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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
27 December 2007 (27.12.2007)

PCT

(10) International Publication Number
WO 2007/149338 A2

(51) International Patent Classification:
H04L 29/08 (2006.01)

(21) International Application Number:
PCT/US2007/014093

(22) International Filing Date: 14 June 2007 (14.06.2007)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
11/453,800 16 June 2006 (16.06.2006) US

(71) Applicant *(for all designated States except US)*: **LUCENT TECHNOLOGIES INC.** [US/US]; 600 Mountain Avenue, Murray Hill, NJ 07974-0636 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: **CHANDRAMENON, Girish, P.** [IN/US]; 1434 Waterford Drive, Edison, NJ 08817 (US). **HAASE, Oliver** [DE/DE]; Jungerhalde 86, Konstanz, 78464 Baden-wuerttemberg (DE). **MILLER, Scott, C.** [US/US]; 19 Polo Club Drive, Freehold, NJ 07728 (US).

(74) Agent: **CURTIN, John, E.**; Lucent Technologies Inc., Docket Administrator- Room 2f-190, 600 Mountain Avenue, Murray Hill, NJ 07974 (US).

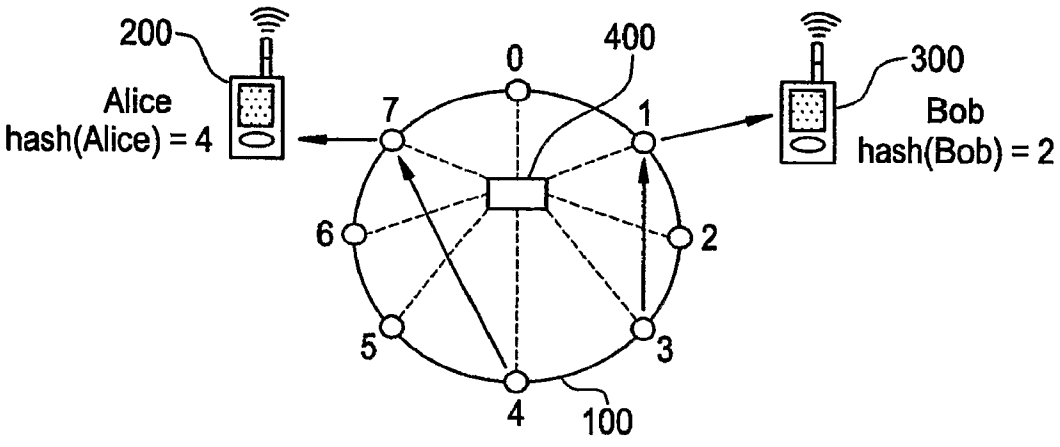
(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, SV, SY, 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, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, LV, MC, MT, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:
— *without international search report and to be republished upon receipt of that report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) **Title:** METHODS, DEVICES AND ARCHITECTURES FOR ESTABLISHING PEER-TO -PEER SESSIONS



(57) **Abstract:** Signaling paths and communication sessions, such as voice-over-IP sessions, can be established in substantially shorter and predictable time periods than previously thought possible over carrier-based, peer-to-peer networks (P2P). Instead of requiring a signaling pathway to traverse a large number of hops/nodes, novel architectures permit a signaling pathway to traverse a few designated host and anchor nodes that are part of a core of a P2P network. The novel architectures and related methods also make it possible to easily manage and maintain P2P networks.

WO 2007/149338 A2

METHODS, DEVICES AND ARCHITECTURES FOR ESTABLISHING PEER-TO-PEER SESSIONS

BACKGROUND OF THE INVENTION

[0001] Peer-to-peer networks are completely decentralized networks consisting of identical nodes, each of which can act as a "client" (in the sense of a traditional client-server network) and assume the responsibilities of a server at the same time. Because each element of a peer-to-peer (sometimes referred to as "P2P") network is the same, they are easily scalable. For example, a peer-to-peer network may grow by simply adding more nodes (e.g., telephones, computers, personal digital assistants ("PDAs")). In addition to being easily scalable it is fairly straightforward to maintain the reliability of peer-to-peer networks.

[0002] Typically, nodes or peers within a P2P network monitor each other. If a node or peer fails one or more of the remaining nodes takes over the responsibilities of the failed node. The architectural simplicity of peer-to-peer networks make them attractive for use with certain types of services, such as voice-over-Internet Protocol ("VoIP"). As issues involving the ease of configuring a network and the maintenance costs of a network grow peer-to-peer networks may provide advantages over existing carrier grid networks.

[0003] In a conventional peer-to-peer network each node or peer (the two words will be used interchangeably herein) stores attributes known as "*key, value*" pairs. For example, a *key* may indicate a user identification ("user ID") while a *value* may represent the address of a node. To establish a communication session between a source and

destination node in a conventional peer-to-peer network the address of the destination node must be determined by locating a given *key,value* pair within a node that includes the *value* (i.e., address) of the destination node using only the *key* transmitted by the source node. More specifically, each node in the network that receives the transmitted *key* compares the *key* to *key,value* pairs stored in its look-up table or the like. Assuming at least one node contains the *key,value* pair associated with the destination node, the *value* (i.e., address) of the destination is returned to the source. Thereafter, the source sets up a signaling path with the destination node. Said another way, if one user wishes to make a VoIP telephone call to another, the caller transmits the intended recipient's user ID to other users (i.e., their devices) in the peer-to-peer network. Using lookup tables and search functions, the address of the recipient is found and returned to the caller to permit the establishment of a signaling path, and eventually a communication session, between the caller and the recipient.

[0004] Early peer-to-peer networks "flooded" requests to establish sessions into the entire network or parts of it until the desired *key,value* pair was identified (i.e., until the address of the recipient of a call was located). However, flooding does not work well in large peer-to-peer networks. To overcome the disadvantages of such a technique, many recent peer-to-peer networks have begun to use distributed hash tables ("DHT") to store and retrieve *key, value* pairs.

[0005] For example, using DHTs peer-to-peer networks assign a *hashed key* to each node in the network. By assigning *hashed keys* to

nodes in the network, the time it takes to locate the address of an intended recipient of a call and the like is decreased.

[0006] In mathematical terms, it can be said that techniques which use DHTs guarantee an upper bound of *O (log N)* steps (which corresponds to a time period). In other words, the use of DHTs assures that the maximum number of steps it will take to identify a user (i.e., establish a session) within a peer-to-peer network is *O(log N)*. This provides a way to estimate the time period it will take to locate a recipient's address; a distinct advantage over flooding techniques which, in some cases, would require an indeterminate number of steps or time period to locate a recipient.

[0007] Though the use of DHTs provides advantages over earlier techniques, the way they are presently used is still inadequate for use in voice applications/services. More particularly, while present techniques utilizing DHTs provide acceptable time periods to locate a recipient's address for file sharing and other data transfer applications/services, they provide unacceptable time periods when used in voice/video applications. Roughly speaking, using an existing DHT technique it takes an average of 10 steps to find a *key, value* pair in a network of 1 million nodes. However, in a worst case scenario it may take as many as 20 steps. Using 10 or 20 steps to locate the address of an intended recipient in voice or video applications is undesirable (i.e., it takes too long). Further, as a conventional P2P network gets larger, there also exists the possibility that the number of steps will increase as the

number of nodes in the network increases. Thus, while it is still possible to determine the number of steps required to establish a session this number may be too high to provide an acceptable service/application.

[0008] Accordingly, it is desirable to provide methods, devices and architectures were DHTs can be used by nodes within a P2P network to establish signaling and communication sessions within an acceptable time period in voice, video and similar applications.

SUMMARY OF THE INVENTION

[0009] The present inventors have discovered peer-to-peer architectures that utilize DHTs to establish voice and video communication sessions in carrier-based infrastructure networks much faster than previously thought possible. In accordance with one embodiment of the present invention, a provisioning server or the like designates nodes within a peer-to-peer network as *host* and *anchor* nodes. By designating certain nodes as host and/or anchor nodes, the architectures provided by the present invention require fewer hops (e.g., intermediate nodes) to establish communication sessions when compared to conventional peer-to-peer networks. The use of fewer hops in combination with novel cached-assisted look-up functions enables the architectures provided by the present invention to establish signaling paths and associated communication sessions in *O(1)* steps instead of a *O(log N)* steps. Said another way, the use of fewer hops and novel look-up functions permits signaling paths and communication sessions to be established between nodes in a peer-to-peer network faster than

previously thought possible. Further, as a P2P network grows, the novel methods provided by the present invention assure that the time it takes to establish a given session remains the same (i.e., a "constant" time period for establishing a session is maintained).

[00010] It should be noted that the techniques provided by the present invention can be used in many different types of P2P network configurations. One such P2P network is known as "Chord".

[00011] In more detail, in accordance with one embodiment of the present invention, a provisioning server assigns a *hashed key* to each host and anchor node in a peer-to-peer network.

[00012] Further, to achieve the speed needed to establish voice/video signaling paths/sessions, the provisioning server geographically associates each user with at least one host node and anchor node. Each host node is operable to store *key,value* pairs (e.g., identity and address) of one or more geographically associated users while each anchor node is assigned a *hashed key*. In accordance with the present invention, instead of requiring a signaling pathway to traverse a large number of hops/nodes, the novel architectures provided by the present invention are made up of a smaller subset of designated host and anchor nodes that form a core of a P2P network. This allows a signaling pathway to be established quickly.

[00013] Though host nodes are geographically associated with users, anchor nodes are associated in a different fashion. In accordance with the present invention, the anchor node that is associated with a given

user is the node that has a *hashed key*, the value (amount, not address) of which is equal to, or greater than, the value of a user's *key*. In addition, each anchor node is operable to store the *key,value* pair (e.g., identity and address) of the primary and secondary host nodes that have been geographically associated with its associated users. It should be further noted that the architectures provided by the present invention may utilize the anchor node-related replication and reliability mechanisms (i.e., back-up) of the underlying P2P methodology (e.g., Chord) to ensure the network operates satisfactorily when an anchor node becomes disabled, etc.. . For example, an anchor node's look-up table entries that indicate/locate a user's primary and secondary host nodes are backed-up using P2P methodology.

[00014] In an additional embodiment of the present invention one or more of the anchor nodes may also be a host node that is operable to store the *key,value* pairs (e.g., identities and addresses) of a plurality of geographically associated users.

[00015] To make the time period required to establish a session even shorter, the present invention also provides architectures which include host nodes that are operable to store, for each of its associated users, the *key,value* pairs (e.g., identities and addresses) of one or more nodes associated with one or more third party users that are frequently involved in a session with an associated user. For example, a host node may include a cache or a caching function which allows it to store the nodes (e.g., host nodes) that are associated with friends a user talks with

most often. By caching the nodes, a signaling path and associated session may be established with these friends quicker without having to use an anchor node, as will be explained in more detail below.

[00016] To make the architectures provided by the present invention even more reliable, a provisioning server may associate one or more users with two or more host nodes (instead of one). In this manner, if a user's primary host node is unable to establish/maintain a signaling path/session due to a failure or the like the user may rely on a secondary host node. As set forth above, the addresses of a user's primary and secondary host node may be stored in the user's associated anchor node look-up tables. In order to ensure that the load on a given P2P network is evenly distributed when a failure occurs, the secondary host nodes associated with a given user may vary from user to user. That is, though a given set of users may be associated with the same primary host node, they may not be associated with the same secondary host node. Thus, if and when the primary host node fails its load (i.e., users) may be distributed to many secondary host nodes instead of one.

BRIEF DESCRIPTION OF THE DRAWINGS

[00017] Fig. 1 depicts a simplified example of a peer-to-peer network in accordance with one embodiment of the present invention.

[00018] Fig. 2 depicts a simplified diagram of a peer-to-peer network in accordance with another embodiment of the present invention.

DETAILED DESCRIPTION OF THE INVENTION, WITH EXAMPLES

[00019] Referring now to Fig. 1 there is shown a peer-to-peer network 100 which includes one or more users, 200, 300, one or more host and/or anchor nodes labeled 0 through 7 and a provisioning device, such as a server, 400. Though referred to throughout this discussion as "users", it should be understood that the users 200,300 may be devices, such as telephones, personal digital assistants, computers, gaming devices, multimedia devices and the like to name some examples of devices that are associated with an actual person(s). In accordance with the present invention, the provisioning server 400 is operable to assign each user a hashed *key* in addition to being assigned an address on the peer-to-peer network 100. To distinguish those hashed keys assigned to a user from those assigned to a node, we will refer to the former as *key* and the later as *hashed key*. The server 400 also assigns a *hashed key* to each of the nodes 0 through 7. To simplify the present explanation, the *hashed keys* of nodes 0 through 7 will also be 0 through 7 (i.e., Node 0 has *hashed key* of 0. Node 4 has a *hashed key* of 4, and so on).

[00020] To establish a signaling path and communication session in accordance with the present invention, the provisioning server 400 may be further operable to associate each user with a host node that is geographically closest to the user. In Fig. 1, for user 200 this is node 7. Therefore, node 7 will be the host node that is associated with the user

200. As briefly mentioned above, to ensure that the peer-to-peer network 100 is reliable in the event that host node 7 fails, the provisioning server 400 may also be operable to associate the user 200 with two or more host nodes. For example, the server 400 may associate the user 200 with host node 0. Thus, should host node 7 fail, host node 0 will help the user 200 establish signaling paths/communication sessions. To distinguish between two host nodes associated with the same user, the first host node may be referred to as a primary host node while the backup host node may be referred to as a secondary host node. In order to ensure that the load on a given P2P network is evenly distributed when a failure occurs, the secondary host nodes associated with a given user may vary from user to user. That is, though a given set of users may be associated with the same primary host node, they may not be associated with the same secondary host node. Thus, if and when the primary host node fails its load (i.e., users) may be distributed to many secondary host nodes instead of one.

[00021] In addition to associating host nodes with users (or vice-versa) the server 400 may also separately associate one or more anchor nodes to each user. In accordance with the present invention, the association of a given anchor node with a given user depends on the technique used by the underlying P2P methodology. In the examples and discussion which follows the Chord P2P methodology is used. In accordance with one exemplary method of the present invention, the server 400 may associate an anchor node with a user as follows. The server 400 determines which node within network 100 has the smallest

hashed key, the value (amount, not an address) of which is equal to, or greater than, the value of the user's *key*. In the example shown in Fig. 1, Alice's *key* has a value of 4. The server 400 passes this information on to every node in the network 100 so that, thereafter, each node will be able to identify the node within network 100 that has a *hashed key* value of 4 as the anchor node for the user 200. In this example, that is node 4.

[00022] Similarly, though only two users 200, 300 are shown in Fig. 1 it should be understood that the provisioning server 400 may associate a plurality of users to each host node. Therefore, host node 7 may be associated with many more users than just user 200. Once associated with a user, a host node is operable to store the *key,value* pair (e.g. user ID and address) of the user.

[00023] Likewise, the anchor node 4, and in general, each anchor node within the network 100, may be operable to store the identity and address of a plurality of host nodes that have been geographically associated with one or more users. Shortly, we will explain how architectures provided by the present invention, such as the network architecture 100 illustrated in Fig. 1, may be used to quickly establish a signaling path and communication session. For now, however, it should be understood that the network architecture 100 shown in Fig. 1 permits a signaling path to be established in *O(1)* steps, amounting to a so-called "constant" time period. By "constant" time period it is meant a time period that does not change when the total number of nodes or number of users

in the network changes (e.g. increases). That is, regardless of the number of nodes and number of users in a P2P network, a signaling path/communication session from a given starting point to a given destination can be established using the same number of steps (the number of steps remains constant).

[00024] The signaling paths and sessions established by the network architecture 100 in Fig. 1 may be used to provide, VoIP services, other voice services, data services, and video services to name just a few examples.

[00025] In still further embodiments of the invention, the provisioning server 400 may be operable to detect and analyze the communication patterns and habits of one or more users. The server 400 may use this information to determine whether the host nodes it has initially geographically associated with a user should be changed. For example, if the server 400 detects that one or more users are often located in the same geographical area, the server 400 may associate such users to the same host node. This re-association may improve the quickness in which a signaling path/session is established.

[00026] We mentioned before that a user may be associated with more than one host node (e.g., in case a user's primary host node fails). The same is true for anchor nodes. In accordance with the present invention, the back-up anchor node technique used is determined by the underlying P2P methodology.

[00027] Having presented a discussion of the various elements in the P2P network architecture 100 shown in Fig. 1, we now turn our attention to how these elements operate to establish a signaling path/communication session in a constant, shorter time period than previously thought possible.

[00028] Referring now to Fig. 2, there is shown another ring-like P2P network architecture 1000 in accordance with an embodiment of the present invention. The network architecture 1000 shown in Fig. 2 includes one or more users 2000, 3000, one or more host and/or anchor nodes labeled 00 through 70 and provisioning server 4000.

[00029] In order to establish a session in a constant time period, in one embodiment of the present invention, a request from user 2000 to establish a signaling path/session (e.g., telephone call) with user 3000 is first routed to the host node 70 of user 2000, then to the anchor node 30 of user 2000, on to the host node 10 of the recipient user 3000 and then on to the recipient user 3000. By recipient it is meant the user who is the intended to receive the telephone call, etc. from the user 2000.

[00030] By routing a request to establish a signaling path/session through designated host and anchor nodes the number of nodes that are involved in establishing a signaling path/session remains constant as the network grows. This is a significant advantage over existing peer-to-peer network architectures. In essence, the present invention creates an infrastructure that contains a set of host nodes and

anchor nodes; a set of nodes within a P2P network (where each node knows all other nodes). Because the number of nodes in a P2P network provided by the present invention is, typically, far smaller and more stable than those in a traditional P2P a signaling path/communication session can be established in a far more predictable and faster manner than previously thought possible. We now present an alternative explanation of how a signaling path/session may be established in accordance with the present invention.

[00031] In the beginning, it can be assumed that the user 2000 knows that she wishes to speak to the user 3000 but she does not know where the user 3000 is located. The user 2000 (i.e., a device in use by the user 2000) may generate and send a request to establish a signaling path/session with the user 3000 to its host node 70. Within this request is a *key* (e.g., user ID) which identifies the user 3000.

[00032] Upon receiving the request from the user 2000, the primary host node 70 may be operable to forward the request to the primary anchor node 30, it being understood that the anchor node 30 is the first node from the user 2000 whose *hash key* value is equal to, or greater than the value of the user's *key*. Upon receiving the request, the anchor node 30 is operable to determine at least one, primary host node that is associated with the user 3000. That is, the anchor node 30 is the element which determines which host node is associated with the intended recipient (i.e., user 3000).

[00033] In accordance with the present invention, each anchor node comprises a storage section which stores the identities and addresses of host nodes and the users associated with these host nodes. In the example shown in Fig. 2, the anchor node 30 would search through or otherwise access its storage section to identify the identity an address of the host node which is associated with the user 3000. In this case, that host node would be node 10 shown in Fig. 2.

[00034] Upon identifying the host node 10 associated with the user/recipient 3000, the anchor node 30 is further operable to forward the request to establish a session to the host node 10. It should be understood that it is assumed that the host node 10 is available to receive such a request.

[00035] In accordance with the present invention, the anchor node 30 may have stored both a primary and secondary host node for the user 3000. This is done in order to ensure that a signaling path/session may be established if the host node 10 is otherwise unavailable. That being said, as the primary host node associated with the user 3000, node 10 will be selected first by the anchor node 30.

[00036] The anchor node 30 is operable to forward the request to form a signaling path/session to host node 10 which, thereafter, begins to initiate a signaling path/session with the user/recipient 3000. It should be understood that in the event that the request is forwarded to a secondary host node, the present invention provides for the establishment

of a signaling path/session in a constant time period as well.. That is to say, the use of a secondary host node does not significantly impact the time it takes to establish a signaling path/session.

[00037] Though the architectures shown in Figs. 1 and 2 are effective in establishing signaling paths/sessions in a significantly shorter time period than conventional techniques, the present inventors realized that the time period could be further shortened by studying the communication and location habits and patterns of a user. In still a further embodiment of the present invention, a provisioning server 4000 or the like in Fig. 2 may monitor and analyze the habits and patterns of users within the network 1000. For example, the server 4000 may identify those users that are frequently involved in a session with user 2000 most often. Thereafter, the provisioning server 4000 may identify the hosts associated with the identified users. Once the users and their associated hosts are identified, the provisioning server 4000 may communicate with the host node of the user 2000 in order to provide the host node 70 with this information. Upon receiving this information, the host node 70 may cache this information in a storage or memory section. Thereafter, when the user 2000 sends a request to establish a signaling path/session with any of the users that she communicates with most often (i.e., the identified users), the host node 70 may consult its cache, and identify the host node associated with one of these users. By identifying the host node of such a "buddy", the host node 70 may bypass the anchor node 30 and send the request directly on to the buddy's host node. By bypassing the anchor

node the time it takes to establish a signaling path/session may be further shortened.

[00038] In some instances, it may occur that the *key* value of a user is greater than all of the *hash key* values of any of the nodes in a network. If this occurs, the present invention may provide for the following exemplary method of associating an anchor node with a user; a method that is based on the Chord methodology as well.

[00039] In accordance with a Chord-related example of the present invention, the provisioning server 4000 and the like may associate a user with an anchor node that has the lowest key value when the value of the user's *key* is greater than the value of each of the *hash keys* of nodes in the network. It should be noted that, as recognized by those skilled in the art, there occurs a scenario known as a "wrap around" where this association technique may be altered somewhat. In accordance with Chord methodology, when a wrap-around scenario presents itself all of the users whose hash keys are larger than the largest key of any node in a given network are anchored at the node that has the lowest key in the network.

[00040] In the discussion above, it was assumed that the architectures 100, 1000 shown in Figs. 1 and 2 were a part of a single network operated by a single service provider. However, the present invention is not limited just to networks operated by the same service provider. In accordance with additional embodiments of the present invention the features and functions of the present invention may be

extended to architectures that include networks run by more than one service provider.

[00041] In yet a further embodiment of the present invention, one or more nodes of each of the networks operated by different service providers may act as a proxying node in order to establish signaling paths/sessions between nodes in two different networks. In this embodiment, the internal network typology of the different service providers is not disclosed. In yet another embodiment, one or more nodes of each of the networks may store or cache the host node addresses of users in the different networks. While this may result in faster establishment of a signaling path/session, it may require the disclosure of the internal typology of a given service provider's network.

[00042] As briefly mentioned above, provisioning servers or the like may be used to associate host and anchor nodes to users and vice-versa. Further, such servers may also be used to assign hash values to users and nodes. In general, a server or device that is used in the architectures provided by the present invention may include one or more computer readable mediums (e.g., memory, processors, hard drives, CD/DVD storage or some combination of these devices) for storing one or more software or firmware programs that may be executed to implement the features and functions of the present invention.

[00043] The discussion above has set forth some examples of the present invention. Some of the features provided by the present

invention are: (a) cached-based P2P networks with reduced look-up/session establishment time periods; (b) separate host and anchor nodes for reducing establishment time periods and enhancing the geographic association of users to nodes; and (c) the even distribution of load upon node failure.

[00044] The true scope of the present invention, however, is set forth in the claims that follow in which the term "session" will be used to denote a signaling path, communication session or both.

WE CLAIM:

1. An architecture that enables a session to be established in a distributed hash table (DHT)-based, peer-to peer network in a constant time period comprising:

a provisioning server operable to,

geographically associate one or more users with at least one primary host node, and

further associate each user to at least one anchor node,

wherein the architecture enables a session to be established in a constant time period.

2. The architecture as in claim 1 further comprising one or more primary host nodes, each host node operable to:

store the identity and address of one or more associated users; and

forward a request to establish a session from an associated user to an anchor node or a host node of a recipient user.

3. The architecture as in claim 1 further comprising one or more anchor nodes, each anchor node operable to:

store the identity and address of at least one primary host node that has been geographically associated with one or more users; and

forward a request to establish a session with an associated user to a host node of an intended recipient of the request using a stored identity and address of the recipient's host node.

4. The architecture as in claim 1 wherein the provisioning server is further operable to associate users associated with the same primary host node to different secondary host nodes to evenly distribute the primary host node's load throughout the network when the primary host node is unavailable.

5. The architecture as in claim 4 wherein each anchor node is further operable to store the identity and address of at least one secondary host node associated with each user,

wherein each secondary host node is a back-up host node for its respective user when the primary host node is unavailable.

6. The architecture as in claim 1 wherein the session comprises a voice-over-IP (VoIP) session.

7. The architecture as in claim 2 wherein each host node is operable to store, for each of its associated users, the identity and address of one more host nodes associated with one or more third party users that are frequently involved in a session with an associated user.

8. A method for establishing a session in a distributed hash table (DHT)-based, peer-to-peer network in a constant time period comprising:

geographically associating one or more users with at least one primary host node;

further associating each user to at least one anchor node;

storing the identity and address of one or more associated users in one or more primary host nodes; and

forwarding a request from a primary host node to an anchor node or a host node of an intended recipient of the request to establish a session between the recipient and a user associated with the forwarding host node.

9. The method as in claim 8 further comprising:

storing the identity and address of at least one primary host node that has been geographically associated with one or more users in one or more anchor nodes; and

forwarding a request from an anchor node to a host node of an intended recipient of the request using a stored identity and address of the recipient's host node to establish a session between the recipient and a user associated with the forwarding anchor node.

10. The method as in claim 8 further comprising:

associating users associated with the same primary host node to different secondary host nodes to evenly distribute the primary host node's load throughout the network when the primary host node is unavailable; and

storing the identity and address of at least one secondary host node associated with each user,

wherein each secondary host node is a back-up host node for its respective user when the primary host node is unavailable.

FIG. 1

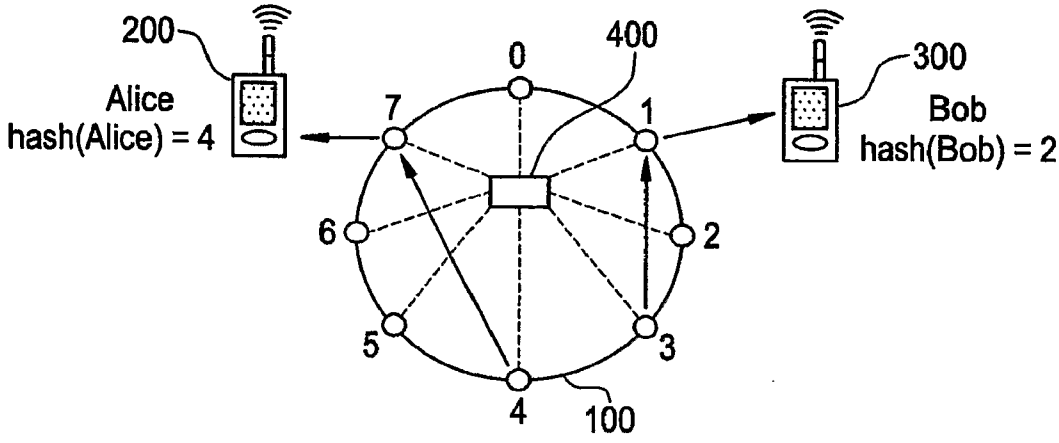
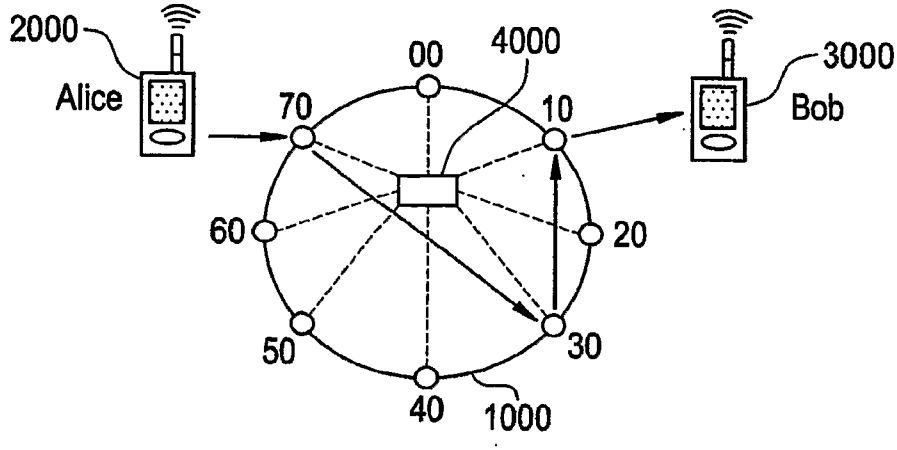

200
Alice
hash(Alice) = 4
0
400
7
1
300
Bob
hash(Bob) = 2
6
2
5
3
4
100

FIG. 2


2000
Alice
00
4000
70
10
3000
Bob
60
20
50
30
40
1000