



- (51) **International Patent Classification:**  
A63F 7/00 (2006.01)
- (21) **International Application Number:**  
PCT/US20 15/024025
- (22) **International Filing Date:**  
2 April 2015 (02.04.2015)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**  
61/974,119 2 April 2014 (02.04.2014) US
- (71) **Applicant: ZYNGA INC.** [US/US]; 699 Eighth Street, San Francisco, California 94103 (US).
- (72) **Inventors: MARAT, Rolan;** 699 Eighth Street, San Francisco, California 94103 (US). **WAJIMA, Kohta Aoki;** 699 Eighth Street, San Francisco, California 94103 (US). **HARRIS, Dylan Kuper;** 699 Eighth Street, San Francisco, California 94103 (US).
- (74) **Agents: MADDEN, Robert B.** et al; Schwegman Lundberg & Woessner, P.A., P.O. Box 2938, Minneapolis, Minnesota 55402 (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:** SYSTEMS AND METHODS OF DYNAMICALLY SELECTING CONTACTS AND PROMOTING PRODUCTS

(57) **Abstract:** A system, a machine-readable storage medium storing instructions, and a computer-implemented method are directed to a Contact Ranking Engine (hereinafter "CR Engine"). The CR Engine identifies a plurality of contacts of a player of a first game on a gaming network environment (or a game networking system). The CR Engine collects contact attributes for each of the plurality of contacts from one or more contact attribute sources. The CR Engine requests a first configuration table related to the first game. Upon receiving the first configuration table, the CR Engine scores the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table. The CR Engine ranks in a contacts ranking list each contact based on a respective score. The CR Engine selects a portion of the contacts ranking list. The CR Engine displays each contact from the selected portion.

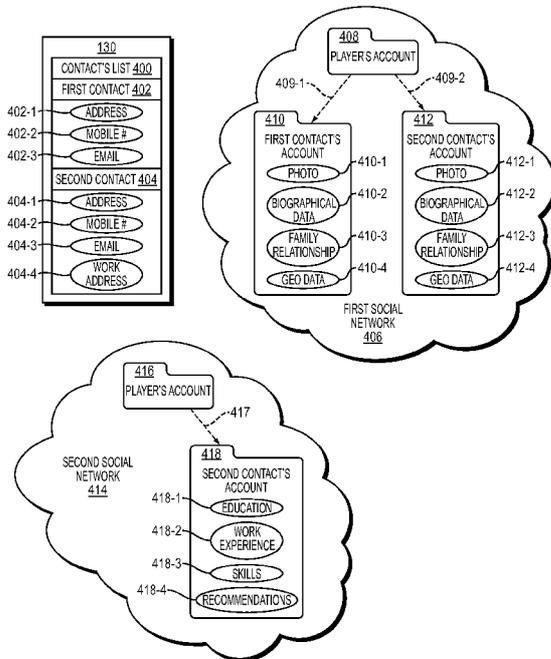


FIG. 4

WO2015/153853 A1

**SYSTEMS AND METHODS OF DYNAMICALLY SELECTING  
CONTACTS AND PROMOTING PRODUCTS**

5

**CROSS-REFERENCE TO RELATED APPLICATIONS**

[0001] This application claims the benefit of priority to U.S. Provisional Patent Application entitled "*Systems and Methods of Dynamically Selecting Contacts and Promoting Products*," Serial No. 61/974,119, filed April 2, 2014,  
10 which is hereby incorporated herein by reference in its entirety.

**TECHNICAL FIELD**

[0002] The present disclosure relates to games and applications in general and in particular to computer-implemented games. In an example embodiment,  
15 a player's contacts may be dynamically selected and products may be dynamically promoted.

**BACKGROUND**

[0003] In many games, there is a virtual world or some other imagined  
20 playing space where a player/user of the game controls one or more player characters (herein "character," "player character," or "PC"). Player characters can be considered in-game representations of the controlling player. As used herein, the terms "player," "user," "entity," and "friend" may refer to the in-game player character controlled by that player, user, entity, or friend, unless  
25 context suggests otherwise. The game display can display a representation of the player character. A game engine accepts inputs from the player, determines player character actions, decides outcomes of events and presents the player with a game display illuminating what happened. In some games, there are multiple players, wherein each player controls one or more player characters.

30 [0004] In many computer games, there are various types of in-game assets (aka "rewards" or "loot") that a player character can obtain within the game. For example, a player character may acquire game points, gold coins, experience

points, character levels, character attributes, virtual cash, game keys, or other in-game items of value. In many computer games, there are also various types of in-game obstacles that a player must overcome to advance within the game. In-game obstacles can include tasks, puzzles, opponents, levels, gates, actions, etc.

5 In some games, a goal of the game may be to acquire certain in-game assets, which can then be used to complete in-game tasks or to overcome certain in-game obstacles. For example, a player may be able to acquire a virtual key (i.e., the in-game asset) that can then be used to open a virtual door (i.e., the in-game obstacle).

10 **[0005]** An electronic social networking system typically operates with one or more social networking servers providing interaction between users such that a user can specify other users of the social networking system as "friends." A collection of users and the "friend" connections between users can form a social graph that can be traversed to find second, third and more remote connections  
15 between users, much like a graph of nodes connected by edges can be traversed.

**[0006]** Many online computer games are operated on an online social networking system. Such an online social networking system allows both users and other parties to interact with the computer games directly, whether to play the games or to retrieve game- or user-related information. Internet users may  
20 maintain one or more accounts with various service providers, including, for example, online game networking systems and online social networking systems. Online systems can typically be accessed using browser clients (e.g., Firefox, Chrome, Internet Explorer).

25 **BRIEF DESCRIPTION OF THE DRAWINGS**

**[0007]** The present disclosure is illustrated by way of example, and not limitation, in the figures of the accompanying drawings, in which like reference numerals indicate similar elements unless otherwise indicated. In the drawings,

**[0008]** FIG. 1 is a schematic diagram showing an example of a system,  
30 according to some example embodiments.

**[0009]** FIG. 2 is a schematic diagram showing an example of a social network within a social graph, according to some embodiments.

**[0010]** FIG. 3 is a block diagram illustrating components of a computing device, according to some example embodiments.

[0011] FIG. 4 is a block diagram illustrating various contact attribute sources according to some example embodiments.

[0012] FIG. 5 is a block diagram illustrating a configuration table applying updateable scoring rules to a plurality of contact attributes sets according to  
5 some example embodiments.

[0013] FIG. 6 is a flowchart showing an example method of creating a contacts ranking list, according to some example embodiments.

[0014] FIG. 7 is a diagrammatic representation of an example data flow between example components of the example system of FIG. 1, according to  
10 some example embodiments.

[0015] FIG. 8 illustrates an example computing system architecture, which may be used to implement a server or a client system illustrated in FIG. 9, according to some example embodiments.

[0016] FIG. 9 illustrates an example network environment, in which various  
15 example embodiments may operate.

### **DETAILED DESCRIPTION**

[0017] A system, a machine-readable storage medium storing instructions, and a computer-implemented method are described herein to determine which  
20 contact(s) from a plurality of a player's contacts should receive a promotion or an invitation related to a game. In the following description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of example embodiments. It will be evident, however, to one skilled in the art that the present technology may be practiced without  
25 these specific details.

[0018] Various embodiments described herein are directed to a Contact Ranking Engine (hereinafter "CR Engine"). The CR Engine collects attributes of a player's contacts and scores each collected attribute. Specifically, the CR Engine identifies a plurality of contacts of a player of a first game on a  
30 gaming network environment (or a game networking system). The Cr Engine collects contact attributes for each of the plurality of contacts from one or more contact attribute sources. The CR Engine requests a first configuration table related to the first game. Upon receiving the first configuration table, the CR Engine scores the collected contact attributes of each contact of the player

according to one or more updateable rules of the first configuration table. The CR Engine ranks in a contacts ranking list each contact based on a respective score. The CR Engine selects a portion of the contacts ranking list. The CR Engine displays a selectable instance of each contact included in the selected  
5 portion of the contacts ranking list.

[0019] The CR Engines receives from the player a selection of a selectable instance of a particular contact. The CR Engines forwards a promotion related to the first game to the particular contact (such as forwarding the promotion to an account of the particular contact on a social network system). The CR Engine  
10 further triggers an award for the player given in exchange for the player having selected the particular contact.

[0020] In an example embodiment, the CR Engine executes on a client device associated with a player of one or more games accessible by the client device on a game networking system. The one or more games can be  
15 downloaded from the game networking system and installed on the client device. The player is associated with one or more contacts, where contacts represent identities of people other than the player.

[0021] The CR Engine accesses the contacts and collects contact attributes for each of the player's contacts. The CR Engine requests a configuration table from the game networking system. The configuration table includes at least one  
20 rule for scoring a contact based on the contact's attributes. The CR Engine receives the configuration table from the game networking system and scores each contact according to the rules of the configuration table.

[0022] The CR Engine presents at least a portion of the highest scoring contacts on a display of the client device for selection by the player. The CR Engine receives a selection by the player of one or more of the displayed contacts. The CR Engine triggers promotions and/or invitations related to the  
25 first game to be sent to the contacts selected by the player.

[0023] In various embodiments, the CR Engines receives an indication(s)  
30 that the player has given permission for the CR Engine to access the player's connections on a first social network system, a second social network system, and a contacts listing on the client device. The CR Engine identifies a first contact which is a first connection (such as a "friend") on the first social network. The CR Engine identifies a second contact which is a second

connection on the first social network. The CR Engine identifies additional contact attributes for both the first and second contacts on the contacts listing on the client device.

5 [0024] The CR Engine further accesses the second social network system and identifies a connection on the second social network system between the player and the second contact. The CR Engine identifies additional contact attributes for the second contact from the second social network - but none for the first contact. That is, based on the sources of contact data that the CR Engine has been permitted to access, the CR Engine collects various kinds of contact  
10 attributes for each contact. As in this example embodiment, the CR Engine collects more contact attributes for the second contact than the first contact due to the player having a connection with the second contact on the second social network system while the player does not have a connection with the first contact on the second social network.

15 [0025] The CR Engine executing on the client device identifies one of a plurality of games installed on the client device from the game networking system. The CR Engine identifies a first game of the plurality of installed games. The CR Engine requests a first configuration table from a plurality of configuration tables available on the game networking system, wherein each of  
20 the configuration tables on the server system is related to a different game. The first configuration table is related to the first game. The first configuration table comprises one or more updateable rules for scoring a contact based on each the contact's attributes. Since the first configuration table is related to the first game, the first configuration table scores contacts in order to identify contacts of  
25 the player that are most relevant to the first game. In the words, the first configuration table includes one or more updateable rules that score the collected contact attributes of the first and second contacts differently than a second configuration table that is related to a different game.

[0026] In various embodiments, a first instance of the first configuration  
30 table related to the first game includes one or more updateable rules for determining contacts most likely to respond to a promotion related to a game. The promotion can be a promotion (or an invitation to join) for the first game or a different game. It is understood that a configuration table's rules are

determined to identify contacts most likely to take advantage of a promotion based on how each respective contact's collected attributes are scored.

[0027] The game networking system updates the configuration table according to new promotions. That is, a configuration table includes one or more updateable rules for identifying a player's contacts that are most likely to respond to a promotion. If the promotion is updated at the game networking system to promote a different game or product, the configuration table includes one or more updateable rules that correspond to the updated promotion. It is understood that the configuration table score a player's contacts differently than when it is updated to correspond to a different promotion.

[0028] When the CR Engine executing on the client system receives the first instance of the first configuration table from the game networking system, the CR Engine stores the first instance of the first configuration table in a cache for efficient access and reuse of the first instance of the first configuration table. If the promotion is to be modified, a second instance of the first configuration table is created on the game networking system by changing the one or more updateable rules of the first configuration table. The updated first configuration table for the modified promotion can then be received by the CR Engine executing on the client device and stored in the cache.

[0029] It is understood that various embodiments include the generation of modules to cause any component(s) of a social networking system, game networking system (i.e. online gaming system) and/or a client system to perform any and all of the actions, operations and steps described herein.

[0030] It is further understood that, in various embodiments, any of the one or more modules may comprise source code that, when compiled by a computing device(s), creates object code that causes the computing device(s) to perform the various actions, operations and steps described herein.

[0031] It is understood that embodiments described herein are not limited to being implemented with online games, but rather, can be implemented for any kind of online environment.

[0032] It is further understood that embodiments described herein include the generation of graphical display data based on a result of any and all of the actions, operations and steps described herein.

Social Network Systems and Game Networking Systems

[0033] FIG. 1 illustrates an example of a system for implementing various disclosed embodiments. In particular embodiments, system 100 comprises player 101, social networking system 120a, game networking system 120b (i.e. online gaming system), client system 130, and network 160. The components of system 100 can be connected to each other in any suitable configuration, using any suitable type of connection. The components may be connected directly or over a network 160, which may be any suitable network. For example, one or more portions of network 160 may be an ad hoc network, an intranet, an extranet, a virtual private network (VPN), a local area network (LAN), a wireless LAN (WLAN), a wide area network (WAN), a wireless WAN (WWAN), a metropolitan area network (MAN), a portion of the Internet, a portion of the Public Switched Telephone Network (PSTN), a cellular telephone network, another type of network, or a combination of two or more such networks.

[0034] Social networking system 120a (i.e. social network system) is a network-addressable computing system that can host one or more social graphs. Social networking system 120a can generate, store, receive, and transmit social networking data. Social networking system 120a can be accessed by the other components of system 100 either directly or via network 160. Game networking system 120b is a network-addressable computing system that can host one or more online games. Game networking system 120b can generate, store, receive, and transmit game-related data, such as, for example, game account data, game input, game state data, and game displays. Game networking system 120b can be accessed by the other components of system 100 either directly or via network 160. Player 101 uses client system 130 to access, send data to, and receive data from social networking system 120a and game networking system 120b. Client system 130 can access social networking system 120a or game networking system 120b directly, via network 160, or via a third-party system. As an example and not by way of limitation, client system 130 accesses game networking system 120b via social networking system 120a. Client system 130 can be any suitable computing device, such as a personal computer, laptop, cellular phone, smart phone, computing tablet, etc.

[0035] Although FIG. 1 illustrates a particular number of players 101, social network systems 120a, game networking systems 120b, client systems 130, and

networks 160, this disclosure contemplates any suitable number of players 101, social network systems 120a, game networking systems 120b, client systems 130, and networks 160. As an example and not by way of limitation, system 100 includes one or more game networking systems 120b and no social networking systems 120a. As another example and not by way of limitation, system 100 includes a system that comprises both social networking system 120a and game networking system 120b. Moreover, although **FIG. 1** illustrates a particular arrangement of player 101, social networking system 120a, game networking system 120b, client system 130, and network 160, this disclosure contemplates any suitable arrangement of player 101, social networking system 120a, game networking system 120b, client system 130, and network 160.

**[0036]** The components of system 100 are connected to each other using any suitable connections 110. For example, suitable connections 110 include wireline (such as, for example, Digital Subscriber Line (DSL) or Data Over Cable Service Interface Specification (DOCSIS)), wireless (such as, for example, Wi-Fi or Worldwide Interoperability for Microwave Access (WiMAX)) or optical (such as, for example, Synchronous Optical Network (SONET) or Synchronous Digital Hierarchy (SDH)) connections. In particular embodiments, one or more connections 110 each include an ad hoc network, an intranet, an extranet, a VPN, a LAN, a WLAN, a WAN, a WWAN, a MAN, a portion of the Internet, a portion of the PSTN, a cellular telephone network, or another type of connection, or a combination of two or more such connections. Connections 110 need not necessarily be the same throughout system 100. One or more first connections 110 may differ in one or more respects from one or more second connections 110. Although **FIG. 1** illustrates particular connections between player 101, social networking system 120a, game networking system 120b, client system 130, and network 160, this disclosure contemplates any suitable connections between player 101, social networking system 120a, game networking system 120b, client system 130, and network 160. As an example and not by way of limitation, in particular embodiments, client system 130 may have a direct connection to social networking system 120a or game networking system 120b, bypassing network 160.

Online Games and Game Systems

[0037] In an online computer game, a game engine manages the game state of the game. Game state comprises all game play parameters, including player character state, non-player character (NPC) state, in-game object state, game world state (e.g., internal game clocks, game environment), and other game play parameters. Each player 101 controls one or more player characters (PCs). The game engine controls all other aspects of the game, including non-player characters (NPCs), and in-game objects. The game engine also manages game state, including player character state for currently active (online) and inactive (offline) players.

[0038] An online game can be hosted by game networking system 120b (i.e. online gaming system), which includes a Notification Generator 150 that performs operations according to embodiments as described herein. The game networking system 120b can be accessed using any suitable connection with a suitable client system 130. A player has a game account on game networking system 120b, wherein the game account can contain a variety of information associated with the player (e.g., the player's personal information, financial information, purchase history, player character state, game state). In some embodiments, a player plays multiple games on game networking system 120b, which maintains a single game account for the player with respect to all the games, or multiple individual game accounts for each game with respect to the player. In some embodiments, game networking system 120b can assign a unique identifier to each player 101 of an online game hosted on game networking system 120b. Game networking system 120b can determine that a player 101 is accessing the online game by reading the user's cookies, which may be appended to HTTP requests transmitted by client system 130, and/or by the player 101 logging onto the online game.

[0039] In particular embodiments, player 101 accesses an online game and control the game's progress via client system 130 (e.g., by inputting commands to the game at the client device). Client system 130 can display the game interface, receive inputs from player 101, transmitting user inputs or other events to the game engine, and receive instructions from the game engine. The game engine can be executed on any suitable system (such as, for example, client

system 130, social networking system 120a, or game networking system 120b). As an example and not by way of limitation, client system 130 can download client components of an online game, which are executed locally, while a remote game server, such as game networking system 120b, provides backend support for the client components and may be responsible for maintaining application data of the game, processing the inputs from the player, updating and/or synchronizing the game state based on the game logic and each input from the player, and transmitting instructions to client system 130. As another example and not by way of limitation, each time player 101 provides an input to the game through the client system 130 (such as, for example, by typing on the keyboard or clicking the mouse of client system 130), the client components of the game transmit the player's input to game networking system 120b.

#### Storing Game-Related Data

15 [0040] A database stores any data relating to game play within a game networking system 120b. The database includes database tables for storing a player game state that includes information about the player's virtual gameboard, the player's character, or other game-related information. For example, player game state includes virtual objects owned or used by the player, placement positions for virtual structural objects in the player's virtual gameboard, and the like. Player game state also includes in-game obstacles of tasks for the player (e.g., new obstacles, current obstacles, completed obstacles, etc.), the player's character attributes (e.g., character health, character energy, amount of coins, amount of cash or virtual currency, etc.), and the like.

25 [0041] The database also includes database tables for storing a player profile that includes user-provided player information that is gathered from the player, the player's client device, or an affiliate social network. The user-provided player information includes the player's demographic information, the player's location information (e.g., a historical record of the player's location during game play as determined via a GPS-enabled device or the internet protocol (IP) address for the player's client device), the player's localization information (e.g., a list of languages chosen by the player), the types of games played by the player, and the like.

[0042] In some example embodiments, the player profile also includes derived player information determined from other information stored in the database. The derived player information includes information that indicates the player's level of engagement with the virtual game, the player's friend preferences, the player's reputation, the player's pattern of game-play, and the like. For example, the game networking system 120b determines the player's friend preferences based on player attributes that the player's first-degree friends have in common, and stores these player attributes as friend preferences in the player profile. Furthermore, the game networking system 120b determines reputation-related information for the player based on user-generated content (UGC) from the player or the player's N<sup>th</sup> degree friends (e.g., in-game messages or social network messages), and stores this reputation-related information in the player profile. The derived player information also includes information that indicates the player's character temperament during game play, anthropological measures for the player (e.g., tendency to like violent games), and the like.

[0043] In some example embodiments, the player's level of engagement is indicated from the player's performance within the virtual game. For example, the player's level of engagement is determined based on one or more of the following: a play frequency for the virtual game or for a collection of virtual games; an interaction frequency with other players of the virtual game; a response time for responding to in-game actions from other players of the virtual game; and the like.

[0044] In some example embodiments, the player's level of engagement includes a likelihood value indicating a likelihood that the player performs a desired action. For example, the player's level of engagement indicates a likelihood that the player chooses a particular environment, or completes a new challenge within a determinable period of time from when it is first presented to him.

[0045] In some example embodiments, the player's level of engagement includes a likelihood that the player is a leading player of the virtual game (a likelihood to lead). The game networking system 120b determines the player's likelihood to lead value based on information from other players that interact with this player. For example, the game networking system 120b determines the player's likelihood to lead value by measuring the other players' satisfaction in

the virtual game, measuring their satisfaction from their interaction with the player, measuring the game-play frequency for the other players in relation to their interaction frequency with the player (e.g., the ability for the player to retain others), and/or the like.

5 [0046] The game networking system 120b also determines the player's likelihood to lead value based on information about the player's interactions with others and the outcome of these interactions. For example, the game networking system 120b determines the player's likelihood to lead value by measuring the player's amount of interaction with other players (e.g., as measured by a number  
10 of challenges that the player cooperates with others, and/or an elapsed time duration related thereto), the player's amount of communication with other players, the tone of the communication sent or received by the player, and/or the like. Moreover, the game networking system 120b determines the player's likelihood to lead value based on determining a likelihood for the other players  
15 to perform a certain action in response to interacting or communicating with the player and/or the player's virtual environment.

Game Systems, Social Networks, and Social Graphs:

[0047] In an online multiplayer game, players control player characters  
20 (PCs), a game engine controls non-player characters (NPCs) and game features, and the game engine also manages player character state and game state and tracks the state for currently active (i.e., online) players and currently inactive (i.e., offline) players. A player character can have a set of attributes and a set of friends associated with the player character. As used herein, the term "player  
25 character state" can refer to any in-game characteristic of a player character, such as location, assets, levels, condition, health, status, inventory, skill set, name, orientation, affiliation, specialty, and so on. Player characters are displayed as graphical avatars within a user interface of the game. In other implementations, no avatar or other graphical representation of the player  
30 character is displayed. Game state encompasses the notion of player character state and refers to any parameter value that characterizes the state of an in-game element, such as a non-player character, a virtual object (such as a wall or castle), etc. The game engine uses player character state to determine the outcome of game events, sometimes also considering set or random variables.

Generally, a player character's probability of having a more favorable outcome is greater when the player character has a better state. For example, a healthier player character is less likely to die in a particular encounter relative to a weaker player character or non-player character. In some embodiments, the game engine  
5 can assign a unique client identifier to each player.

[0048] In particular embodiments, player 101 accesses particular game instances of an online game. A game instance is copy of a specific game play area that is created during runtime. In particular embodiments, a game instance is a discrete game play area where one or more players 101 can interact in  
10 synchronous or asynchronous play. A game instance may be, for example, a level, zone, area, region, location, virtual space, or other suitable play area. A game instance may be populated by one or more in-game objects. Each object is defined within the game instance by one or more variables, such as, for example, position, height, width, depth, direction, time, duration, speed, color, and other  
15 suitable variables. A game instance may be exclusive (i.e., accessible by specific players) or non-exclusive (i.e., accessible by any player). In particular embodiments, a game instance is populated by one or more player characters controlled by one or more players 101 and one or more in-game objects controlled by the game engine. When accessing an online game, the game engine  
20 allows player 101 to select a particular game instance to play from a plurality of game instances. Alternatively, the game engine automatically selects the game instance that player 101 will access. In particular embodiments, an online game comprises only one game instance that all players 101 of the online game can access.

[0049] In particular embodiments, a specific game instance is associated with one or more specific players. A game instance is associated with a specific player when one or more game parameters of the game instance are associated with the specific player. As an example and not by way of limitation, a game instance associated with a first player is named "First Player's Play Area." This  
30 game instance is populated with the first player's PC and one or more in-game objects associated with the first player. In particular embodiments, a game instance associated with a specific player is only accessible by that specific player. As an example and not by way of limitation, a first player accesses a first game instance when playing an online game, and this first game instance is

inaccessible to all other players. In other embodiments, a game instance associated with a specific player is accessible by one or more other players, either synchronously or asynchronously with the specific player's game play. As an example and not by way of limitation, a first player is associated with a first game instance, but the first game instance can be accessed by all first-degree friends in the first player's social network. In particular embodiments, the game engine creates a specific game instance for a specific player when that player accesses the game. As an example and not by way of limitation, the game engine creates a first game instance when a first player initially accesses an online game, and that same game instance is loaded each time the first player accesses the game. As another example and not by way of limitation, the game engine creates a new game that is created randomly or selected from a set of predetermined game instances. In particular embodiments, the set of in-game actions available to a specific player is different in a game instance that is associated with that player compared to a game instance that is not associated with that player. The set of in-game actions available to a specific player in a game instance associated with that player is a subset, superset, or independent of the set of in-game actions available to that player in a game instance that is not associated with him. As an example and not by way of limitation, a first player is associated with Blackacre Farm in an online farming game. The first player is able to plant crops on Blackacre Farm. If the first player accesses game instance associated with another player, such as Whiteacre Farm, the game engine does not allow the first player to plant crops in that game instance. However, other in-game actions are available to the first player, such as watering or fertilizing crops on Whiteacre Farm.

**[0050]** In particular embodiments, a game engine can interface with a social graph. Social graphs are models of connections between entities (e.g., individuals, users, contacts, friends, players, player characters, non-player characters, businesses, groups, associations, concepts, etc.). These entities are considered "users" of the social graph; as such, the terms "entity" and "user" may be used interchangeably when referring to social graphs herein. A social graph can have a node for each entity and edges to represent relationships between entities. A node in a social graph can represent any entity. In particular embodiments, a unique client identifier can be assigned to each user in the social

graph. This disclosure assumes that at least one entity of a social graph is a player or player character in an online multiplayer game, though this disclosure any suitable social graph users.

**[0051]** The minimum number of edges required to connect a player (or  
5 player character) to another user is considered the degree of separation between them. For example, where the player and the user are directly connected (one edge), they are deemed to be separated by one degree of separation. The user would be a so-called "first-degree friend" of the player. Where the player and the user are connected through one other user (two edges), they are deemed to be  
10 separated by two degrees of separation. This user would be a so-called "second-degree friend" of the player. Where the player and the user are connected through N edges (or N-1 other users), they are deemed to be separated by N degrees of separation. This user would be a so-called "Nth-degree friend." As used herein, the term "friend" means only first-degree friends, unless context  
15 suggests otherwise.

**[0052]** Within the social graph, each player (or player character) has a social network. A player's social network includes all users in the social graph within  $N_{\max}$  degrees of the player, where  $N_{\max}$  is the maximum degree of separation allowed by the system managing the social graph (such as, for example, social  
20 networking system 120a or game networking system 120b). In one embodiment,  $N_{\max}$  equals 1, such that the player's social network includes only first-degree friends. In another embodiment,  $N_{\max}$  is unlimited and the player's social network is coextensive with the social graph.

**[0053]** In particular embodiments, the social graph is managed by game  
25 networking system 120b, which is managed by the game operator. In other embodiments, the social graph is part of a social networking system 120a managed by a third-party (e.g., Facebook, Friendster, Myspace). In yet other embodiments, player 101 has a social network on both game networking system  
30 120b and social networking system 120a, wherein player 101 can have a social network on the game networking system 120b that is a subset, superset, or independent of the player's social network on social networking system 120a. In such combined systems, game network system 120b can maintain social graph information with edge type attributes that indicate whether a given friend is an "in-game friend," an "out-of-game friend," or both. The various embodiments

disclosed herein are operable when the social graph is managed by social networking system 120a, game networking system 120b, or both.

[0054] FIG. 2 shows an example of a social network within a social graph. As shown, Player 201 can be associated, connected or linked to various other users, or "friends," within the social network 250. These associations, connections or links can track relationships between users within the social network 250 and are commonly referred to as online "friends" or "friendships" between users. Each friend or friendship in a particular user's social network within a social graph is commonly referred to as a "node." For purposes of illustration and not by way of limitation, the details of social network 250 will be described in relation to Player 201. As used herein, the terms "player," "user" and "account" can be used interchangeably and can refer to any user or character in an online game networking system or social networking system. As used herein, the term "friend" can mean any node within a player's social network.

[0055] As shown in FIG. 2, Player 201 has direct connections with several friends. When Player 201 has a direct connection with another individual, that connection is referred to as a first-degree friend. In social network 250, Player 201 has two first-degree friends. That is, Player 201 is directly connected to Friend 1<sub>1</sub> 211 and Friend 2<sub>1</sub> 221. In a social graph, it is possible for individuals to be connected to other individuals through their first-degree friends (i.e., friends of friends). As described above, each edge required to connect a player to another user is considered the degree of separation. For example, FIG. 2 shows that Player 201 has three second-degree friends to which he is connected via his connection to his first-degree friends. Second-degree Friend 1<sub>2</sub> 212 and Friend 2<sub>2</sub> 222 are connected to Player 201 via his first-degree Friend 1<sub>1</sub> 211. The limit on the depth of friend connections, or the number of degrees of separation for associations, that Player 201 is allowed is typically dictated by the restrictions and policies implemented by social networking system 120a.

[0056] In various embodiments, Player 201 can have Nth-degree friends connected to him through a chain of intermediary degree friends as indicated in FIG. 2. For example, Nth-degree Friend 1<sub>N</sub> 219 is connected to Player 201 via second-degree Friend 3<sub>2</sub> 232 and one or more other higher-degree friends. Various embodiments may take advantage of and utilize the distinction between the various degrees of friendship relative to Player 201.

[0057] In particular embodiments, a player (or player character) can have a social graph within an online multiplayer game that is maintained by the game engine and another social graph maintained by a separate social networking system. FIG. 2 depicts an example of in-game social network 260 and out-of-game social network 250. In this example, Player 201 has out-of-game connections 255 to a plurality of friends, forming out-of-game social network 250. Here, Friend 1<sub>i</sub> 211 and Friend 2<sub>i</sub> 221 are first-degree friends with Player 201 in his out-of-game social network 250. Player 201 also has in-game connections 265 to a plurality of players, forming in-game social network 260. Here, Friend 2<sub>i</sub> 221, Friend 3<sub>i</sub> 231, and Friend 4<sub>i</sub> 241 are first-degree friends with Player 201 in his in-game social network 260. In some embodiments, it is possible for a friend to be in both the out-of-game social network 250 and the in-game social network 260. Here, Friend 2<sub>i</sub> 221 has both an out-of-game connection 255 and an in-game connection 265 with Player 201, such that Friend 2<sub>i</sub> 221 is in both Player 201's in-game social network 260 and Player 201's out-of-game social network 250.

[0058] As with other social networks, Player 201 can have second-degree and higher-degree friends in both his in-game and out of game social networks. In some embodiments, it is possible for Player 201 to have a friend connected to him both in his in-game and out-of-game social networks, wherein the friend is at different degrees of separation in each network. For example, if Friend 2<sub>2</sub> 222 had a direct in-game connection with Player 201, Friend 2<sub>2</sub> 222 would be a second-degree friend in Player 201's out-of-game social network, but a first-degree friend in Player 201's in-game social network. In particular embodiments, a game engine can access in-game social network 260, out-of-game social network 250, or both.

[0059] In particular embodiments, the connections in a player's in-game social network can be formed both explicitly (e.g., users must "friend" each other) and implicitly (e.g., system observes user behaviors and "friends" users to each other). Unless otherwise indicated, reference to a friend connection between two or more players can be interpreted to cover both explicit and implicit connections, using one or more social graphs and other factors to infer friend connections. The friend connections can be unidirectional or bidirectional. It is also not a limitation of this description that two players who are deemed

"friends" for the purposes of this disclosure are not friends in real life (i.e., in disintermediated interactions or the like), but that could be the case.

[0060] FIG. 3 is a block diagram illustrating components of a game networking system, according to some example embodiments. The CR Engine, which executes on a client device 130, includes a contacts module 310, a collection module 320, a request module 330, a scoring module 340, a ranking module 350, a selection module 360 and a display module 370.

[0061] In various example embodiments, the contacts module 310 is a hardware-implemented module configured to control, manage and store information related to identifying one or more contacts of a player of at least one game provided by the game networking system 120b.

[0062] In various example embodiments, the collection module 320 is a hardware-implemented module configured to control, manage and store information related to collection contact attributes of each respective contact from one more more contact attribute sources.

[0063] In various example embodiments, the request module 330 is a hardware-implemented module configured to control, manage and store information related to requesting and receiving one or more configuration tables from the game networking system 120b.

[0064] In various example embodiments, the scoring module 340 is a hardware-implemented module configured to control, manage and store information related to scoring each contact based on the contact's collected attributes - where a score for a contact represents a likelihood that the contact will respond to a promotion associated with the configuration table.

[0065] In various example embodiments, the ranking module 350 is a hardware-implemented module configured to control, manage and store information related to ranking the scores of each contact in a contact rankings list.

[0066] In various example embodiments, the selection module 360 is a hardware-implemented module that controls, manages and stores information related to selecting a portion of contacts ("best contacts") from the contact rankings list.

[0067] In various example embodiments, the display module 370 is a hardware-implemented module that controls, manages and stores information

related to displaying a prompt for the player to select one or more best contacts included for display in the prompt. For each best contact that is selected by the player from the prompt, the promotion associated with the configuration table is send to the respective best contact.

5 [0068] FIG. 4 is a block diagram illustrating various contact attribute sources according to some example embodiments. FIG. 5 is a block diagram illustrating a configuration table applying updateable scoring rules to a plurality of contact attributes sets according to some example embodiments. FIG. 6 is a flowchart 600 showing an example method of creating a contacts ranking list, according to  
10 some example embodiments. The operations of flowchart 600 of FIG. 6 will be described below with respect to the block diagrams of FIGS. 4-5. It is understood that while FIGS. 4-5 only illustrate a first and second contact 402, 404, various embodiments can be directed to any number of contacts.

[0069] At operation 610, the CR Engine identifies a plurality of contacts  
15 402, 404 of a player of a first game associated with a game networking system. At operation 620, the CR Engine collects contact attributes 402-1... 402-3, 404-1...404-3, 409-1, 409-2, 410-1. ..410-3, 412-1. ..412-4, 417, 418-1. ..418-4 for each of the plurality of contacts from one or more contact attribute sources 400, 406, 414. For example, as illustrated in FIG. 4, the CR Engine, executing on  
20 client device 130, receives permission from the Player 201 to access a Contacts List 400 local to the client device 130, the Player's 201 account on a first social network 406 and a second social network 414, where the first and second social networks 406, 414 are external to the gaming networking system 120b and are different from each other.

25 [0070] The CR Engine accesses the Contacts List 400 and identifies at least a first contact 402 and a second contact 404. Based on the received permissions, the CR Engine collects different types of contact attributes for the first and second contacts 402, 404 that are stored in the Contacts List 400. The CR Engine collects the address 402-1, the mobile number 402-2 and email address  
30 402-3 of the first contact 402 in the Contacts List 400. The CR Engine collects the address 404-1, the mobile number 404-2, email address 404-3 and work address 404-4 of the second contact 404 in the Contacts List 400.

[0071] Based on the received permissions, the CR Engine accesses the Player's 201 account 408 on the first social network 406 and identifies social

network connections 409-1, 409-2 between the account 408 and social network accounts 410, 412 that correspond to the same identity of the first and second contacts 402, 404. The connections 409-1, 409-2 may be the same or different in terms of degree.

5 [0072] The CR Engine collects contact attributes for the first contact's account 410 on the first social network 406. The CR Engine collects presence of a photo 410-1, biographical data 410-2 (such as if any of the following are listed: hometown, current city, birthday, employer, etc.), family relationship 410-3 (if any) with Player 201 and geographic data 410-4 (such as various locations  
10 tagged to data in the account 410).

[0073] The CR Engine collects contact attributes for the second contact's account 412 on the first social network 406. The CR Engine collects presence of a photo 412-1, biographical data 412-2, family relationship 412-3 (if any) with Player 201 and geographic data 412-4 (such as various locations tagged to data  
15 in the account 412).

[0074] Based on the received permissions, the CR Engine accesses the Player's 201 account 416 on the second social network 414 and identifies a social network connection 417 between the account 416 and a social network account 418 that corresponds to the identity of the second contact 404.

20 [0075] The CR Engine collects contact attributes for the second contact's account 418 on the second social network 414. The CR Engine collects education information 418-1 (such as schools attended, degrees earned), work experience 418-2 (such as current and former employer, current and former job titles), professional skills 418-3, and an indication of presence of user-provided  
25 recommendations 418-4.

[0076] At operation 630, the CR Engine requests from a server device related to the game networking system a first configuration table 506 related to the first game.

[0077] Turning now the FIG. 5, the CR Engine, executing on client device  
30 130, creates a first contact attribute set 502 for all the types of contact attributes collected for the first contact 402 and a second contact attribute set 504 for all the types of contact attributes collected for the second contact 404. The CR Engine requests a configuration table 506 from a server device associated with the game networking system 120b. The configuration table 506 is related to a

game installed on the client device 130. The configuration table is related to prompt intended to induce behavior. For example, the configuration table can be related to a promotion of the game intended to induce a user or another player to perform a desired action. The promotion can be an invitation intended to induce a user or another player to install the game, install a different game, join the game, join a different game, or make a purchase related to the game or a different game.

5  
10  
15  
[0078] At operation 640, the CR Engine scores the collected contact attributes 402-1... 402-3, 404-1... 404-3, 409-1, 409-2, 410-1...410-3, 412-1...412-4, 417, 418-1...418-4 of each contact 402, 404 of the player according to one or more updateable rules 508, 510, 512 of the first configuration table 506. The configuration table 506 includes one or more sets of updateable rules 508, 510, 512 for scoring various types of contact attributes. For example, the configuration table 506 includes updateable rules 508 for scoring different types of contact attributes found for a particular contact in a contacts list, updateable rules 510 for scoring different types of contact attributes found for a particular contact in an account on the first social network and updateable rules 512 for scoring different types of contact attributes found for a particular contact in an account on the second social network.

20  
25  
30  
[0079] For example, in order to calculate a score for the first and second contacts 402, 404, the CR Engine applies an address scoring rule, a mobile number scoring rule and an email address scoring rule from the updateable contact list scoring rules 508 to the first contact attribute set 502 and the second attribute set 504 based on both the first and second contacts 402, 404 having an address 402-1, 404-1, a mobile number 402-2, 404-2, and an email address 402-3, 404-3 listed in the contact's list 400 on the client device 130. In addition, the CR Engine applies a work address scoring rule from the updateable contact list scoring rules 508 to the second attribute set 504 based on the second contact 404 having a work address 404-4 listed in the contact's list 400 on the client device 130.

[0080] To continue calculating a score for the first and second contacts 402, 404, the CR Engine applies a photo scoring rule, a biographical data scoring rule, a family relationship scoring rule, a geographical data scoring rule and a social network connection proximity scoring rule from the updateable first social

network scoring rules 510 to the first contact attribute set 502 and the second attribute set 504. For example, the CR Engine adjusts a respective score for the first and second contacts 402, 404 based on presence of a profile photo 410-1, 412-1. The CR Engine further adjusts the respective score for the first and second contacts 402, 404 based on the type of family relationship 410-3, 412-3 the first and second contacts 402, 404 have with the player. The CR Engine further adjusts the respective score for the first and second contacts 402, 404 based on any type (and/or amount) of geographical tags 410-4, 412-2 (such as, for example, location-based "check-ins") the first and second contacts 402, 404 have applied to various sections in their accounts 410, 412. The CR Engine further adjusts the respective score for the first and second contacts 402, 404 based on a proximity of a connection 409- 1, 409-2 between a respective account 410, 412 and the player's account 408.

**[0081]** To continue calculating a score for the second contacts 404, the CR Engine applies an education scoring rule, a work experience scoring rule, a skills scoring rule and a recommendations scoring rule from the updateable second social network scoring rules 512 to the second attribute set 504. For example, the CR Engine adjusts the respective score for the second contact 404 based on presence (and/or type, amount) of education 418-1 listed in the second contact's account 418. The CR Engine further adjusts the respective score for the second contact 404 based on presence (and/or type, amount) of work experience 418-2 listed in the second contact's account 418. The CR Engine further adjusts the respective score for the second contact 404 based on presence (and/or type, amount) of professional skills 418-3 listed in the second contact's account 418. The CR Engine further adjusts the respective score for the second contact 404 based on presence (and/or type, amount) of recommendations 418-4 from other users of the second social network 414 posted on the second contact's account 418. The CR Engine further adjusts the respective score for the second contacts 404 based on a proximity (and/or type) of a connection 417 between the second contact's account 418 and the player's account 416 on the second social network 414.

**[0082]** At operation 650, the CR Engine ranks in a contacts ranking list each contact based on a respective score. Based on the score for the first and second

contacts 402, 404 (as well as other contacts not illustrated in FIGS. 4 & 5), the CR Engine determines a list of ranked scores.

[0083] At operation 660, the CR Engine selects a portion of the contacts ranking The CR Engine selects a portion of the list of ranked scores, such as, for  
5 example, where the CR Engine has ranked the scores of twenty contacts, then selects contacts with the five highest ranked scores.

[0084] At operation 670, the CR Engine displays on the client device a selectable instance of each contact included in the selected portion of the contacts ranking list. The CR Engine displays a prompt on the client device  
10 that is based on the selected contacts. As shown in FIG. 5, the CR Engine has scored and ranked the second contact 404 higher than the first contact 402 and the CR Engine has selected the second contact 404 to be included in a graphical prompt that prompts the user to send a promotion or invitation to the second contact.

[0085] In another embodiment, the configuration table 506 includes previously-calculated scores for a contact (hereinafter "previously-responsive contact") that ultimately responded to a promotion. When the CR Engine scores a plurality of contacts and identifies a particular contact as having similar contact attributes as the previously-responsive contact, and/or having a score within a  
15 threshold range of the score calculated for the previously-responsive contact, the CR Engine automatically include the particular contact as a selectable instance in the graphical prompt.

#### Data Flow

[0086] FIG. 7 illustrates an example data flow between the components of system 700. In particular embodiments, system 700 can include client system 730, social networking system 120a (i.e. social network system), and game networking system 120b (i.e. online game system system). The components of system 700 can be connected to each other in any suitable configuration, using  
25 any suitable type of connection. The components are connected directly or over any suitable network. Client system 730, social networking system 120a, and game networking system 120bb can each have one or more corresponding data stores such as local data store 735, social data store 745, and game data store 765, respectively. Social networking system 120a and game networking system

120b can also have one or more servers that can communicate with client system 730 over an appropriate network. Social networking system 120a and game networking system 120b can have, for example, one or more internet servers for communicating with client system 730 via the Internet. Similarly, social  
5 networking system 120a and game networking system 120b can have one or more mobile servers for communicating with client system 730 via a mobile network (e.g., GSM, PCS, Wi-Fi, WPAN, etc.). In some embodiments, one server communicates with client system 730 over both the Internet and a mobile network. In other embodiments, separate servers can be used.

10 [0087] Client system 730 can receive and transmit data 723 to and from game networking system 120b. This data can include, for example, webpages, messages, game inputs, game displays, HTTP packets, data requests, transaction information, updates, and other suitable data. At some other time, or at the same  
15 time, game networking system 120b can communicate data 743, 747 (e.g., game state information, game system account information, page info, messages, data requests, updates, etc.) with other networking systems, such as social networking system 120a (e.g., Facebook, Myspace, etc.). Client system 730 can also receive and transmit data 727 to and from social networking system 120a. This data can  
20 include, for example, webpages, messages, social graph information, social network displays, HTTP packets, data requests, transaction information, updates, and other suitable data.

[0088] Communication between client system 730, social networking system 120a, and game networking system 120b can occur over any appropriate electronic communication medium or network using any suitable  
25 communications protocols. For example, client system 730, as well as various servers of the systems described herein, includes Transport Control Protocol/Internet Protocol (TCP/IP) networking stacks to provide for datagram and transport functions. Of course, any other suitable network and transport layer protocols can be utilized.

30 [0089] In addition, hosts or end-systems described herein may use a variety of higher layer communications protocols, including client-server (or request-response) protocols, such as the HyperText Transfer Protocol (HTTP) and other communications protocols, such as HTTPS, FTP, SNMP, TELNET, and a number of other protocols, may be used. In some embodiments, no protocol

may be used and, instead, transfer of raw data may be utilized via TCP or User Datagram Protocol. In addition, a server in one interaction context may be a client in another interaction context. In particular embodiments, the information transmitted between hosts may be formatted as HyperText Markup Language (HTML) documents. Other structured document languages or formats can be used, such as XML, and the like. Executable code objects, such as JavaScript and ActionScript, can also be embedded in the structured documents.

[0090] In some client-server protocols, such as the use of HTML over HTTP, a server generally transmits a response to a request from a client. The response comprises one or more data objects. For example, the response comprises a first data object, followed by subsequently transmitted data objects. In particular embodiments, a client request causes a server to respond with a first data object, such as an HTML page, which itself refers to other data objects. A client application, such as a browser, will request these additional data objects as it parses or otherwise processes the first data object.

[0091] In particular embodiments, an instance of an online game can be stored as a set of game state parameters that characterize the state of various in-game objects, such as, for example, player character state parameters, non-player character parameters, and virtual item parameters. In particular embodiments, game state is maintained in a database as a serialized, unstructured string of text data as a so-called Binary Large Object (BLOB). When a player accesses an online game on game networking system 120b, the BLOB containing the game state for the instance corresponding to the player can be transmitted to client system 730 for use by a client-side executed object to process. In particular embodiments, the client-side executable is a FLASH-based game, which can de-serialize the game state data in the BLOB. As a player plays the game, the game logic implemented at client system 730 maintains and modifies the various game state parameters locally. The client-side game logic is also batch game events, such as mouse clicks, and transmit these events to game networking system 120b. Game networking system 120b itself operates by retrieving a copy of the BLOB from a database or an intermediate memory cache (memcache) layer. Game networking system 120b can also de-serialize the BLOB to resolve the game state parameters and execute its own game logic based on the events in the batch file of events transmitted by the client to synchronize the game state on the

server side. Game networking system 120b then re-serializes the game state, now modified, into a BLOB and pass this to a memory cache layer for lazy updates to a persistent database.

[0092] With a client-server environment in which the online games runs, one  
5 server system, such as game networking system 120b, supports multiple client systems 730. At any given time, there may be multiple players at multiple client systems 730 all playing the same online game. In practice, the number of players playing the same game at the same time may be very large. As the game progresses with each player, multiple players provide different inputs to the  
10 online game at their respective client systems 730, and multiple client systems 730 transmit multiple player inputs and/or game events to game networking system 120b for further processing. In addition, multiple client systems 730 transmit other types of application data to game networking system 120b.

[0093] In particular embodiments, a computer-implemented game is a text-  
15 based or turn-based game implemented as a series of web pages that are generated after a player selects one or more actions to perform. The web pages are displayed in a browser client executed on client system 730. As an example and not by way of limitation, a client application downloaded to client system 730 operates to serve a set of webpages to a player. As another example and not  
20 by way of limitation, a computer-implemented game may be an animated or rendered game executable as a stand-alone application or within the context of a webpage or other structured document. In particular embodiments, the computer-implemented game is implemented using Adobe Flash-based technologies. As an example and not by way of limitation, a game may be fully or partially  
25 implemented as a SWF object that is embedded in a web page and executable by a Flash media player plug-in. In particular embodiments, one or more described webpages may be associated with or accessed by social networking system 120a. This disclosure contemplates using any suitable application for the retrieval and rendering of structured documents hosted by any suitable network-addressable  
30 resource or website.

[0094] Application event data of a game is any data relevant to the game (e.g., player inputs). In particular embodiments, each application datum has a name and a value, and the value of the application datum can change (i.e., be updated) at any time. When an update to an application datum occurs at client

system 730, either caused by an action of a game player or by the game logic itself, client system 730 informs game networking system 120b of the update. For example, if the game is a farming game with a harvest mechanic (such as Zynga FarmVille), an event can correspond to a player clicking on a parcel of land to harvest a crop. In such an instance, the application event data identifies an event or action (e.g., harvest) and an object in the game to which the event or action applies. For illustration purposes and not by way of limitation, system 700 is discussed in reference to updating a multi-player online game hosted on a network-addressable system (such as, for example, social networking system 120a or game networking system 120b), where an instance of the online game is executed remotely on a client system 1030, which then transmits application event data to the hosting system such that the remote game server synchronizes game state associated with the instance executed by the client system 1030.

**[0095]** In particular embodiment, one or more objects of a game can be represented as an Adobe Flash object. Flash manipulates vector and raster graphics, and supports bidirectional streaming of audio and video. "Flash" may mean the authoring environment, the player, or the application files. In particular embodiments, client system 1030 includes a Flash client. The Flash client is configured to receive and run Flash application or game object code from any suitable networking system (such as, for example, social networking system 120a or game networking system 120b). In particular embodiments, the Flash client runs in a browser client executed on client system 730. A player can interact with Flash objects using client system 730 and the Flash client. The Flash objects can represent a variety of in-game objects. Thus, the player performs various in-game actions on various in-game objects by make various changes and updates to the associated Flash objects. In particular embodiments, in-game actions can be initiated by clicking or similarly interacting with a Flash object that represents a particular in-game object. For example, a player can interact with a Flash object to use, move, rotate, delete, attack, shoot, or harvest an in-game object. This disclosure contemplates performing any suitable in-game action by interacting with any suitable Flash object. In particular embodiments, when the player makes a change to a Flash object representing an in-game object, the client-executed game logic updates one or more game state parameters associated with the in-game object. To ensure synchronization

between the Flash object shown to the player at client system 730, the Flash client sends the events that caused the game state changes to the in-game object to game networking system 120b. However, to expedite the processing and hence the speed of the overall gaming experience, the Flash client collects a batch of some number of events or updates into a batch file. The number of events or updates determined by the Flash client dynamically or determined by game networking system 120b based on server loads or other factors. For example, client system 730 sends a batch file to game networking system 120b whenever 50 updates have been collected or after a threshold period of time, such as every minute.

[0096] As used herein, the term "application event data" may refer to any data relevant to a computer-implemented game application that affects one or more game state parameters, including, for example and without limitation, changes to player data or metadata, changes to player social connections or contacts, player inputs to the game, and events generated by the game logic. In particular embodiments, each application datum has a name and a value. The value of an application datum may change at any time in response to the game play of a player or in response to the game engine (e.g., based on the game logic). In particular embodiments, an application data update occurs when the value of a specific application datum is changed. In particular embodiments, each application event datum includes an action or event name and a value (such as an object identifier). Thus, each application datum is represented as a name-value pair in the batch file. The batch file includes a collection of name-value pairs representing the application data that have been updated at client system 730. In particular embodiments, the batch file is a text file and the name-value pairs in string format.

[0097] In particular embodiments, when a player plays an online game on client system 730, game networking system 120b serializes all the game-related data, including, for example and without limitation, game states, game events, user inputs, for this particular user and this particular game into a BLOB and stores the BLOB in a database. The BLOB is associated with an identifier that indicates that the BLOB contains the serialized game-related data for a particular player and a particular online game. In particular embodiments, while a player is not playing the online game, the corresponding BLOB is stored in the database.

This enables a player to stop playing the game at any time without losing the current state of the game the player is in. When a player resumes playing the game next time, game networking system 120b retrieves the corresponding BLOB from the database to determine the most-recent values of the game-related data. In particular embodiments, while a player is playing the online game, game networking system 120b may also load the corresponding BLOB into a memory cache so that the game system may have faster access to the BLOB and the game-related data contained therein.

#### 10 Systems and Methods

[0098] In particular embodiments, one or more described webpages are associated with a networking system or networking service. However, alternate embodiments may have applications for the retrieval and rendering of structured documents hosted by any type of network addressable resource or web site. Additionally, as used herein, a user may be an individual, a group, or an entity (such as a business or third party application).

[0099] FIG. 8 illustrates an example computing system architecture used to implement a server 922 or a client system 930 illustrated in FIG. 9. In one embodiment, hardware system 800 comprises a processor 802, a cache memory 804, and one or more executable modules and drivers, stored on a tangible computer readable medium, directed to the functions described herein. Additionally, hardware system 800 includes a high performance input/output (I/O) bus 806 and a standard I/O bus 808. A host bridge 810 couples processor 802 to high performance I/O bus 806, whereas I/O bus bridge 812 couples the two buses 806 and 808 to each other. A system memory 814 and one or more network/communication interfaces 816 coupled to bus 806. Hardware system 800 further includes video memory (not shown) and a display device coupled to the video memory. Mass storage 818 and I/O ports 820 are coupled to bus 808. Hardware system 800 optionally includes a keyboard, a pointing device, and a display device (not shown) coupled to bus 808. Collectively, these elements are intended to represent a broad category of computer hardware systems, including but not limited to general purpose computer systems based on the x86-compatible processors manufactured by Intel Corporation of Santa Clara, California, and the x86-compatible processors manufactured by Advanced Micro

Devices (AMD), Inc., of Sunnyvale, California, as well as any other suitable processor.

[0100] The elements of hardware system 800 are described in greater detail below. In particular, network interface 816 provides communication  
5 between hardware system 800 and any of a wide range of networks, such as an Ethernet (e.g., IEEE 802.3) network, a backplane, etc. Mass storage 818 provides permanent storage for the data and programming instructions to perform the above-described functions implemented in servers 422, whereas system memory 814 (e.g., DRAM) provides temporary storage for the data and  
10 programming instructions when executed by processor 802. I/O ports 820 are one or more serial and/or parallel communication ports that provide communication between additional peripheral devices, which is coupled to hardware system 800.

[0101] Hardware system 800 may include a variety of system  
15 architectures and various components of hardware system 800 may be rearranged. For example, cache 804 is on-chip with processor 802. Alternatively, cache 804 and processor 802 may be packed together as a "processor module," with processor 802 being referred to as the "processor core." Furthermore, certain embodiments of the present disclosure may not require nor include all of  
20 the above components. For example, the peripheral devices shown coupled to standard I/O bus 808 may couple to high performance I/O bus 806. In addition, in some embodiments, only a single bus may exist, with the components of hardware system 800 being coupled to the single bus. Furthermore, hardware system 800 may include additional components, such as additional processors,  
25 storage devices, or memories.

[0102] An operating system manages and controls the operation of hardware system 800, including the input and output of data to and from software applications (not shown). The operating system provides an interface between the software applications being executed on the system and the  
30 hardware components of the system. Any suitable operating system may be used, such as the LINUX Operating System, the Apple Macintosh Operating System, available from Apple Computer Inc. of Cupertino, Calif, UNIX operating systems, Microsoft (r) Windows(r) operating systems, BSD operating systems, and the like. Of course, other embodiments are possible. For example, the

functions described herein may be implemented in firmware or on an application-specific integrated circuit. Particular embodiments may operate in a wide area network environment, such as the Internet, including multiple network addressable systems.

5 [0103] FIG. 9 illustrates an example network environment, in which various example embodiments may operate. Network cloud 960 generally represents one or more interconnected networks, over which the systems and hosts described herein can communicate. Network cloud 960 includes packet-based wide area networks (such as the Internet), private networks, wireless  
10 networks, satellite networks, cellular networks, paging networks, and the like. As FIG. 9 illustrates, particular embodiments operate in a network environment comprising one or more networking systems, such as social networking system 120a, game networking system 120b, and one or more client systems 930. The components of social networking system 120a and game networking system  
15 120b operate analogously; as such, hereinafter they may be referred to simply at networking system 920. Client systems 930 are operably connected to the network environment via a network service provider, a wireless carrier, or any other suitable means.

[0104] Networking system 920 is a network addressable system that, in  
20 various example embodiments, comprises one or more physical servers 922 and data stores 924. The one or more physical servers 922 are operably connected to computer network 960 via, by way of example, a set of routers and/or networking switches 926. In an example embodiment, the functionality hosted by the one or more physical servers 922 include web or HTTP servers, FTP  
25 servers, as well as, without limitation, webpages and applications implemented using Common Gateway Interface (CGI) script, PHP Hyper-text Preprocessor (PHP), Active Server Pages (ASP), Hyper Text Markup Language (HTML), Extensible Markup Language (XML), Java, JavaScript, Asynchronous JavaScript and XML (AJAX), Flash, ActionScript, and the like.

30 [0105] Physical servers 922 may host functionality directed to the operations of networking system 920. Hereinafter servers 922 may be referred to as server 922, although server 922 may include numerous servers hosting, for example, networking system 920, as well as other content distribution servers, data stores, and databases. Data store 924 may store content and data relating to,

and enabling, operation of networking system 920 as digital data objects. A data object, in particular embodiments, is an item of digital information typically stored or embodied in a data file, database, or record. Content objects may take many forms, including: text (e.g., ASCII, SGML, HTML), images (e.g., jpeg, tif and gif), graphics (vector-based or bitmap), audio, video (e.g., mpeg), or other multimedia, and combinations thereof. Content object data may also include executable code objects (e.g., games executable within a browser window or frame), podcasts, etc. Logically, data store 924 corresponds to one or more of a variety of separate and integrated databases, such as relational databases and object-oriented databases, that maintain information as an integrated collection of logically related records or files stored on one or more physical systems. Structurally, data store 924 may generally include one or more of a large class of data storage and management systems. In particular embodiments, data store 924 may be implemented by any suitable physical system(s) including components, such as one or more database servers, mass storage media, media library systems, storage area networks, data storage clouds, and the like. In one example embodiment, data store 924 includes one or more servers, databases (e.g., MySQL), and/or data warehouses. Data store 924 may include data associated with different networking system 920 users and/or client systems 930.

20 **[0106]** Client system 930 is generally a computer or computing device including functionality for communicating (e.g., remotely) over a computer network. Client system 930 may be a desktop computer, laptop computer, personal digital assistant (PDA), in- or out-of-car navigation system, smart phone or other cellular or mobile phone, or mobile gaming device, among other suitable computing devices. Client system 930 may execute one or more client applications, such as a web browser (e.g., Microsoft Internet Explorer, Mozilla Firefox, Apple Safari, Google Chrome, and Opera), to access and view content over a computer network. In particular embodiments, the client applications allow a user of client system 930 to enter addresses of specific network resources to be retrieved, such as resources hosted by networking system 920. These addresses can be Uniform Resource Locators (URLs) and the like. In addition, once a page or other resource has been retrieved, the client applications may provide access to other pages or records when the user "clicks" on hyperlinks to other resources. By way of example, such hyperlinks may be located within the

webpages and provide an automated way for the user to enter the URL of another page and to retrieve that page.

[0107] A webpage or resource embedded within a webpage, which may itself include multiple embedded resources, may include data records, such as plain textual information, or more complex digitally encoded multimedia content, such as software programs or other code objects, graphics, images, audio signals, videos, and so forth. One prevalent markup language for creating webpages is the Hypertext Markup Language (HTML). Other common web browser-supported languages and technologies include the Extensible Markup Language (XML), the Extensible Hypertext Markup Language (XHTML), JavaScript, Flash, ActionScript, Cascading Style Sheet (CSS), and, frequently, Java. By way of example, HTML enables a page developer to create a structured document by denoting structural semantics for text and links, as well as images, web applications, and other objects that can be embedded within the page. Generally, a webpage may be delivered to a client as a static document; however, through the use of web elements embedded in the page, an interactive experience may be achieved with the page or a sequence of pages. During a user session at the client, the web browser interprets and displays the pages and associated resources received or retrieved from the website hosting the page, as well as, potentially, resources from other websites.

[0108] When a user at a client system 930 desires to view a particular webpage (hereinafter also referred to as target structured document) hosted by networking system 920, the user's web browser, or other document rendering engine or suitable client application, formulates and transmits a request to networking system 920. The request generally includes a URL or other document identifier as well as metadata or other information. By way of example, the request may include information identifying the user, such as a user ID, as well as information identifying or characterizing the web browser or operating system running on the user's client computing device 930. The request may also include location information identifying a geographic location of the user's client system or a logical network location of the user's client system. The request may also include a timestamp identifying when the request was transmitted.

[0109] Although the example network environment described above and illustrated in FIG. 9 described with respect to social networking system 120a and game networking system 120b, this disclosure encompasses any suitable network environment using any suitable systems. As an example and not by way of limitation, the network environment may include online media systems, online reviewing systems, online search engines, online advertising systems, or any combination of two or more such systems.

[0110] Furthermore, the above-described elements and operations can be comprised of instructions that are stored on non-transitory storage media. The instructions can be retrieved and executed by a processing system. Some examples of instructions are software, program code, and firmware. Some examples of non-transitory storage media are memory devices, tape, disks, integrated circuits, and servers. The instructions are operational when executed by the processing system to direct the processing system to operate in accord with the disclosure. The term "processing system" refers to a single processing device or a group of inter-operational processing devices. Some examples of processing devices are integrated circuits and logic circuitry. Those skilled in the art are familiar with instructions, computers, and storage media.

#### 20 Miscellaneous

[0111] One or more features from any embodiment may be combined with one or more features of any other embodiment without departing from the scope of the disclosure.

[0112] A recitation of "a", "an," or "the" is intended to mean "one or more" unless specifically indicated to the contrary. In addition, it is to be understood that functional operations, such as "awarding", "locating", "permitting" and the like, are executed by game application logic that accesses, and/or causes changes to, various data attribute values maintained in a database or other memory.

[0113] The present disclosure encompasses all changes, substitutions, variations, alterations, and modifications to the example embodiments herein that a person having ordinary skill in the art would comprehend. Similarly, where appropriate, the appended claims encompass all changes, substitutions,

variations, alterations, and modifications to the example embodiments herein that a person having ordinary skill in the art would comprehend.

[0114] For example, the methods, game features and game mechanics described herein may be implemented using hardware components, software components, and/or any combination thereof. By way of example, while  
5 embodiments of the present disclosure have been described as operating in connection with a networking website, various embodiments of the present disclosure can be used in connection with any communications facility that supports web applications. Furthermore, in some embodiments the term "web  
10 service" and "website" may be used interchangeably and additionally may refer to a custom or generalized API on a device, such as a mobile device (e.g., cellular phone, smart phone, personal GPS, personal digital assistance, personal gaming device, etc.), that makes API calls directly to a server. Still further, while the embodiments described above operate with business-related virtual  
15 objects (such as stores and restaurants), the invention can be applied to any in-game asset around which a harvest mechanic is implemented, such as a virtual stove, a plot of land, and the like. The specification and drawings are, accordingly, to be regarded in an illustrative rather than a restrictive sense. It will, however, be evident that various modifications and changes may be made  
20 thereunto without departing from the broader spirit and scope of the disclosure as set forth in the claims and that the disclosure is intended to cover all modifications and equivalents within the scope of the following claims.

CLAIMS

What is claimed is:

- 5 1. A method comprising:
- identifying, by a client device, a plurality of contacts of a player of a first game associated with a game networking system;
- collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources;
- 10 requesting, from a server device related to the game networking system, a first configuration table related to the first game;
- scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table;
- ranking in a contacts ranking list each contact based on a respective
- 15 score;
- selecting a portion of the contacts ranking list; and
- displaying on the client device a selectable instance of each contact included in the selected portion of the contacts ranking list.
- 20 2. The method as in claim 1, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources comprises:
- receiving permission from the player to access a contacts listing local to the client device;
- receiving permission from the player to access an account of the player in
- 25 a first social network system external to the game networking system of the first game; and
- receiving permission from the player to access an account of the player in a second social network system external to the game networking system of the first game, the first social network system different than the second social
- 30 network system.

3. The method as in claim 2, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources further comprises:

5 identifying a first contact in the contacts listing;

collecting a first instance of a first type of contact attribute of the first contact from the contact listing;

10 identifying a connection between the account of the player in the first social network system and an account of the first contact in the first social network system;

collecting a first instance of a second type of contact attribute from the account of the first contact in the first social network system; and

15 including the first instance of the first type of contact attribute and the first instance of the second type of contact attribute in a contact attribute set for the first contact.

4. The method as in claim 3, wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

20 applying a scoring rule from the first configuration table for each type of contact attribute in the contact attribute set for the first contact to calculate a score for the first contact;

25 determining the score for the first contact is within a threshold range of a previously-calculated score for another contact that responded to a first promotion associated with the first game; and

including the first contact in the portion of the contacts ranking list for display of the first contact on the client device.

5. The method as in claim 3, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources further comprises:

identifying a second contact in the contacts listing;

5 collecting a second instance of the first type of contact attribute of the second contact from the contact listing;

identifying a connection between the account of the player in the first social network system and an account of the second contact in the first social network system;

10 collecting a second instance of a second type of contact attribute from the account of the second contact in the first social network system;

collecting a first instance of a third type of contact attribute from the account of the second contact in the first social network system; and

15 including the second instance of the first type of contact attribute, the second instance of the second type of contact attribute and the first instance of the third type of contact attribute in a contact attribute set for the second contact.

6. The method as in claim 5, wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

20 identifying a connection between the account of the player in the second social network system and an account of the second contact in the second social network system;

25 collecting a first instance of a fourth type of contact attribute from the account of the second contact in the second social network system; and

including the first instance of the fourth type of contact attribute in the contact attribute set for the second contact.

7. The method as in claim 6, wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

calculating respective scores for the first and second contacts based on:

5 applying a first scoring rule for the first type of contact attribute in the contact attribute sets for the first and second contacts;

applying a second scoring rule for the second type of contact attribute in the contact attribute sets for the first and second contacts;

10 applying a third scoring rule for the third type of contact attribute in the contact attribute set for the second contact; and

applying a fourth scoring rule for the fourth type of contact attribute in the contact attribute set for the second contact.

8. A machine-readable storage medium storing instructions which, when  
15 executed by one or more processors, cause the one or more processors to perform operations comprising:

identifying a plurality of contacts of a player of a first game associated with a game networking system;

20 collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources;

requesting a first configuration table related to the first game;

scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table;

25 ranking in a contacts ranking list each contact based on a respective score;

selecting a portion of the contacts ranking list; and

displaying a selectable instance of each contact included in the selected portion of the contacts ranking list.

9. The machine-readable storage medium as in claim 8, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources comprises:

receiving permission from the player to access a local contacts listing;

5 receiving permission from the player to access an account of the player in an external first social network system; and

receiving permission from the player to access an account of the player in an external second social network system, the first social network system different than the second social network system.

10

10. The machine-readable storage medium as in claim 9, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources further comprises:

identifying a first contact in the contacts listing;

15 collecting a first instance of a first type of contact attribute of the first contact from the contact listing;

identifying a connection between the account of the player in the first social network system and an account of the first contact in the first social network system;

20 collecting a first instance of a second type of contact attribute from the account of the first contact in the first social network system; and

including the first instance of the first type of contact attribute and the first instance of the second type of contact attribute in a contact attribute set for the first contact.

25

11. The machine-readable storage medium as in claim 10, wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

applying a scoring rule from the first configuration table for each type of contact attribute in the contact attribute set for the first contact to calculate a score for the first contact;

5 determining the score for the first contact is within a threshold range of a previously-calculated score for another contact that responded to a first promotion associated with the first game; and

including the first contact in the portion of the contacts ranking list for display of the first contact on a client device.

10 12. The machine-readable storage medium as in claim 10, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources further comprises:

identifying a second contact in the contacts listing;

15 collecting a second instance of the first type of contact attribute of the second contact from the contact listing;

identifying a connection between the account of the player in the first social network system and an account of the second contact in the first social network system;

20 collecting a second instance of a second type of contact attribute from the account of the second contact in the first social network system;

collecting a first instance of a third type of contact attribute from the account of the second contact in the first social network system; and

25 including the second instance of the first type of contact attribute, the second instance of the second type of contact attribute and the first instance of the third type of contact attribute in a contact attribute set for the second contact.

13. The machine-readable storage medium as in claim 12, wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

identifying a connection between the account of the player in the second social network system and an account of the second contact in the second social network system;

5 collecting a first instance of a fourth type of contact attribute from the account of the second contact in the second social network system; and

including the first instance of the fourth type of contact attribute in the contact attribute set for the second contact.

14. The machine-readable storage medium as in claim 13, wherein scoring  
10 the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

calculating respective scores for the first and second contacts based on:

applying a first scoring rule for the first type of contact attribute in the contact attribute sets for the first and second contacts;

15 applying a second scoring rule for the second type of contact attribute in the contact attribute sets for the first and second contacts;

applying a third scoring rule for the third type of contact attribute in the contact attribute set for the second contact; and

20 applying a fourth scoring rule for the fourth type of contact attribute in the contact attribute set for the second contact.

15. A computer system comprising:

a processor;

25 a memory device holding an instruction set executable on the processor to cause the computer system to perform operations comprising:

identifying a plurality of contacts of a player of a first game associated with a game networking system;

collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources;

- requesting a first configuration table related to the first game;
- scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table;
- ranking in a contacts ranking list each contact based on a respective
- 5 score;
- selecting a portion of the contacts ranking list; and
- displaying a selectable instance of each contact included in the selected portion of the contacts ranking list on the computer system.
- 10 16. The computer system as in claim 15, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources comprises:
- receiving permission from the player to access a local contacts listing;
- receiving permission from the player to access an account of the player in
- 15 an external first social network system; and
- receiving permission from the player to access an account of the player in an external second social network system, the first social network system different than the second social network system.
- 20 17. The computer system as in claim 16, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources further comprises:
- identifying a first contact in the contacts listing;
- collecting a first instance of a first type of contact attribute of the first
- 25 contact from the contact listing;
- identifying a connection between the account of the player in the first social network system and an account of the first contact in the first social network system;
- collecting a first instance of a second type of contact attribute from the
- 30 account of the first contact in the first social network system; and

including the first instance of the first type of contact attribute and the first instance of the second type of contact attribute in a contact attribute set for the first contact.

- 5 18. The computer system as in claim 17, wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

applying a scoring rule from the first configuration table for each type of contact attribute in the contact attribute set for the first contact to calculate a  
10 score for the first contact;

determining the score for the first contact is within a threshold range of a previously-calculated score for another contact that responded to a first promotion associated with the first game; and

15 including the first contact in the portion of the contacts ranking list for display of the first contact on the computer system.

19. The computer system as in claim 17, wherein collecting contact attributes for each of the plurality of contacts from one or more contact attribute sources further comprises:

20 identifying a second contact in the contacts listing;

collecting a second instance of the first type of contact attribute of the second contact from the contact listing;

25 identifying a connection between the account of the player in the first social network system and an account of the second contact in the first social network system;

collecting a second instance of a second type of contact attribute from the account of the second contact in the first social network system;

collecting a first instance of a third type of contact attribute from the account of the second contact in the first social network system; and

including the second instance of the first type of contact attribute, the second instance of the second type of contact attribute and the first instance of the third type of contact attribute in a contact attribute set for the second contact.

5 20. The computer system as in claim 19, wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

10 identifying a connection between the account of the player in the second social network system and an account of the second contact in the second social network system;

collecting a first instance of a fourth type of contact attribute from the account of the second contact in the second social network system; and

including the first instance of the fourth type of contact attribute in the contact attribute set for the second contact; and

15 wherein scoring the collected contact attributes of each contact of the player according to one or more updateable rules of the first configuration table comprises:

calculating respective scores for the first and second contacts based on:

20 applying a first scoring rule for the first type of contact attribute in the contact attribute sets for the first and second contacts;

applying a second scoring rule for the second type of contact attribute in the contact attribute sets for the first and second contacts;

25 applying a third scoring rule for the third type of contact attribute in the contact attribute set for the second contact; and

applying a fourth scoring rule for the fourth type of contact attribute in the contact attribute set for the second contact.

30

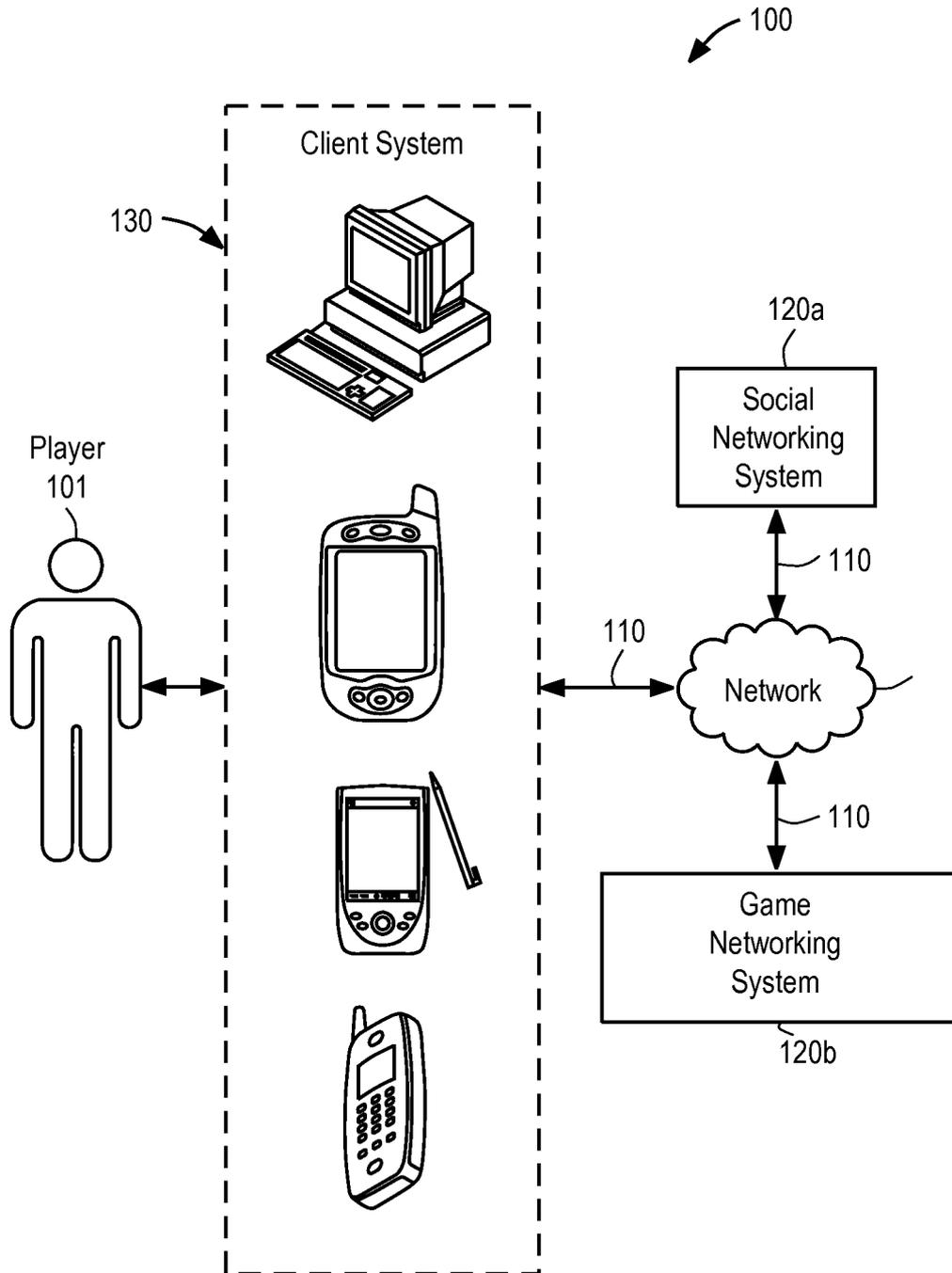


FIG. 1

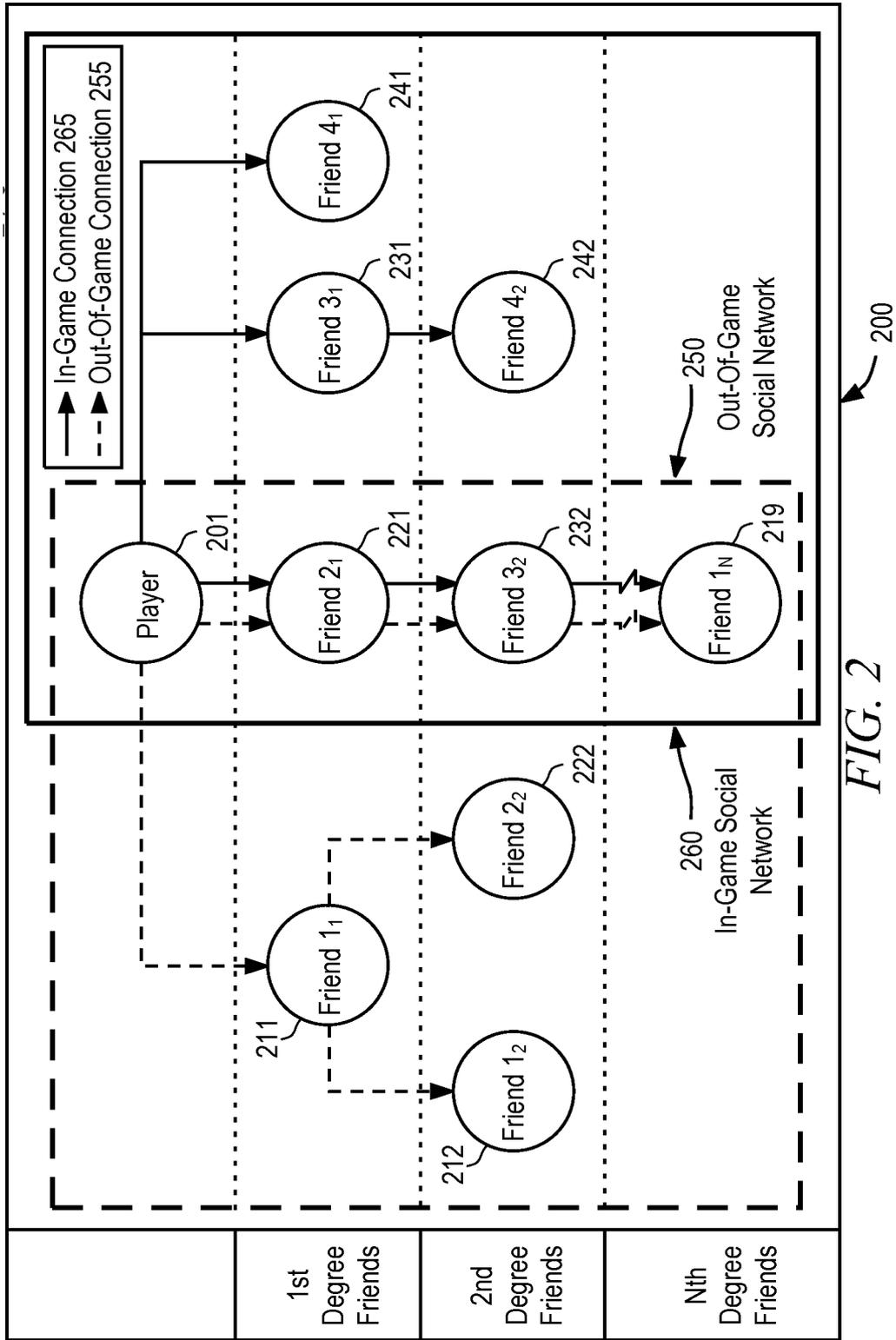


FIG. 2

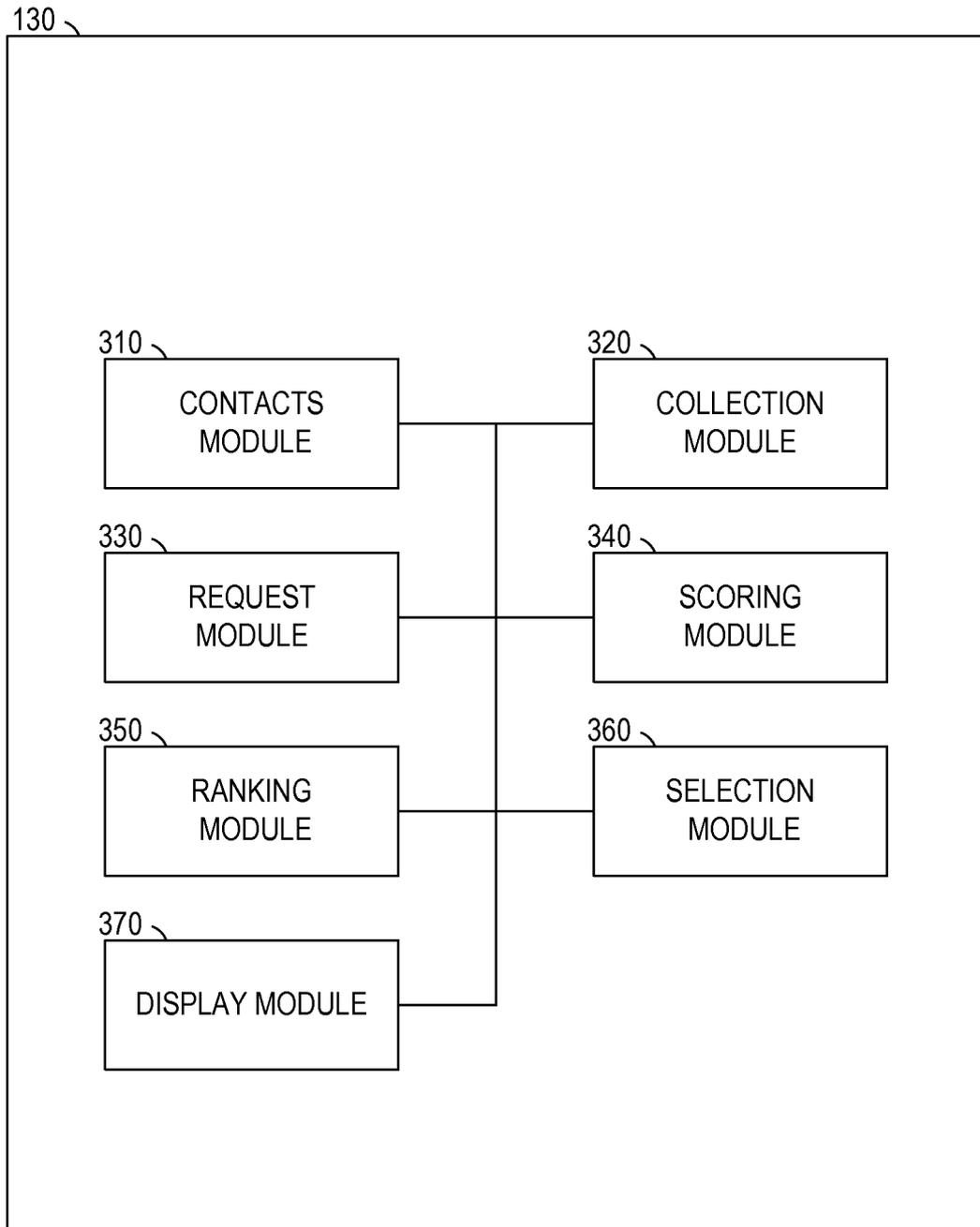


FIG. 3

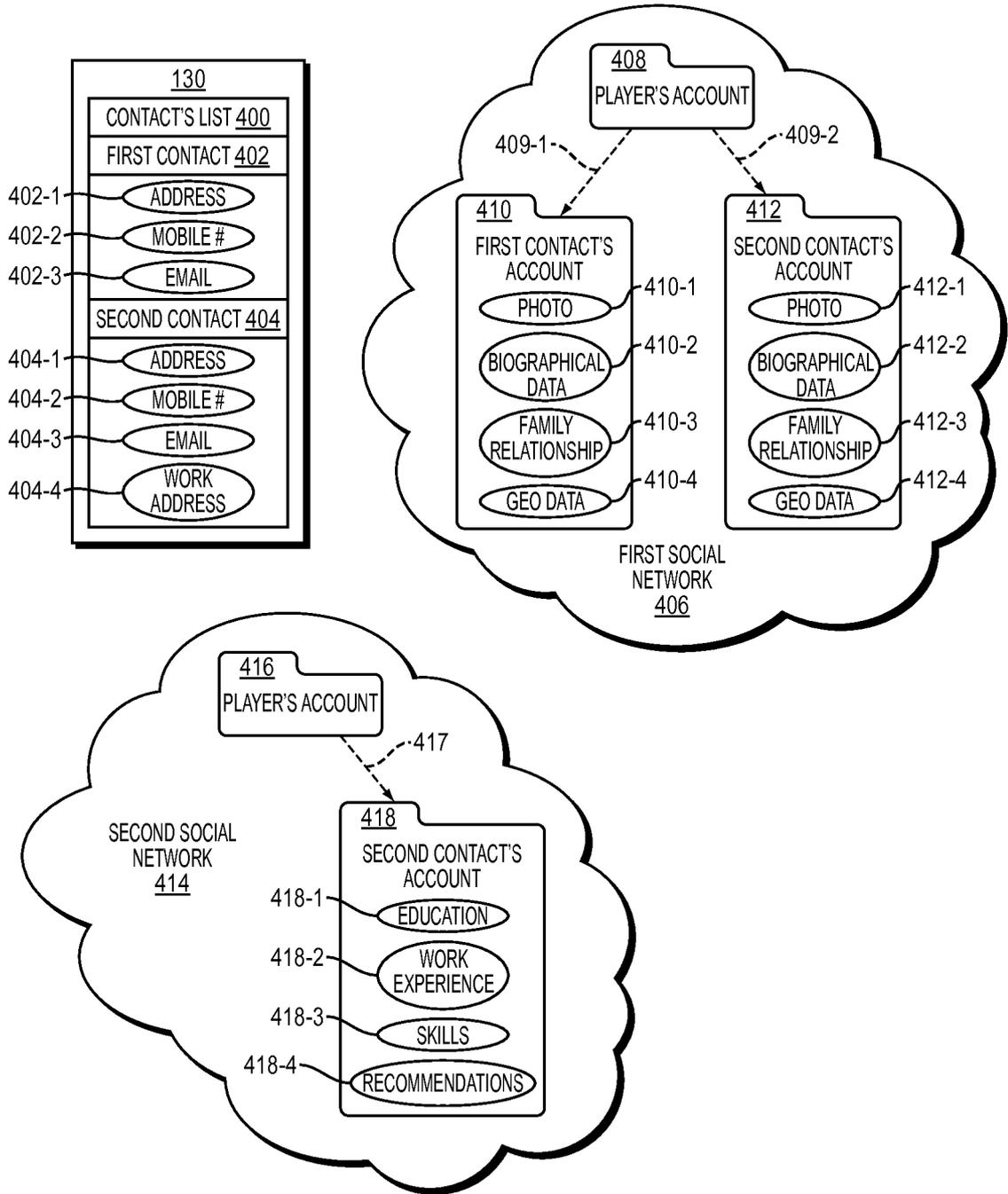


FIG. 4

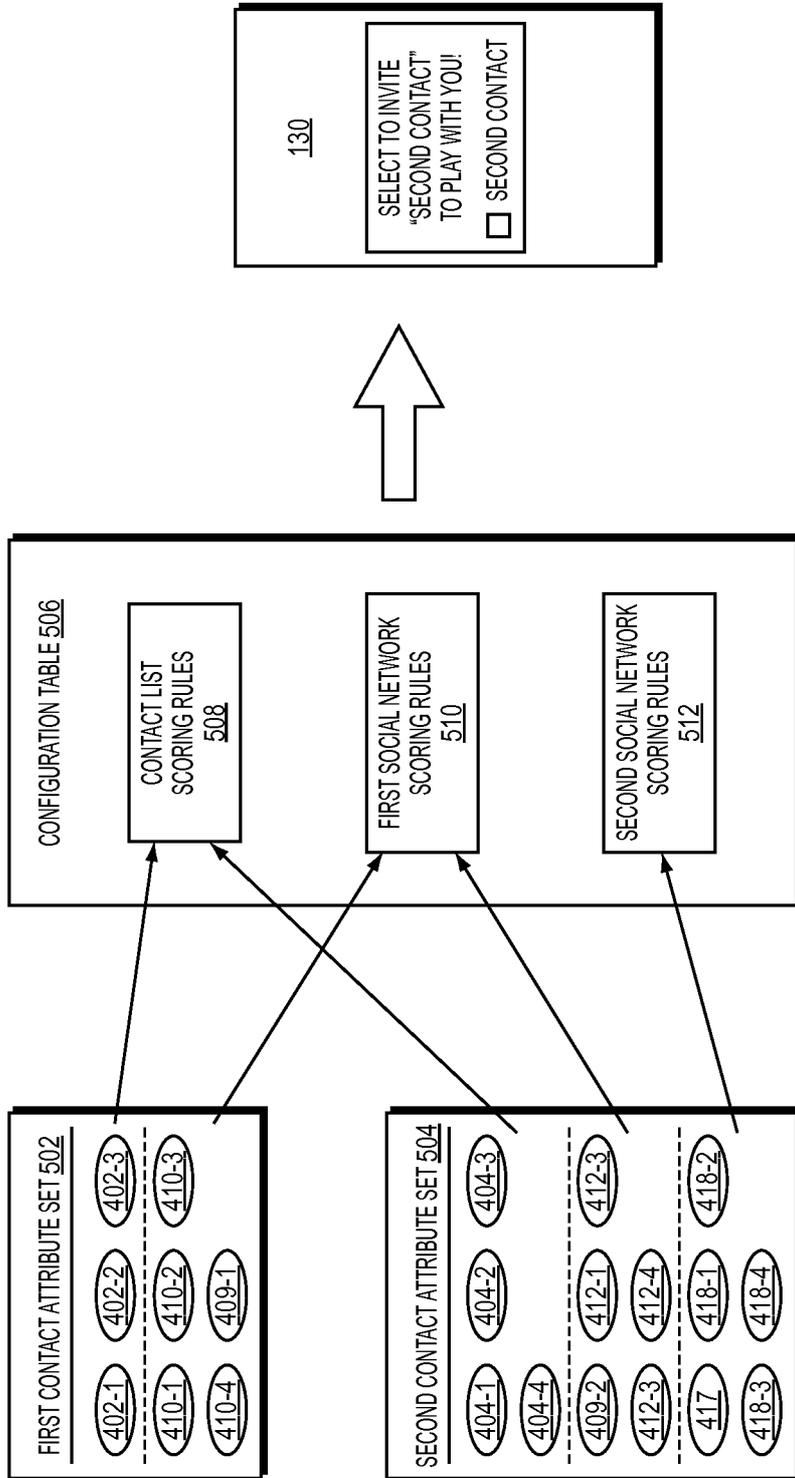


FIG. 5

6/9

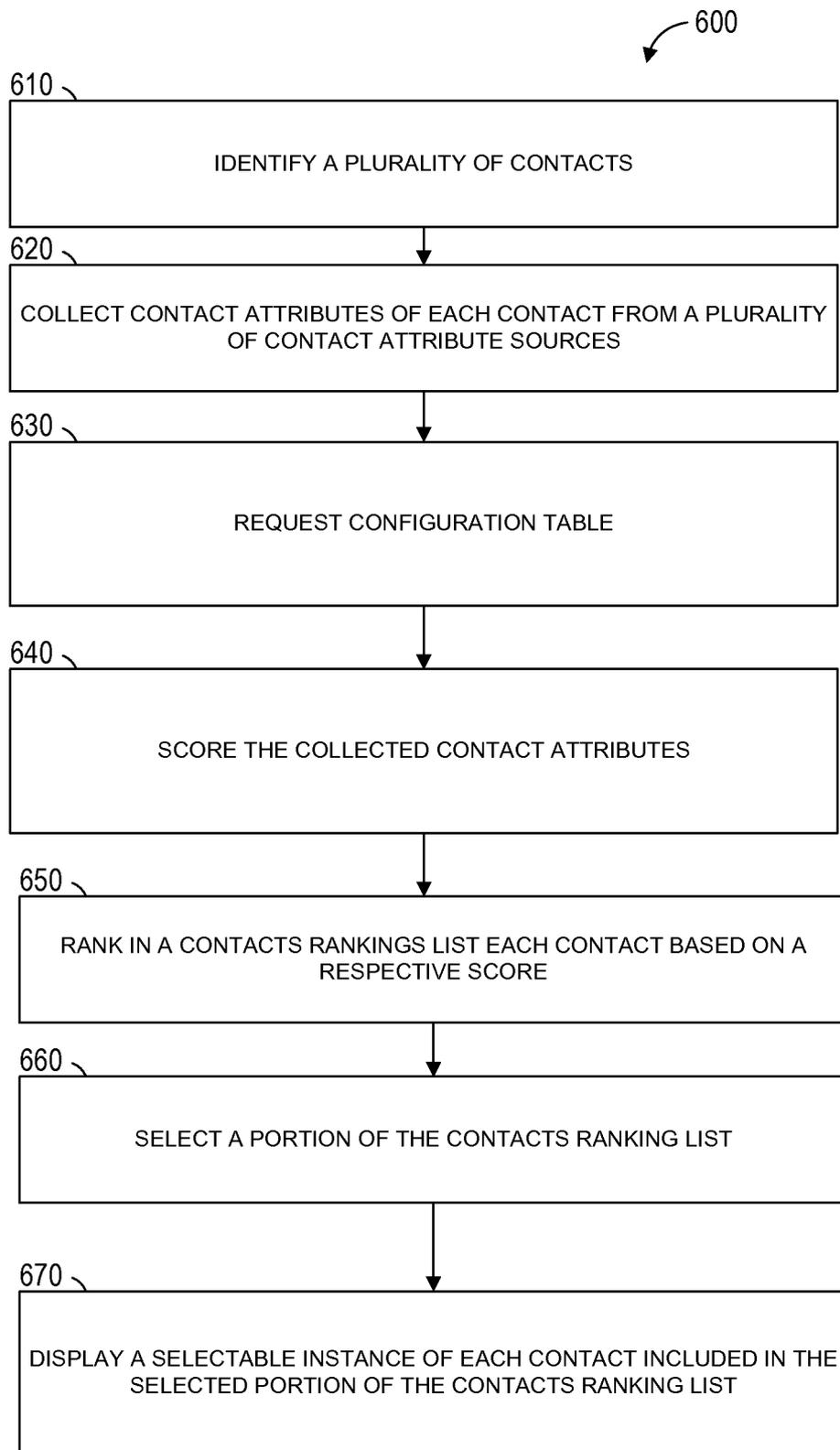


FIG. 6

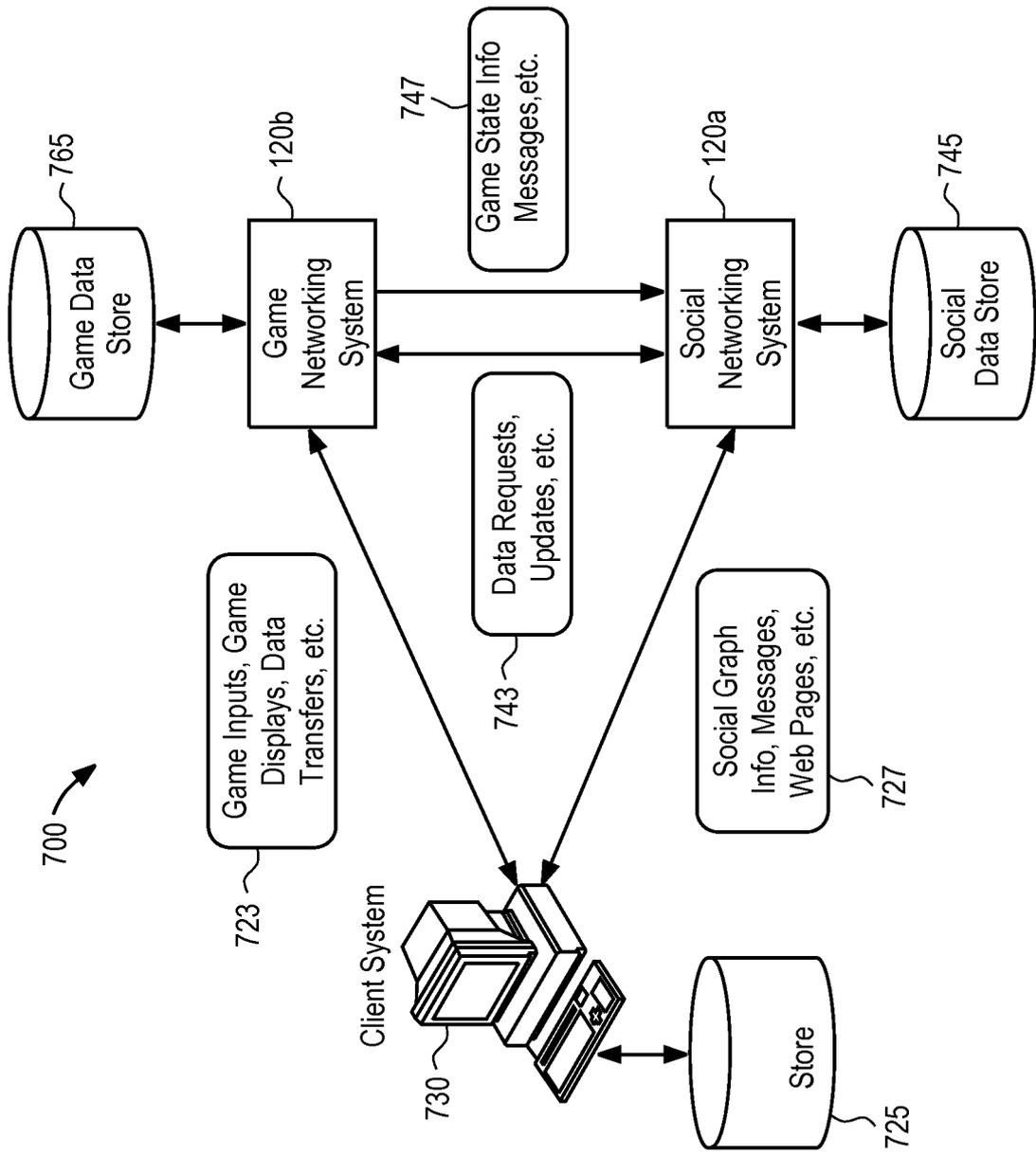


FIG. 7 725

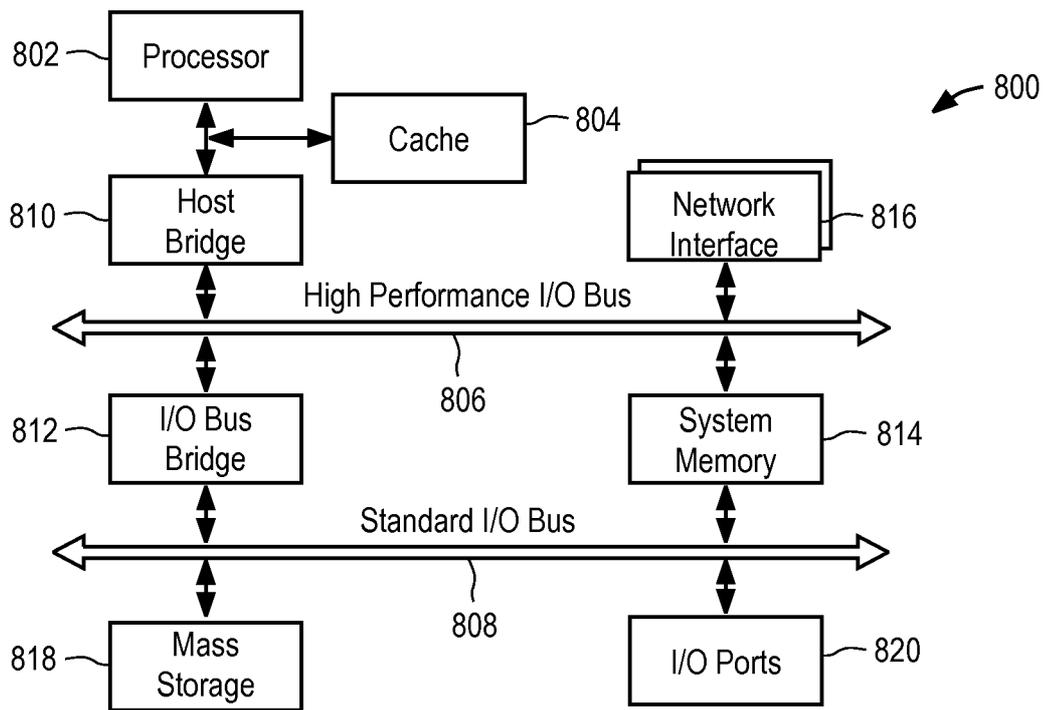


FIG. 8

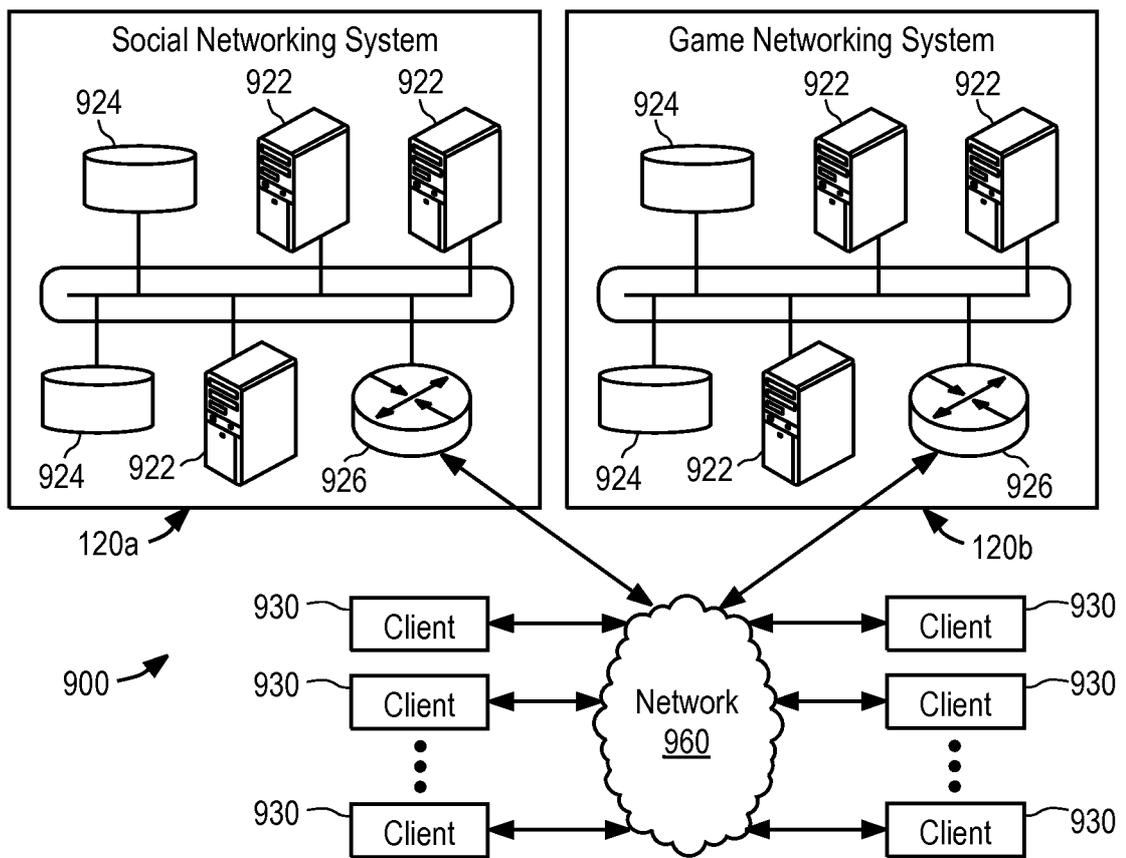


FIG. 9

**INTERNATIONAL SEARCH REPORT**

International application No.  
PCT/US201 5/024025

**A. CLASSIFICATION OF SUBJECT MATTER**

**IPC(8) - A63F 7/00 (2015.01 )**

**CPC - A63F 7/00 (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) - A63F 7/00, 9/00, 13/00, 13/12 (2015.01)

USPC - 463/7, 22, 42

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

CPC - A63F 7/00, 9/00, 13/00, 13/12 (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 Scholar, Google.

Search terms used: scoring, ranking, attributes, players, friends, social network

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2012/0282995 A1 (ALLEN et al) 08 November 2012 (08.11.2012) entire document	1-20
Y	US 2012/0015741 A1 (CRAINE et al) 19 January 2012 (19.01.2012) entire document	1-20
Y	US 2013/0165234 A1 (HALL et al) 27 June 2013 (27.06.2013) entire document	3-7,10-14,17-20
Y	US 2013/0006778 A1 (RAGHUNANDAN) 03 January 2013 (03.01.2013) entire document	5-7,12-14,19,20
Y	US 2013/0254283 A1 (GARCIA-MARTINEZ) 03 January 2013 (03.01.2013) entire document	7,14, 20

**H** Further documents are listed in the continuation of Box C.     

* Special categories of cited documents:	"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
"A" document defining the general state of the art which is not considered to be of particular relevance	"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
"E" earlier application or patent but published on or after the international filing date	"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
"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)	"&" document member of the same patent family
"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	

Date of the actual completion of the international search 02 June 2015	Date of mailing of the international search report <b>0 2 JUL 2015</b>
---	---

Name and mailing address of the ISA/US 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 R. Copenheaver PCT Helpdesk: 571-272-4300 PCT OSP: 571-272-7774
---	---