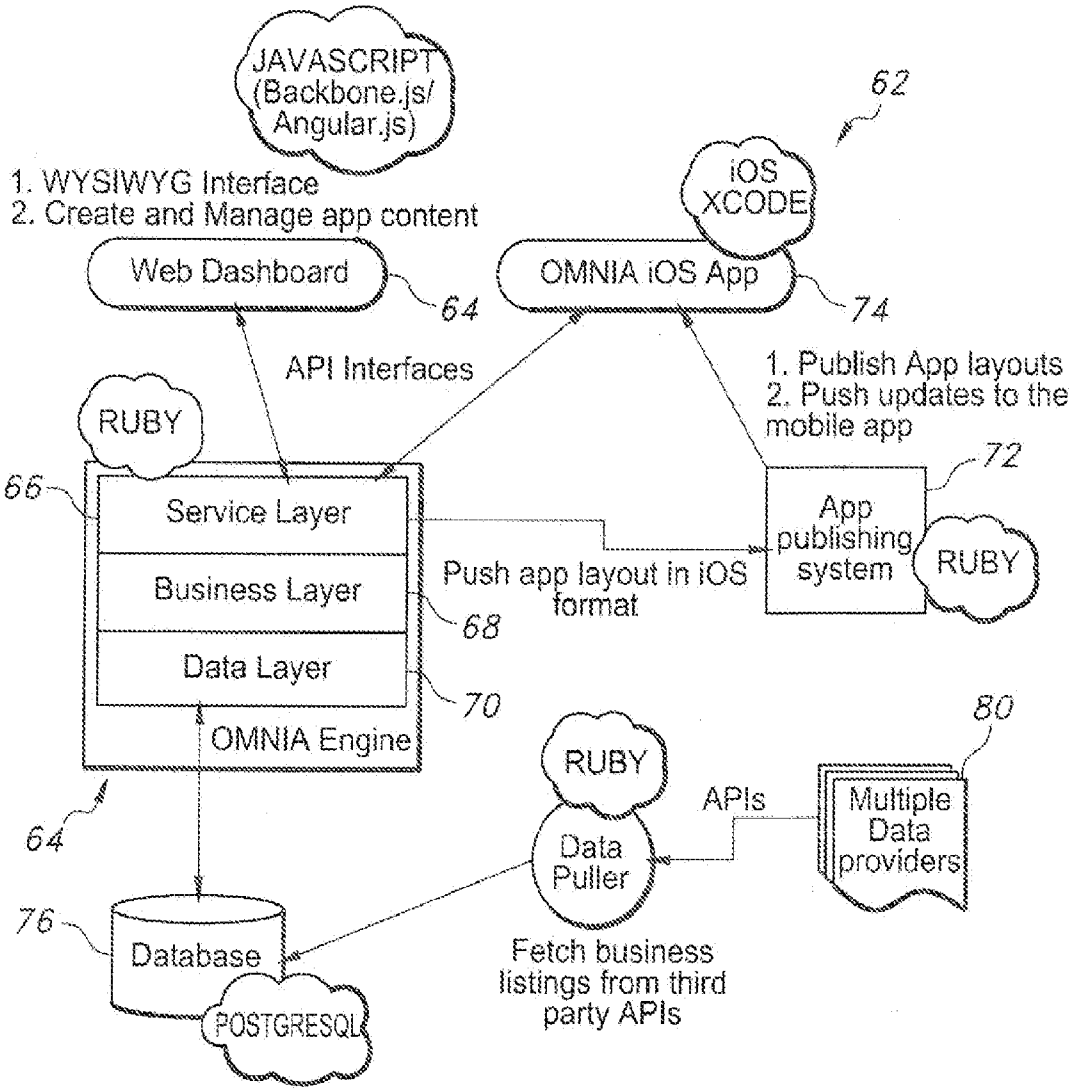
JAVASCRIPT
(Backbone.js/
Angular.js)
1. WYSIWYG Interface
2. Create and Manage app content
Web Dashboard
64
62
iOS
XCODE
OMNIA iOS App
74
API Interfaces
1. Publish App layouts
2. Push updates to the
mobile app
RUBY
66
Service Layer
Business Layer
Data Layer
OMNIA Engine
68
70
64
Push app layout in iOS
format
72
App
publishing
system
RUBY
80
RUBY
Data
Puller
APIs
Multiple
Data
providers
76
Database
POSTGRESQL
Fetch business
listings from third
party APIs

FIG. 6

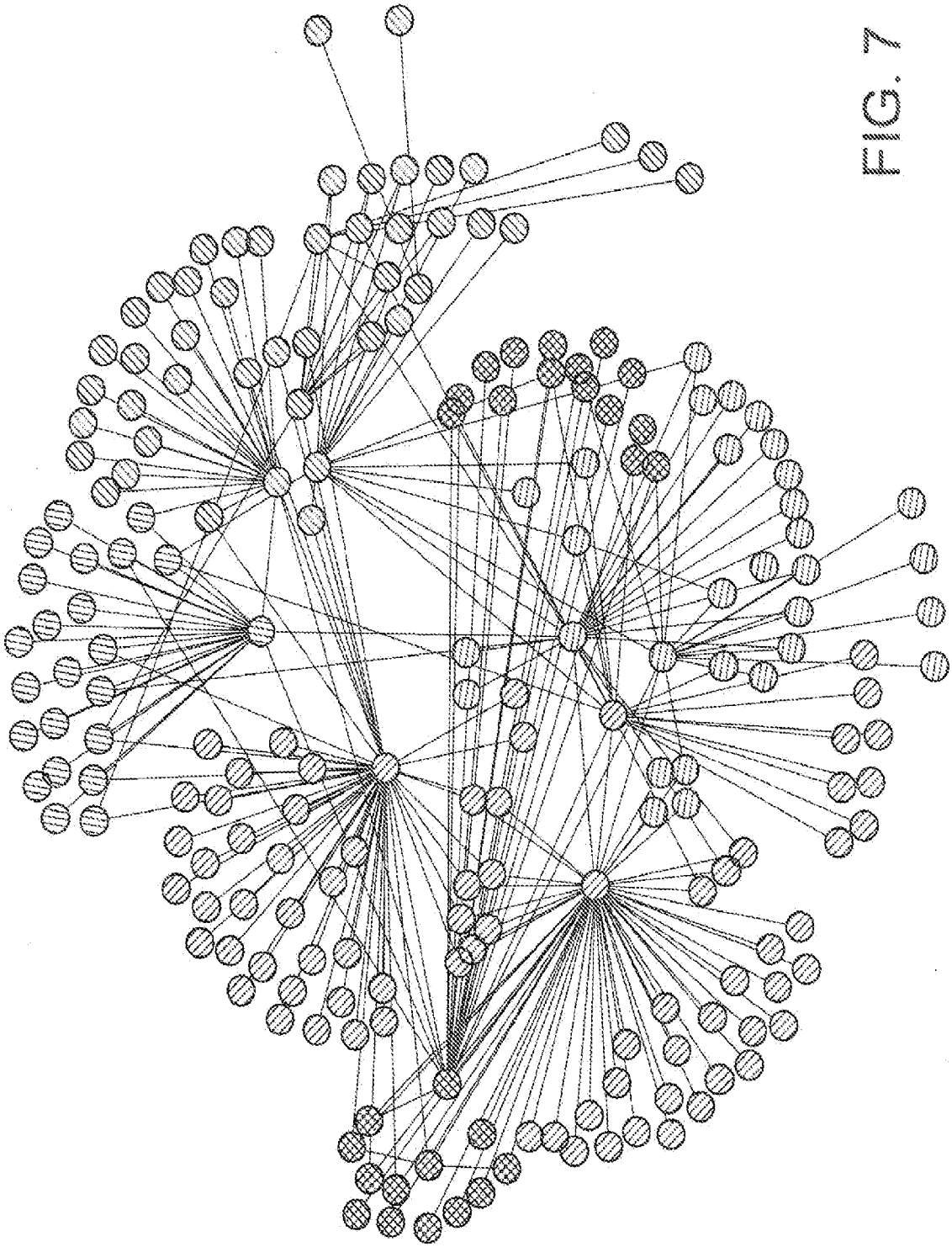

FIG. 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2015/036039

A. CLASSIFICATION OF SUBJECT MATTER
IPC(8) - H04W 4/02 (2015.01)
CPC - H04W 4/02 (2015.04)
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)
IPC(8) - G06Q 30/02, H04W 4/00, H04W 4/02, H04W 4/18, H04W 8/18 (2015.01)
USPC - 455/414.2, 456.1, 456.3, 456.5; 705/14.49, 14.66, 14.67

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
CPC - G06Q 30/02, G06Q 30/0251, G06Q 30/0255, G06Q 30/0261, G06Q 30/0267, H04W 4/00, H04W 4/02, H04W 4/18, H04W 8/18 (2015.04) (keyword delimited)

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
Orbit, Google Patents, Google.
Search terms used: mobile, display, application, content, identification, database, activities, preferences, geographic location, proximity

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X -- Y	US 2011/0258049 A1 (RAMER et al) 20 October 2011 (20.10.2011) entire document	1-5, 7-18 and 20 ---- 6 and 19
Y	US 2011/0029362 A1 (ROEDING et al) 03 February 2011 (03.02.2011) entire document	6 and 19
A	US 2013/0210461 A1 (MOLDAVSKY et al) 15 August 2013 (15.08.2013) entire document	1-20
A	US 2010/0138416 A1 (BELLOTTI) 03 June 2010 (03.06.2010) entire document	1-20
A	US 2009/0197582 A1 (LEWIS et al) 06 August 2009 (06.08.2009) entire document	1-20
A	US 2003/0069693 A1 (SNAPP et al) 10 April 2003 (10.04.2003) entire document	1-20

☐ 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	Date of mailing of the international search report
11 August 2015	31 AUG 2015
Name and mailing address of the ISA/ Mail Stop PCT, Attn: ISA/US, Commissioner for Patents P.O. Box 1450, Alexandria, Virginia 22313-1450 Facsimile No. 571-273-8300	Authorized officer Blaine Copenheaver PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774

Form PCT/ISA/210 (second sheet) (January 2015)