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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
6 May 2016 (06.05.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/069915 A1

(51) International Patent Classification:
G06F 21/53 (2013.01) *G06F 21/64* (2013.01)

(21) International Application Number:
PCT/US2015/058102

(22) International Filing Date:
29 October 2015 (29.10.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/073,008 30 October 2014 (30.10.2014) US

(71) Applicant: **SYNC-N-SCALE, LLC** [US/US]; 357 Beloit Street, Burlington, Wisconsin 53105 (US).

(72) Inventor: **VOBA, Son**; 24429 NE 26th Street, Sammamish, Washington 98074 (US).

(74) Agents: **KEES, Nicholas, A** et al.; Godfrey & Kahn, S.C., 780 North Water Street, Milwaukee, Wisconsin 53202 (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: METHOD AND SYSTEM FOR GEOLOCATION VERIFICATION OF RESOURCES

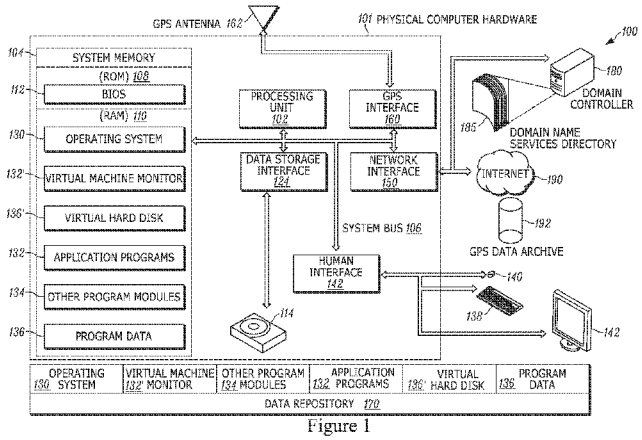


Figure 1

(57) Abstract: A method of geolocation verification, including obtaining the geolocation of an operating system, generating a unique system ID for an installed operating system, and transmitting the geolocation of the operating system and a system ID to a data repository. The method further includes receiving a request to either initiate deployment of, or grant access to, a computer object associated with the operating system, identifying if the computer object requires geolocation verification, then identifying an object ID associated with the computer object and communicating each of the object ID, the geolocation of the operating system, and the system ID, to a domain controller for assessment. The method also includes searching the data repository to identify one or more geolocation object resource claims associated with the object ID, and comparing the geolocation resource claims with the communicated geolocation of the operating system.

WO 2016/069915 A1

METHOD AND SYSTEM FOR GEOLOCATION VERIFICATION OF RESOURCES

CROSS REFERENCE TO RELATED APPLICATIONS

[0001] The present application claims priority to U.S. Provisional Patent Application No. 62/073,008 filed on October 30, 2014, which is herein incorporated by reference in its entirety.

FIELD

[0002] The method and system relate in general to the field of geolocation verification for various types of resources.

BACKGROUND

[0003] A datacenter is where application software and customer data running on the software are located. Vendors of cloud-based IT services must maintain transparency of where they replicate customer data at any given time for protection against failure or local disaster. If a datacenter ceases functioning for any reason, customer data will not be lost if the application software and customer data running on that application software are also available from a second or possibly third datacenter. And assuming it works smoothly enough, customers might not even be notified when such a failover occurs. Depending on the particular service, failover may not result in any service interruption at all.

[0004] Global enterprises have leveraged the Internet and cloud-based computing services, along with datacenters, to establish private communication networks and capture efficiencies from global technology. As a result of such globalization, in recent years, many nations have issued geographic location rules restricting how corporations can handle and transmit their customers' data across borders, including through these private networks. Certain entities require that specific types of data, for example government data, employee data, or

telecommunications traffic data be stored within a limited geographical border, and in some cases, such data may even not be accessed from outside of a geographical border.

[0005] A geographic location is specified by a set of coordinates representing the latitude, longitude and elevation which are the principal elements of a geographic coordinate system. The latitude (φ, or phi) of a point on the Earth's surface is the angle between the equatorial plane and the straight line that passes through that point and through (or close to) the center of the Earth. The equator divides the globe into Northern and Southern Hemispheres. The longitude (λ, or lambda) of a point on the Earth's surface is the angle east or west from a reference meridian to another meridian that passes through that point. The internationally recognized reference Prime Meridian passes through a point in Greenwich, England, and determines the proper Eastern and Western Hemispheres. The elevation of a point on the Earth's surface is typically its height relative to sea level; while altitude is used for points above sea level, such as an aircraft in flight or a spacecraft in orbit, depth is used for points below sea level.

[0006] Geographic location of a position on Earth can be obtained from beacons like Wi-Fi access points and cell towers, from the IP address of a device, or it may come from other sources such as a Global Navigation Satellite System (GNSS) or Global Positioning System (GPS) device. The accuracy of geographic location information depends on the source, and may vary from the actual position of a device, computer or resource within the following exemplary ranges:

- GPS: within approximately 10 meters
- Wi-Fi: between approximately 30 meters and 500 meters
- Cell towers: between approximately 300 meters and 3,000 meters
- IP address: between approximately 1,000 meters and 5,000 meters

[0007] When dealing with information technology operations, it is common to utilize the attribution of one or more types of claims. A claim is a unique piece of information about a user, device, computer, or resource. These are very often attributes that can be found as properties of a computer object in a domain name services directory – things like a user's functional title, organizational department or office location, are claims that can be defined. So is the business impact classification of a data file, or the health status of a computer. A computer object or entity can involve more than one claim, and any combination of claims can be used to authorize access to resources. The following exemplary types of claims are typically available in a commercially available domain name services directory:

- User claims: attributes that are associated with a specific user.
- Device claims: attributes that are associated with a specific computer object.
- Resource attributes: global resource properties that are marked for use in authorization decisions.

[0008] Claims are generally protected in a domain name services directory which is operated upon and published by a domain controller and its surrogates. This is to prevent them from being tampered with by unauthorized personnel, and accessible only to properly authenticated users, devices and computers. Claims make it possible for administrators to make precise organization- or enterprise-wide statements about users, devices, computers, and resources that can be incorporated in expressions, rules, and policies for IT operations.

BRIEF SUMMARY

[0009] In at least some embodiments, the method and system for geolocation verification of resources is directed to the use of geographic location data (e.g., GPS signals) to determine whether a computer object (such as a virtual hard disk), can be deployed and its virtual machine

can be operated by a physical server located in a specific geographical location. Further, in at least some embodiments, the domain controller arbitrates by resolving whether the current geographic location of the operating physical server is inside of the geographic region where a virtual machine can be operated.

[0010] Exemplary embodiments include methods and systems for determining the current geographic location of a physical server, including a general-purpose computer, a GPS system interface, a connecting cable, and a GPS antenna. To obtain authorization for deployment of a virtual hard disk based on the current geographic location of a physical server, the physical server acquires its current geographic location and the identification of the virtual hard disk to be deployed, and communicates the information to a domain controller. The domain controller performs the verification process and communicates a positive or negative assessment result to the physical server to indicate if deployment of the virtual machine contained inside the virtual hard disk can proceed. In at least some embodiments, the operating system (for both physical server and virtual machine) is capable of self-validation using the same method and system. The operating system forwards its current geographic location and its own identification to the domain controller. Upon receiving a satisfactory result to proceed, it may continue operating beyond the initial boot phases.

[0011] In at least some embodiments, the method and system for geolocation verification of resources relates to a method that includes providing an operating system in communication with at least one processor and at least one memory communicatively coupled to the at least one processor, the memory having stored therein computer-executable instructions; creating, via execution of the computer-executable instructions by the computing device, a computer object, the computer object including an authentication data portion and a program data portion, the

authentication data portion being accessible by an application independently from accessing the program data portion of the computer object; generating a unique object ID for the computer object and writing the unique object ID into the authentication data portion and the program data portion; obtaining one or more geographic object resource claims for the computer object, wherein the geographic object resource claims include one or more geographic locations where the computer object is authorized to be operated; and communicating the unique object ID and the one or more geographic object resource claims to an authorizing entity.

[0012] In at least some additional embodiments, the method and system for geolocation verification of resources relates to a method that includes obtaining the geolocation of an operating system installed on a computing device that includes a processor and a memory having stored therein computer-executable instructions; generating a unique system ID for the installed operating system; transmitting the geolocation of the operating system and the system ID to a data repository for a domain controller in communication with the operating system; receiving at the computing device, a request to at least one of initiate deployment of, or grant access to a computer object associated with the operating system; identifying if the computer object requires geolocation verification, and if verification is required, then identifying an object ID associated with the computer object and communicating each of the object ID, the geolocation of the operating system, and the system ID, to the domain controller for assessment; performing a geolocation verification analysis that includes searching the data repository using the object ID to identify one or more geolocation object resource claims associated with the object ID, wherein the geographic object resource claims include one or more geographic locations where the computer object is authorized to be operated; and comparing the geolocation resource claims

with the communicated geolocation of the operating system to provide confirmation or denial of geolocation verification for the computer object.

[0013] In at least some further embodiments, the method and system for geolocation verification of resources relates to a system including an operating system installed on a computing device. The computing device includes a processor and a memory having stored therein computer-executable instructions. At least one of, the operating system includes a unique system ID, and the computing device includes a unique device ID. A Global Positioning System interface is in communication with at least one of the computing device and the operating system, and is capable of providing a geolocation for at least one of the computing device and the operating system. A network interface transmits the geolocation of at least one of the computing device and the operating system, and at least one of the system ID and the device ID, to a domain controller in communication with a data repository. A computer object, associated with the operating system, includes a unique object ID and one or more geographic object resource claims. The geographic object resource claims include one or more geolocation boundaries where the computer object is authorized to be operated.

[0014] Other embodiments, aspects, features, objectives and advantages of the method and system will be understood and appreciated upon a full reading of the detailed description and the claims that follow.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] Embodiments of the method and system are disclosed with reference to the accompanying drawings and are for illustrative purposes only. The method and system is not limited in its application to the details of construction or the arrangement of the components

illustrated in the drawings. The method and system are capable of other embodiments or of being practiced or carried out in other various ways. In the drawings,

[0016] FIG. 1 illustrates a block diagram an exemplary computing environment and system;

[0017] FIG. 2 illustrates a block diagram of an exemplary architecture of software and hardware;

[0018] FIG. 3A illustrates a circular defined boundary of a geographical region of a site within a building;

[0019] FIG. 3B illustrates a rectangular defined boundary of a geographical region of a site within a building;

[0020] FIG. 3C illustrates another rectangular defined boundary of a geographical region of a site within a building;

[0021] FIG. 3D illustrates a polygonal defined boundary of a geographical region of a site encompassing one or more buildings;

[0022] FIG. 3E illustrates a boundary of a geographical region of a jurisdiction, more particularly, the State of Wyoming (USA);

[0023] FIG. 4 illustrates a flow diagram of an exemplary method for marking a virtual hard disk to be operated in one or more specific geographic regions;

[0024] FIG. 5 illustrates a flow chart of an exemplary method for controlled deployment of a virtual machine that has been marked to operate in one or more specific geographic regions;

[0025] FIG. 6 illustrates a flow chart of an exemplary method for controlled deployment of an application container that has been marked to operate in one or more specific geographic regions;

[0026] FIG. 7 illustrates a flow chart that represents an exemplary method for controlled access to a data repository that has been marked to be accessible from within one or more specific geographic regions.

DETAILED DESCRIPTION

[0027] The following detailed description refers to the accompanying drawings that illustrate exemplary embodiments of the present invention. However, the scope of the present invention is not limited to these embodiments. Thus, embodiments beyond those shown in the accompanying drawings, such as modified versions of the illustrated embodiments, may nevertheless be encompassed by the method and system of geographical verification.

[0028] References in the specification to "one embodiment," "an embodiment," "an example embodiment," or the like, indicate that the embodiment described may include a particular feature, structure, or characteristic, but every embodiment may not necessarily include the particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same embodiment. Furthermore, when a particular feature, structure, or characteristic is described in connection with an embodiment, it is submitted that it is within the knowledge of one skilled in the relevant art(s) to implement such feature, structure, or characteristic in connection with other embodiments whether or not explicitly described.

[0029] FIG. 1 illustrates a block diagram of an exemplary computing environment and system 100, and FIG. 2 illustrates a block diagram of an exemplary architecture of software and

hardware. In at least some embodiments, a system 100 includes a computer 101. Computer 101 may represent, for example, an enterprise or corporate server operating in its respective datacenter, or a server operated by a commercial hosting datacenter. However, this example is not intended to be limiting and computer 101 may operate in other environments as well, and persons skilled in the relevant art(s) will appreciate that the method and system described herein may be performed by a wide variety of computers other than computer 101. Computer 101 can include a device ID that uniquely identifies the computer 101, where the device ID can include numerous types of identifiers, such as serial numbers, etc. In addition, an operating system 130 can be installed on computer 101, which can include a system ID that is created when the operating system is installed on the computer 101. In each instance when an operating system is installed, a new unique system ID can be created for it.

[0030] Referring to FIG. 1, in at least some embodiments, computer 101 comprises a plurality of interconnected hardware components including, but not limited to, a processing unit 102, system memory 104, a data storage interface 124, a network interface 150, a human interface 142, and a GPS interface 160. System memory 104 includes computer storage media in the form of volatile and/or non-volatile memory such as ROM 108 and RAM 110. A basic input/output system 112 (BIOS), containing the basic routines that help to transfer information between elements within computer 101, such as during start-up, is typically stored in ROM 108. RAM 110 typically contains data and/or program modules that are immediately accessible to and/or presently being operated on by processing unit 102. By way of example, and not limitation, FIG. 1 illustrates an operating system 130, application programs 132, other program modules 134, program data 136, a virtual machine monitor 132', and a virtual hard disk 136'. Application programs 132 are sometimes referred to herein as executable binary files.

[0031] The computer 101 may also include other removable/non-removable, volatile/non-volatile computer storage media. By way of example only, FIG. 1 illustrates a hard disk drive 114 that reads from or writes to non-removable, non-volatile magnetic media. Other removable/non-removable, volatile/non-volatile computer storage media that can be used in the exemplary operating environment include, but are not limited to an optical disk, such as a CDROM or digital versatile disks, magnetic tape cassettes, flash memory cards, digital video tape, solid state RAM, solid state ROM, and the like. Hard disk drive 114 is typically connected to a system bus 106 through data storage interface 124. The computer 101 is configured to read from and write to a data repository that is reachable over at least one of an Intranet and Internet through its network interface 150 which is typically connected to system bus 106. In at least one embodiment, a domain name services directory 185 serves as an Intranet data repository, and a GPS data archive 192 serves as an Internet data repository. References herein to the use of domain name services directory 185 should be understood to include other data repositories that are provided in addition to, or in place of, domain name services directory 185, including but not limited to GPS data archive 192. Additional elements of computer 101 will be described in more detail below.

[0032] As discussed above, the method and system for geolocation verification of resources involves ascertaining the current geographic location of computer 101. More particularly, in at least some embodiments, the method and system for geolocation verification of resources includes ascertaining and attesting that a computer or its operating system is operating within an authorized geographic location. Specific geographical location determination can be performed in several different manners, although in at least one embodiment, the Global Positioning System (GPS) is utilized as a principal source to provide the desired geographical

location data, while other embodiments can utilize other sources for obtaining geographical location data.

[0033] As shown in FIG. 1, processing unit 102, which comprises one or more microprocessors or microprocessor cores in computer 101, is designed to execute program instructions stored in system memory 104 to cause GPS interface 160 to communicatively connect to a GPS antenna 162, and to perform its designated functions. This interface facilitates the reception of GPS signals and the delivery of current time of day, latitude, longitude and transmitting satellite PRNs (Pseudo Random Noise) to operating system 130 of computer 101. In at least one embodiment, GPS interface 160 comprises a PCIe add-on card designed and manufactured by Sync-n-Scale, LLC of Burlington, Wisconsin. However, this example is not intended to be limiting, and GPS interface 160 may comprise any conventional or subsequently-developed hardware, or a combination of hardware and software that is designed to rely on comparable navigation aiding systems and to perform the aforementioned functions.

[0034] In addition to obtaining the geographical location of a particular computer, the method and system for geolocation verification of resources can further include using geolocation information as a prerequisite indicator for starting and operating a computer object, such as a virtual machine, a self-contained application program, or a container, as discussed in detail below. Referring now to FIG. 2, virtual machine monitor 132', also called a hypervisor, is shown. A hypervisor is a special type of application program 132 that emulates another computer capable of running an operating system. The emulated computer is called a virtual machine, for example virtual machine 101'. Virtual machine 101'is capable of running operating system 130', which is the same operating system as computer operating system 130. Although not discussed in detail, more than one virtual machine 101' can be emulated via virtual machine monitor 132'.

[0035] Virtual machine monitor 132' can create and start virtual machine 101' using stored instructions in virtual hard disk 136'. Computer 101, or another similarly capable device, may be used to create the virtual hard disk 136' image, and to store it in a separate Intranet or Internet data repository, as discussed in greater detail below. A virtual hard disk 136' is, in at least some embodiments, a special type of program data 136 constructed in the form of a single computer file following a specific format. This file encapsulates an emulated storage device capable of carrying one or more file systems and supporting standard disk and file read and write operations. As a standard computer file, the virtual hard disk 136' is subject to storage, operating and access policies that other computer files would adhere to on the same computer. These policies are expressed as device claims and resource attributes, and are maintained in a domain name services directory, such as domain name services directory 185, which can be an organization- or enterprise-wide domain name services directory, or other comparable data repository. At creation, a virtual hard disk 136' would be cryptographically protected, or otherwise protected for security purposes, using classical data protection methods available to the hypervisor 132. This protection is performed to prevent tampering of or unauthorized access to the contents of a virtual hard disk 136' that forms a computer object.

[0036] Referring again to FIG. 1, in at least some embodiments, application program 132 can be self-sufficient and take the form of a single executable binary file or a special type of program module 134, also called a container. A container, can include everything an application program needs to run including: executable instructions, application run-time libraries, system tools and libraries. Similar to other objects, it can be cryptographically protected, or otherwise protected for security purposes at creation using classical data protection methods available to the operating system to prevent tampering.

[0037] As discussed above, in at least some embodiments, the method and system for geographic verification includes associating a geographic region with various computer object types. In at least some embodiments, the computer object can be comprised of a computer file, a storage device that contains a computer file, an interface that receives and emits a computer file from a storage device, a database that embodies a computer file, a storage device and an interface, and various other combinations.

[0038] The geographic region providing the permissible location(s) where the computer object is permissibly operated may be identified by a collection of geolocation data points, and optionally by additional geometric dimensions. Referring to FIGS. 3A, 3B, 3C, 3D, and 3E, a grouping of these data points may contain any one of: a) a pair of longitude and latitude coordinates 322 plus a radius 323 (FIG. 3A), b) a pair of longitude and latitude coordinates 322 plus radiating length 324 and radiating width 325 (FIG. 3B), or c) a collection of two or more pairs of longitude and latitude coordinates (FIGS. 3C, 3D, 3E). Each grouping of these geolocation data points frame a perimeter, such as imaginary circle 330, rectangles 332, 334, or polygon 336. Each perimeter representing an actual contiguous and bounded geographic region on the Earth's surface where the computer object can be deployed, started and operated.

[0039] A geographic region can be as small as the immediate floor space surrounding a server rack footprint inside a datacenter building 320 (FIG. 3A), or as large as a jurisdiction 340 whose boundary is expressed as a series of geographic coordinate pairs 342, 344, 346, and 348, as shown in FIG. 3E. Further, a rectangular geographic region can be calculated or derived from geographic coordinates. The four corners of a rectangular geographic region perimeter can be calculated from the given geographic coordinate 322 and the radiating length 324 and width 325

as shown in FIG. 3B, or derived from the two given geographic coordinates 322' and 322" to obtain the other two 326 and 327, as shown in FIG. 3C.

[0040] Each grouping of these geolocation data points can be identified as a geolocation resource claim (i.e. resource attribute) of a computer device (e.g., server, etc.), an operating system associated with a computer device, or a computer object. Each computer object may be assigned multiple geolocation claims. Geolocation claims for a computer object may be referred to herein as geographic object resource claims. Geographic object resource claims can be recorded and associated with a computer object's unique identifier, referred to herein as an object ID (discussed in greater detail below). This information can be maintained in a domain name services directory 185 or similar type data repository. In at least some embodiments, access to domain name services directory 185 is limited to one or more levels of authorized access. Similar to assigning a computer object one or more geolocation claims, computer operating system 130 or hypervisor 132' can also be assigned at least one authorized operating geographic region. Each geographic region establishes an area within which operating system 130 is allowed to access similarly confined resources. This geographical extent includes one or more geolocation claims of the associated operating system or hypervisor instance (not the computer itself), and can be referred to herein as geographic resource device claims. Multiple geographic resource device claims can be assigned for this purpose. As with the geographic resource object claim, the geographic resource device claim and associated details are protected inside a different, or the same, domain name services directory 185 and would not be accessible by the administrator or operator of the computer without the required authorization.

[0041] Subsequent to the creation of one or more geolocation claims, an authorized user may add, subtract or replace the geolocation coordinates in the geolocation claim and extract

them for information technology (IT) administrative purposes. Such changes can be accomplished using human interface 142, which will normally include devices such as a monitor 138, a keyboard 139, and a mouse 140. Separate from the geolocation claim, an assigned geolocation indicator (e.g., file attribute), which can inform a user of an intended permissible geolocation and identify if the computer object is tagged for verification (geo-tagged), can be associated with the computer object and therefore visible upon casual inspection by all users. The addition of an assigned geolocation indicator does not affect geolocation claims, as they remain protected and are therefore inaccessible under casual inspection. This separation of access authorization and control can be implemented for example, by domain controller 180 in combination with domain name services directory 185 or another data repository.

[0042] Marking a computer object with one or more geographic object resource claims (geo-tagging), can be performed with many types of computer-based operating systems and can take place in various locations, such as at a data center, or at a deployment center. An exemplary marking process is described with reference to flowchart 400 in FIG. 4. To begin the process, at step 402, the geographic resource object claim(s) is formed. More particularly, the geographical coordinates for one or more locations, such as that of a datacenter to be authorized to operate the computer object, are obtained and stored. In at least some embodiments, the geographical coordinates are stored as a resource attribute in domain name services directory 185 associated with a domain controller 180. At step 404, a unique object ID is generated for the computer object which will be used in the verification process. In at least some embodiments, the unique object ID is a Globally Unique ID (GUID), although other types of identifiers can be used.

[0043] At step 406, the computer object to require geolocation-based authorization for access or operation is created by the operating system, and the unique object ID is embedded or

otherwise associated with the object. This task can be performed in many different ways. For example, when the computer object is file based, such as a virtual hard disk (e.g., virtual hard disk 136') for operating a virtual machine (e.g., virtual machine 101'), the virtual hard disk image file may be created with more than one portion (e.g. partition, section, etc.). In at least some embodiments, the sections can include an authentication data portion and a program data portion. The program data portion can contain, for example, the instructions for operating virtual machine 101' and the unique object ID. The authentication data portion can contain, for example, the unique object ID and geolocation verification requirement information about virtual machine 101', such as the geographic resource object claims. It is intended that the authentication data section would be accessible independently from accessing the program data portion, and only by an authorized entity. This would allow for authentication without access to the object's program content. In addition, the aforementioned process of providing a first secured portion that is separately viewable from a second secured portion can be used for various types of computer objects, as noted above.

[0044] At step 408, the operating system stores the unique object ID for the computer object in domain name services directory 185. At step 410, domain controller 180 binds the geographic resource object claim to the unique object ID in the domain name services directory 185 as a resource attribute. At step 412, the operating system applies additional resource attributes and data protection methods to the computer object for authorized access purposes and saves the computer object for later deployment. As noted above, one added attribute can be an assigned geolocation indicator. Once geo-tagged for geolocation, computer objects can be distributed to data centers or other sites for later access and can be recognizable as geo-tagged to an operating system trying to access the geo-tagged object.

[0045] Referring now to FIGS. 5, 6, and 7, to put a geo-tagged computer object into service at a datacenter or other service point, and in some cases, continue the use of a geo-tagged object, an access request is made and verification is performed. This process is discussed in detail below with reference to flow charts 500, 600, and 700. More particularly, the deployment/accessing of various types of objects, including an exemplary virtual machine (FIG. 5), an exemplary container (FIG. 6), and an exemplary data repository (FIG. 7) is discussed. As the procedural steps are, in at least some embodiments, similar for many types of computer objects, including these three types of computer objects, the steps will be addressed simultaneously for the three flow charts.

[0046] Beginning with steps 502, 602, 702, when operating system 130 begins a startup or resume event, and at times during its continued operation, operating system 130 obtains from its active GPS interface 160 (having a connected GPS antenna 162), the actual and attestable geolocation where it is currently being operated, also referred to herein as a device geolocation claim. If the computer being activated is a virtual machine (e.g., 101'), its operating system 130' obtains the geolocation information from its hypervisor 132'. At steps 504, 604, 704, the actual geolocation where the operating system 130 is currently operating is made available to system administrators and operators, and forwarded to the appropriate domain controller 180 to update domain name services directory 185. This process can be continuous, occurring before, during, and after a computer object is deployed or otherwise verified for operation.

[0047] At steps 506, 606, 706 the operating system 130 receives a request to initiate deployment of, or otherwise grant access to the computer object (virtual hard disk 136', container 134, or executable binary file 132). If the computer object is identified as geo-tagged, such as with a geolocation indicator, then the process advances to step 508, 608, 708, wherein

operating system 130 communicates the unique object ID of the computer object to domain controller 180 for assessment. In at least some embodiments, the operating system 130 also communicates the system ID for the operating system 130. If the computer object is not identified by the operating system as geo-tagged, then the process advances to steps 514, 614, 714, wherein deployment and/or access to the computer object is granted. At steps 510, 610, 710, using the communicated object ID, the domain controller 180 searches the domain name services directory 185 to identify the geographic object resource claims for the computer object. Once identified, domain controller 180 initiates a comparison between the geographic object resource claims bound to the object ID as recorded in domain name services directory 185, with the reported device geolocation claim (geolocation coordinates) of operating system 130 that are also being stored in the domain name services directory 185. In at least some embodiments, the comparison includes domain controller 180 solving a point-in-polygon (PIP) problem using the communicated device geolocation claim and the geographic object resource claims. The PIP geometric calculation solves the question of whether a given point in a plane lies inside, outside, or on the boundary of a polygon. Classical geometric calculation and methods can also be used to determine whether a given geolocation associated with the computer object falls in or outside of an allowable defined contiguous geographic region on the Earth's surface. Such calculations are performed by the domain controller 180 in response to the access authorization request. More particularly, if at steps 512, 612, 712, the device geolocation claim is declared to fall within the boundary (i.e. region) representing the protected geographic object resource claim(s) assigned to the computer object, then the domain controller will declare the assessment to be a success and return a pass result confirming geolocation verification. If the device geolocation claim is declared to fall outside of the boundary (i.e. region) representing the protected geographic object

resource claim(s) assigned to the computer object, then the process moves to steps 516, 616, 716 and the domain controller will declare the assessment to be a failure. In response, operating system 130 acts accordingly. That is, the process moves to steps 514, 614, 714, where operating system 130 and its hypervisor 132' are authorized to proceed with the deployment of virtual machine 101', running a guest operating system 130', executable binary file 132, or operating container 134.

[0048] Advantageously, this method can also provide an additional layer of protection of virtual hard disk 136' by preventing it from being copied within an organization and operated outside a specific physical datacenter site and in an unauthorized manner. This method also allows for discovery and mitigation when an attempt to deploy and start a virtual machine by a rogue employee who utilizes legitimate authorized access to the storage of a virtual hard disk for back-up and other data management operations, but not authorized to deploy and start a virtual machine using its contents. In addition to being able to attest to the location and jurisdiction contractual obligation of a system, this method in accordance with the principles of this invention also allows the vendors to determine best efficient datacenter sites to meet their customer demands while operating their workloads within environmental and infrastructural constraints such as electricity and communications. Furthermore, in at least some embodiments, this method can be used to ascertain if a user is operating a device or computer within a geographic region and therefore permitted by pre-determined policies to create a data repository 170 and attach it to an operating system, or to access an existing data repository being curated in the same or another geographic region.

[0049] As mentioned above, while exemplary embodiments of the present invention have been described in connection with various computing devices and system architectures, the

underlying concepts may be applied to any computing device or system in which it is desirable to implement controlled deployment and operation of a virtual machine in a geographically specific datacenter.

[0050] Thus, the methods and systems of the present invention may be applied to a variety of applications and systems. While exemplary hardware interfaces, names and examples are chosen herein as representative of various choices, these hardware interfaces, names and examples are not intended to be limiting. One of ordinary skill in the art will appreciate that there are numerous ways of obtaining geolocation data that achieves the same, similar or equivalent systems and methods achieved by the invention.

[0051] The various techniques described herein may be implemented in connection with hardware or software or, where appropriate, with a combination of both. Thus, the methods and system of the present invention, or certain aspects or portions thereof, may take the form of program code (i.e., instructions) embodied in tangible media, such as CD-ROMs, flash drives, hard drives, or any other machine readable storage medium, wherein, when the program code is loaded into and executed by a machine, such as a computer, the machine becomes a system for practicing the invention. In the case of program code execution on programmable computers, the computing device will generally include a processor, a storage medium readable by the processor (including volatile and non-volatile memory and/or storage elements), at least one input device, and at least one output device. One or more programs that may utilize the signal processing services of the present invention, e.g., through the use of a data processing API or the like, are preferably implemented in a high level procedural or object oriented programming language to communicate with a computer. However, the program(s) can be implemented in assembly or

machine language, if desired. In any case, the language may be a compiled or interpreted language, and combined with hardware implementations.

[0052] The methods and system of the present invention may also be practiced via communications embodied in the form of program code that is transmitted over some transmission medium, such as over electrical wiring or cabling, through fiber optics, or via any other form of transmission, wherein, when the program code is received and loaded into and executed by a machine, such as an EPROM, a gate array, a programmable logic device (PLD), a client computer, a video recorder or the like, or a receiving machine having the signal processing capabilities as described in exemplary embodiments above becomes a system for practicing the invention. When implemented on a general-purpose processor, the program code combines with the processor to provide a unique system that operates to invoke the functionality of the present invention. Additionally, any storage techniques used in connection with the present invention may invariably be a combination of hardware and software.

[0053] While the present invention has been described in connection with the preferred embodiments of the various figures, it is to be understood that other similar embodiments may be used or modifications and additions may be made to the described embodiment for performing the same function of the present invention without deviating therefrom. Furthermore, it should be emphasized that a variety of computer platforms, including handheld device operating systems and other application specific operating systems are contemplated, especially as the number of wireless networked devices continues to proliferate. Therefore, the present invention should not be limited to any single embodiment, but rather should be construed in breadth and scope in accordance with the appended claims.

[0054] While the invention has been described with reference to preferred embodiments, it is to be understood that the invention is not intended to be limited to the specific embodiments set forth above. Thus, it is recognized that those skilled in the art will appreciate that certain substitutions, alterations, modifications, and omissions may be made without departing from the spirit or intent of the invention. Accordingly, the foregoing description is meant to be exemplary only, the invention is to be taken as including all reasonable equivalents to the subject matter of the invention, and should not limit the scope of the invention set forth in the following claims. Further, the steps described herein with reference to the method of operation are not to be considered limiting and can include variations, such as additional steps, removed steps, and re-ordered steps.

CLAIMS

I CLAIM:

1. A method comprising:

providing an operating system in communication with at least one processor and at least one memory communicatively coupled to the at least one processor, the memory having stored therein computer-executable instructions;

creating, via execution of the computer-executable instructions by the computing device, a computer object, the computer object including an authentication data portion and a program data portion, the authentication data portion being accessible by an application independently from accessing the program data portion of the computer object;

generating a unique object ID for the computer object and writing the unique object ID into the authentication data portion and the program data portion;

obtaining one or more geographic object resource claims for the computer object, wherein the geographic object resource claims include one or more geographic locations where the computer object is authorized to be operated; and

communicating the unique object ID and the one or more geographic object resource claims to an authorizing entity.

2. The method of claim 1 wherein the computer-executable instructions are further executed by the operating system to include:

obtaining the computer object; and

reading only the authentication data portion without accessing the program data portion.

3. The method of claim 2, wherein the computer-executable instructions are further executed by the operating system to include communicating with a Global Positioning Satellite interface to identify the geographic location where the operating system is currently operating.

4. The method of claim 3, wherein the computer-executable instructions are further executed by the operating system to include:

communicating the following: a) object ID of the computer object, b) at least one of a device ID for a computer running the operating system and a system ID associated with the operating system, and c) the geographic location identified by the GPS to the authorizing entity; and

receiving in response to the communication, a positive geolocation verification from the authorizing entity or a negative geolocation verification, wherein upon receipt of a positive geolocation verification, the operating system is authorized to access the program data portion in the computer object, and wherein upon receipt of a negative geolocation verification, the operating system is not authorized to access the program data portion.

5. The method of claim 1, wherein the computer object is at least one of a computer file, a storage device containing a computer file, an interface that receives and emits a computer file from a storage device, and a database.

6. The method of claim 1, wherein the computer object is a virtual hard disk image of a virtual machine installed on a physical server.

7. The method of claim 1, wherein the operating system is installed and operated on a physical server.

8. The method of claim 1, wherein the operating system is installed and operating on a virtual machine in communication with a physical server.

9. A method of geolocation verification comprising:

obtaining the geolocation of an operating system installed on a computing device that includes a processor and a memory having stored therein computer-executable instructions;

generating a unique system ID for the installed operating system;

transmitting the geolocation of the operating system and the system ID to a data repository for a domain controller in communication with the operating system;

receiving at the computing device, a request to at least one of initiate deployment of, or grant access to a computer object associated with the operating system;

identifying if the computer object requires geolocation verification, and if verification is required, then identifying an object ID associated with the computer object and communicating each of the object ID, the geolocation of the operating system, and the system ID, to the domain controller for assessment;

performing a geolocation verification analysis that includes searching the data repository using the object ID to identify one or more geolocation object resource claims associated with the object ID, wherein the geographic object resource claims include one or more geographic locations where the computer object is authorized to be operated; and

comparing the geolocation resource claims with the communicated geolocation of the operating system to provide confirmation or denial of geolocation verification for the computer object.

10. The method of claim 9, wherein the computer object includes an authentication data portion and a program data portion, the authentication data portion being accessible by an application independently from accessing the program data portion of the computer object.

11. The method of claim 10, wherein identifying if the computer object requires geolocation verification does not include accessing the program data portion, and wherein only a confirmation of geolocation verification authorizes the operating system to access the program data portion.

12. The method of claim 10, wherein identifying the computer object ID does not include accessing the program data portion, and wherein confirmation of geolocation verification authorizes the operating system to access the program data portion.

13. The method of claim 9, wherein comparing the geolocation resource claims with the communicated geolocation of the computing device includes determining if the geolocation of the computing device is identified as being located within a geographical boundary defined by the geolocation resource claims.

14. The of claim 10, wherein a denial of geolocation verification of the computer object prevents the operating system from accessing the program data portion of the computer object.

15. The method of claim 10, wherein the computer object is at least one of a computer file, a storage device containing a computer file, a programming interface that receives and emits a computer file from a storage device, and a database query interface.

16. A system for geolocation verification comprising:

a computing device including a processor and a memory having stored therein computer-executable instructions;

an operating system installed on the computing device, wherein at least one of, the operating system includes a unique system ID, and the computing device includes a unique device ID;

a Global Positioning System interface in communication with at least one of the computing device and the operating system, capable of providing a geolocation for the at least one of the computing device and the operating system;

a network interface for transmitting the geolocation of the at least one of the computing device and the operating system, and the at least one of the system ID and the device ID, to a domain controller in communication with a data repository; and

a computer object associated with the operating system, the computer object including a unique object ID and one or more geographic object resource claims, the geographic object resource claims including one or more geolocation boundaries within which the computer object is authorized to be operated.

17. The system of claim 16, wherein the computer object includes an authentication data portion and a program data portion, the authentication data portion being accessible independently from accessing the program data portion, with the program data portion being inaccessible prior to positive geolocation verification by an authorizing entity communicable with the operating system.

18. The system of claim 17, wherein geolocation verification is receivable from a domain controller in communication with the operating system, and wherein a positive geolocation verification is receivable by the operating system when the geolocation provided by the at least one of the computing device and the operating system is a geolocation that is situated within the boundaries established by the geographic object resource claims.

19. The system of claim 18, wherein the computer object is at least one of a computer file, a storage device containing a computer file, an interface that receives and emits a computer file from a storage device, and a database.

20. The system of claim 16, wherein the operating system is a guest operating system operating on a virtual machine in communication with the computing device.

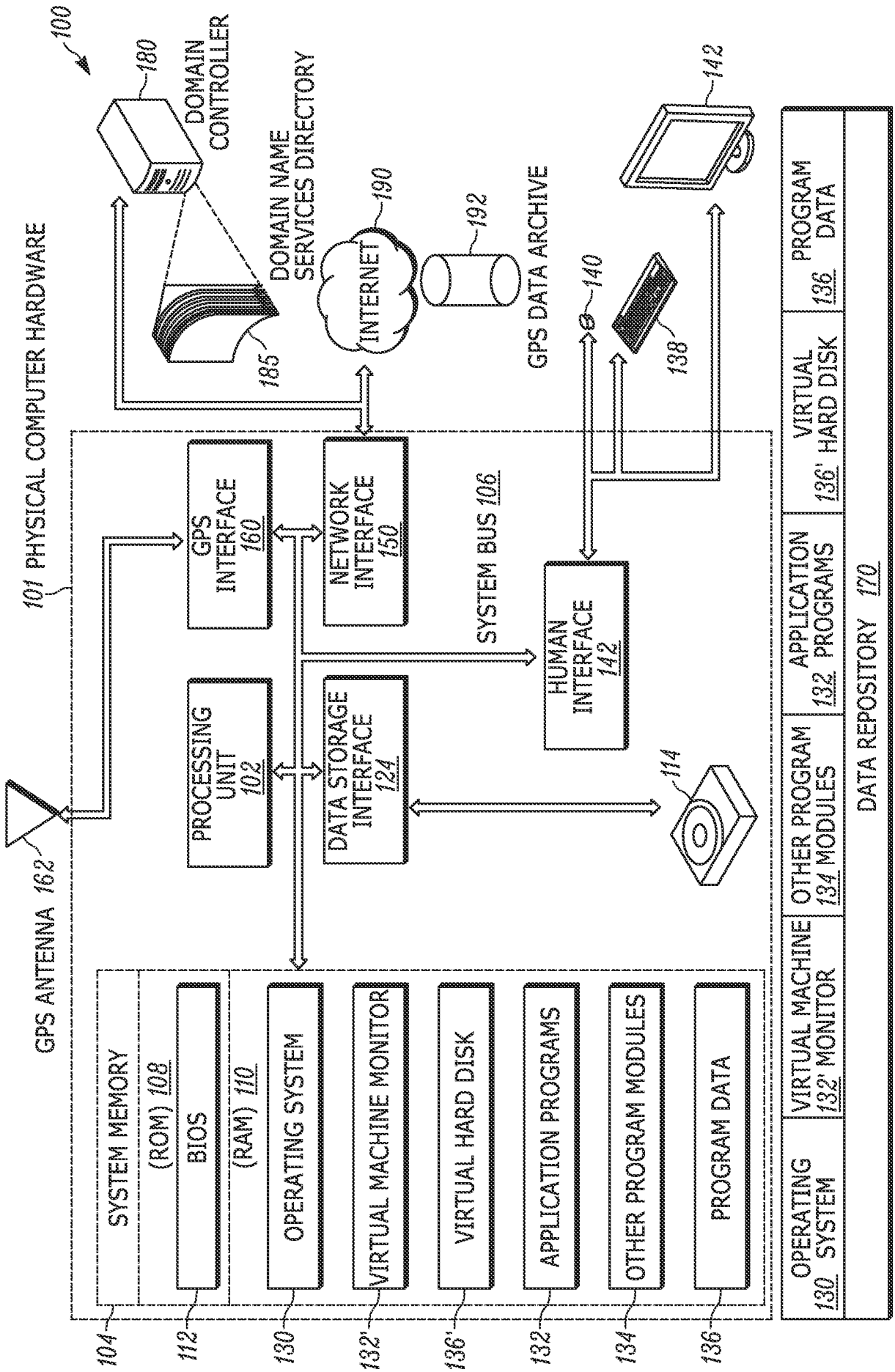

100
101 PHYSICAL COMPUTER HARDWARE
180
DOMAIN CONTROLLER
DOMAIN NAME SERVICES DIRECTORY
185
190
INTERNET
192
GPS DATA ARCHIVE
142
140
138
GPS ANTENNA 162
GPS INTERFACE 160
NETWORK INTERFACE 150
SYSTEM BUS 106
PROCESSING UNIT 102
DATA STORAGE INTERFACE 124
HUMAN INTERFACE 142
114
VIRTUAL HARD DISK 136' HARD DISK
PROGRAM DATA 136
APPLICATION 132 PROGRAMS
OTHER PROGRAM 134 MODULES
VIRTUAL MACHINE 132' MONITOR
OPERATING 130 SYSTEM
DATA REPOSITORY 170
SYSTEM MEMORY
(ROM) 108
BIOS
(RAM) 110
OPERATING SYSTEM
VIRTUAL MACHINE MONITOR
VIRTUAL HARD DISK
APPLICATION PROGRAMS
OTHER PROGRAM MODULES
PROGRAM DATA
104
112
130
132'
136'
132
134
136

Figure 1

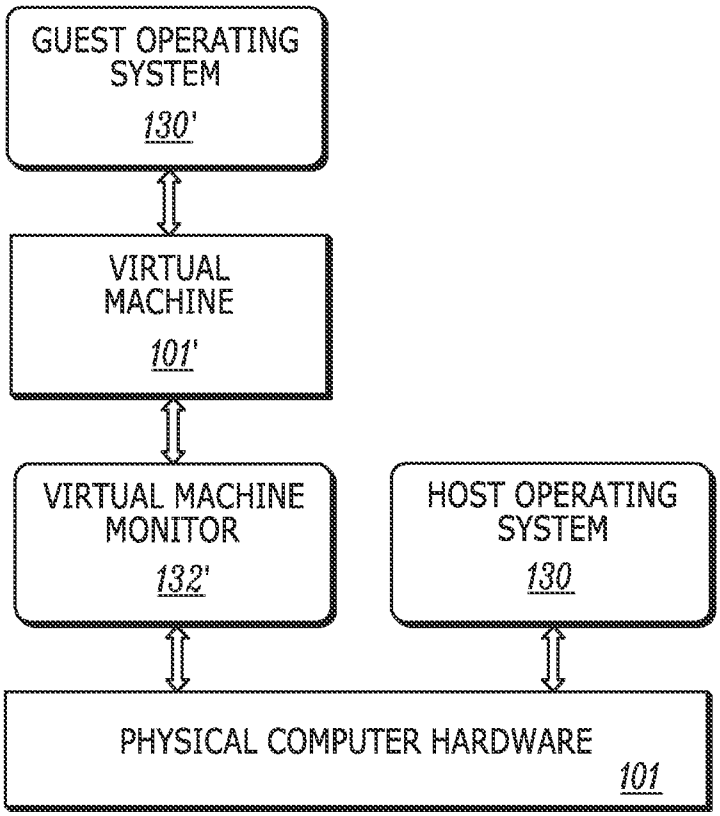


Figure 2

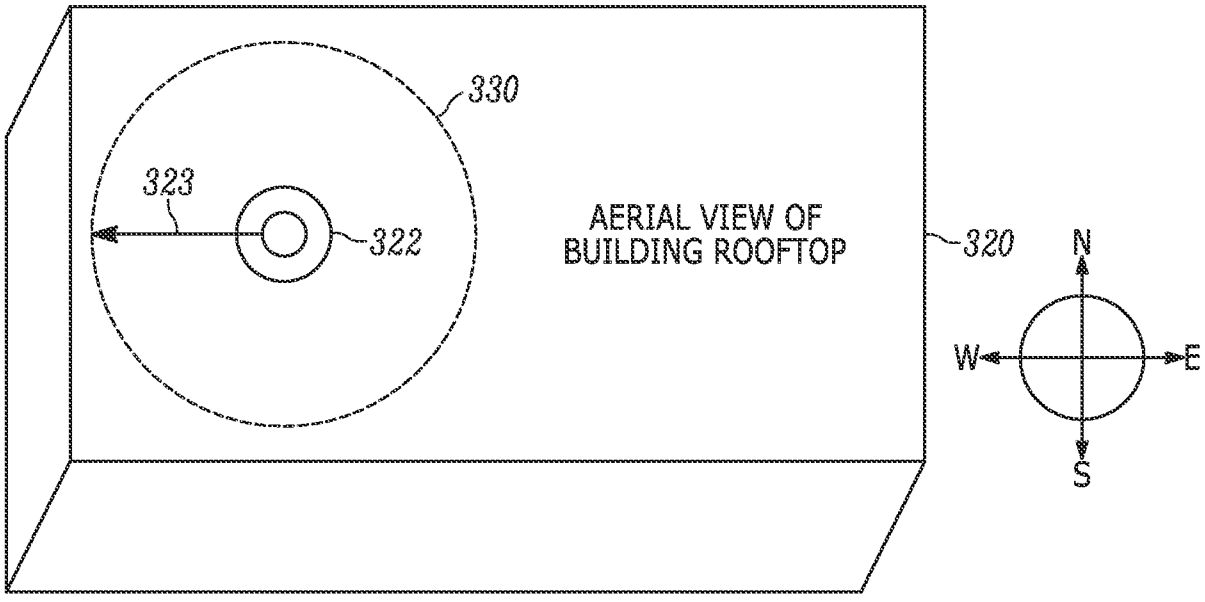


Figure 3A

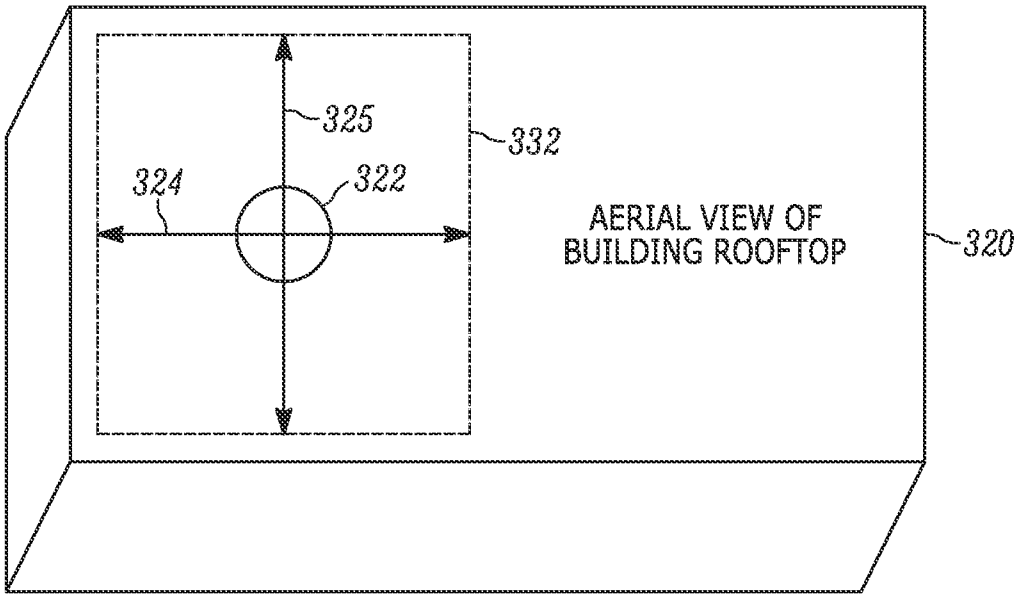


Figure 3B

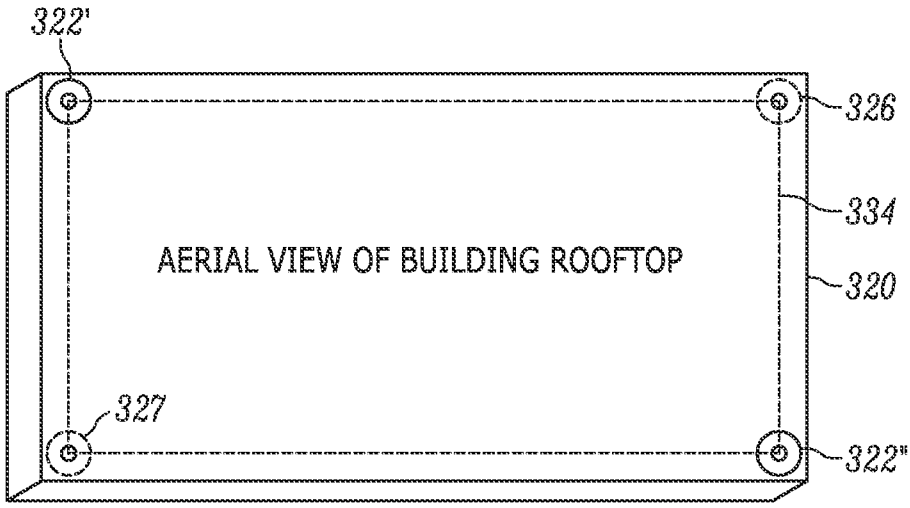


Figure 3C

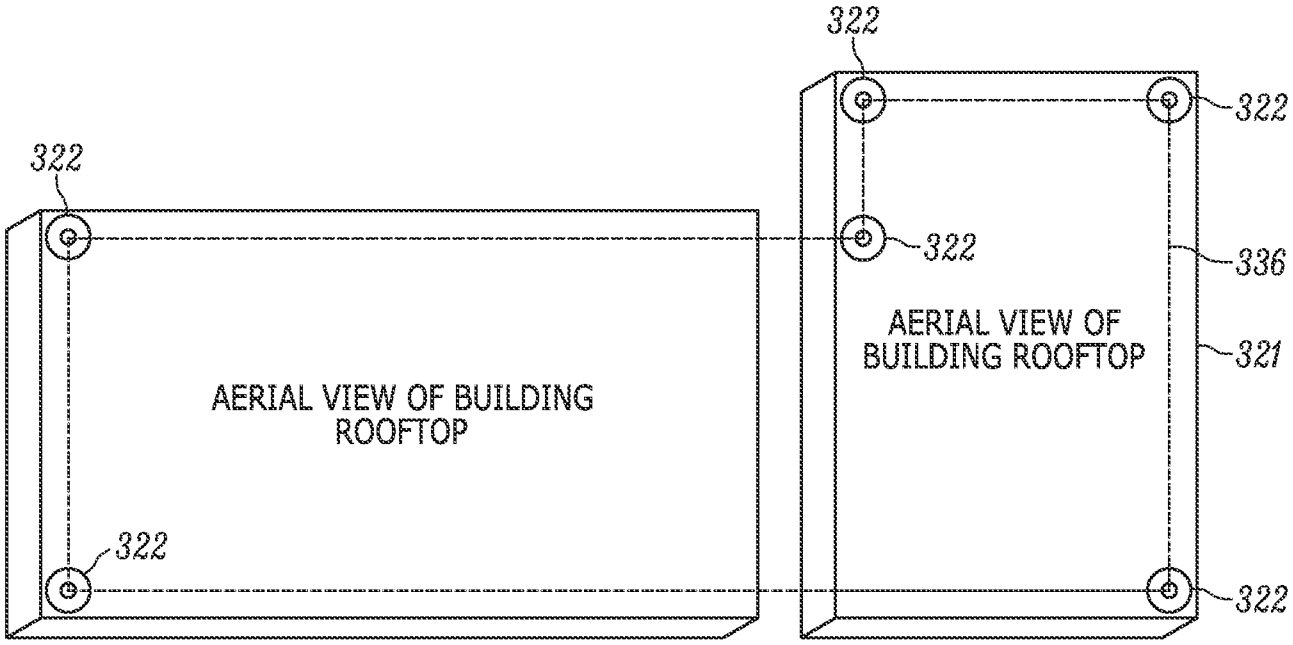


Figure 3D

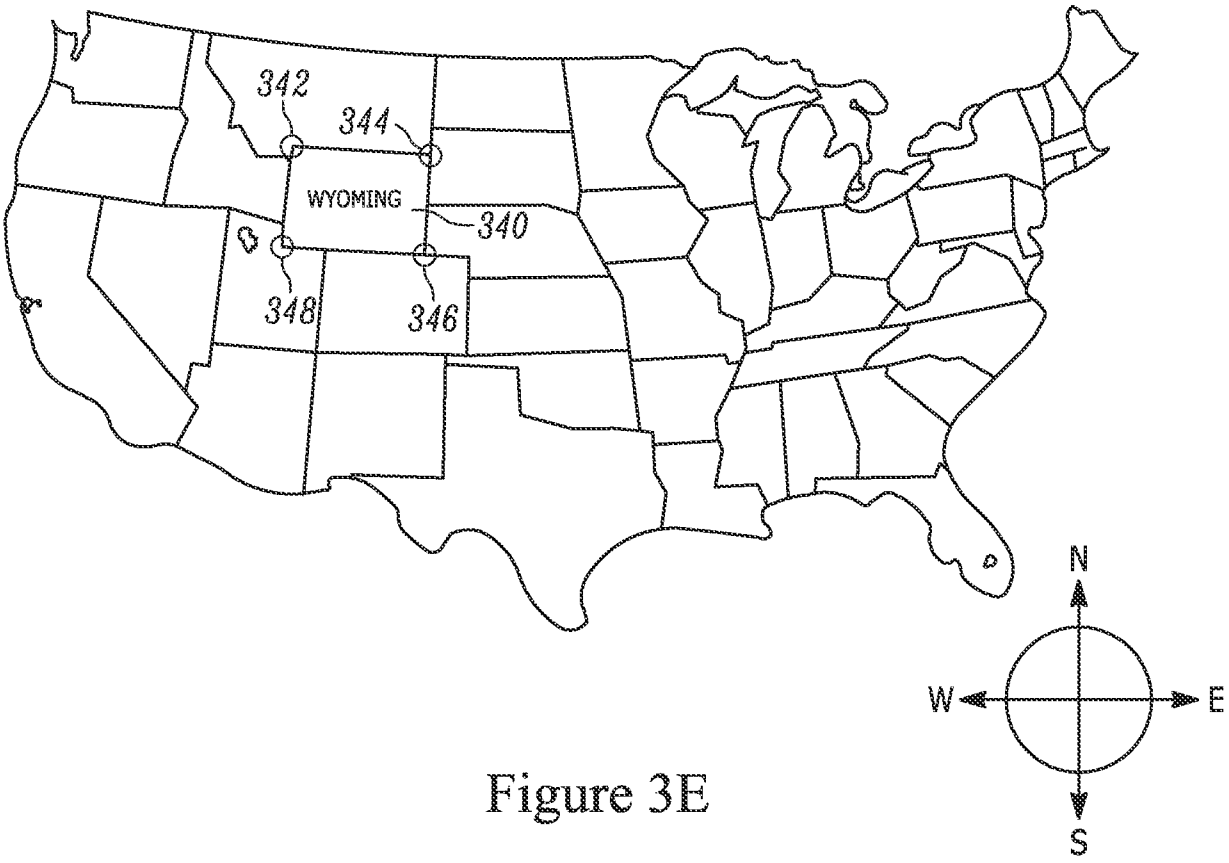


Figure 3E

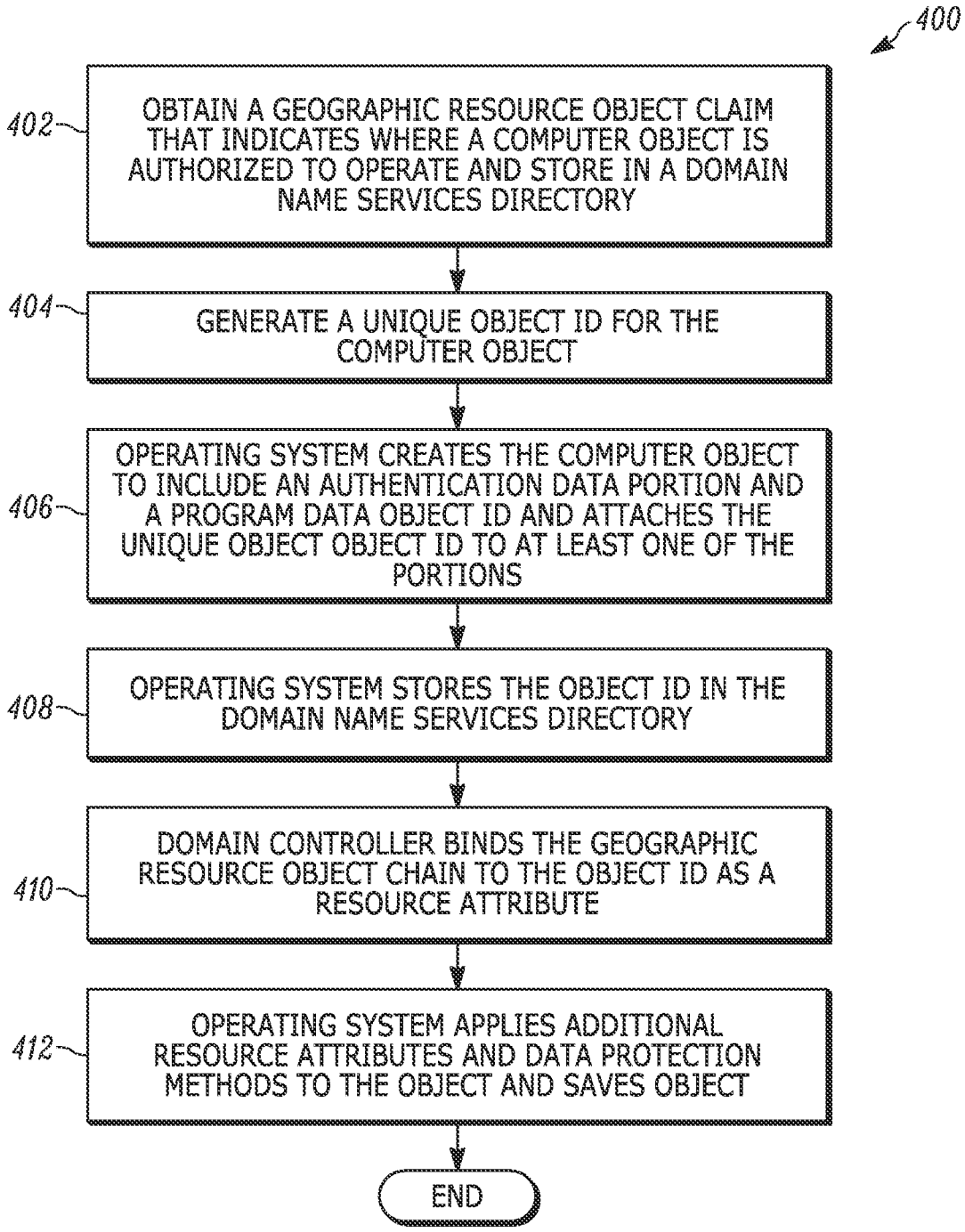

400
402
OBTAIN A GEOGRAPHIC RESOURCE OBJECT CLAIM THAT INDICATES WHERE A COMPUTER OBJECT IS AUTHORIZED TO OPERATE AND STORE IN A DOMAIN NAME SERVICES DIRECTORY
404
GENERATE A UNIQUE OBJECT ID FOR THE COMPUTER OBJECT
406
OPERATING SYSTEM CREATES THE COMPUTER OBJECT TO INCLUDE AN AUTHENTICATION DATA PORTION AND A PROGRAM DATA OBJECT ID AND ATTACHES THE UNIQUE OBJECT OBJECT ID TO AT LEAST ONE OF THE PORTIONS
408
OPERATING SYSTEM STORES THE OBJECT ID IN THE DOMAIN NAME SERVICES DIRECTORY
410
DOMAIN CONTROLLER BINDS THE GEOGRAPHIC RESOURCE OBJECT CHAIN TO THE OBJECT ID AS A RESOURCE ATTRIBUTE
412
OPERATING SYSTEM APPLIES ADDITIONAL RESOURCE ATTRIBUTES AND DATA PROTECTION METHODS TO THE OBJECT AND SAVES OBJECT
END

Figure 4

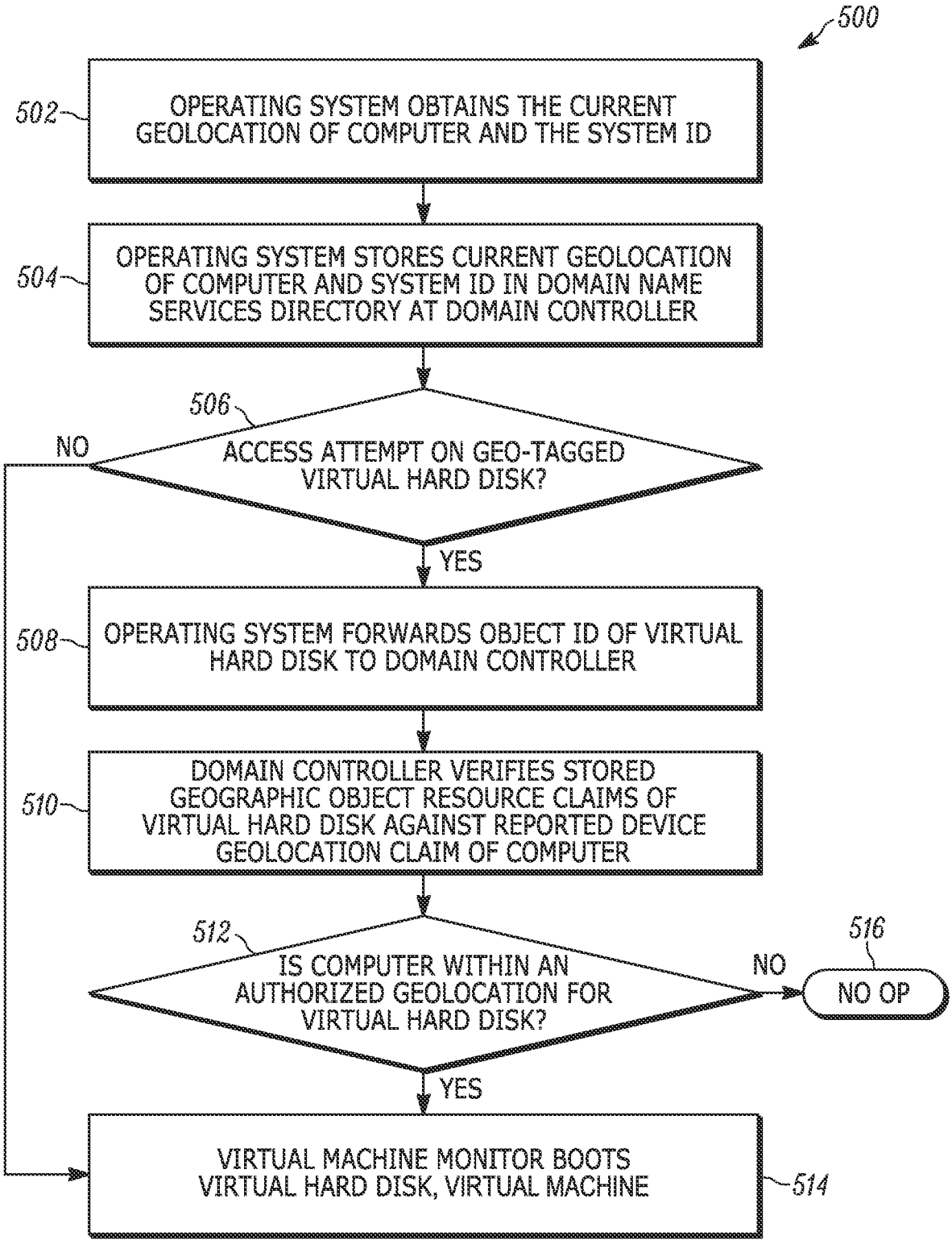

500
502
OPERATING SYSTEM OBTAINS THE CURRENT GEOLOCATION OF COMPUTER AND THE SYSTEM ID
504
OPERATING SYSTEM STORES CURRENT GEOLOCATION OF COMPUTER AND SYSTEM ID IN DOMAIN NAME SERVICES DIRECTORY AT DOMAIN CONTROLLER
506
ACCESS ATTEMPT ON GEO-TAGGED VIRTUAL HARD DISK?
NO
YES
508
OPERATING SYSTEM FORWARDS OBJECT ID OF VIRTUAL HARD DISK TO DOMAIN CONTROLLER
510
DOMAIN CONTROLLER VERIFIES STORED GEOGRAPHIC OBJECT RESOURCE CLAIMS OF VIRTUAL HARD DISK AGAINST REPORTED DEVICE GEOLOCATION CLAIM OF COMPUTER
512
IS COMPUTER WITHIN AN AUTHORIZED GEOLOCATION FOR VIRTUAL HARD DISK?
NO
516
NO OP
YES
VIRTUAL MACHINE MONITOR BOOTS VIRTUAL HARD DISK, VIRTUAL MACHINE
514

Figure 5

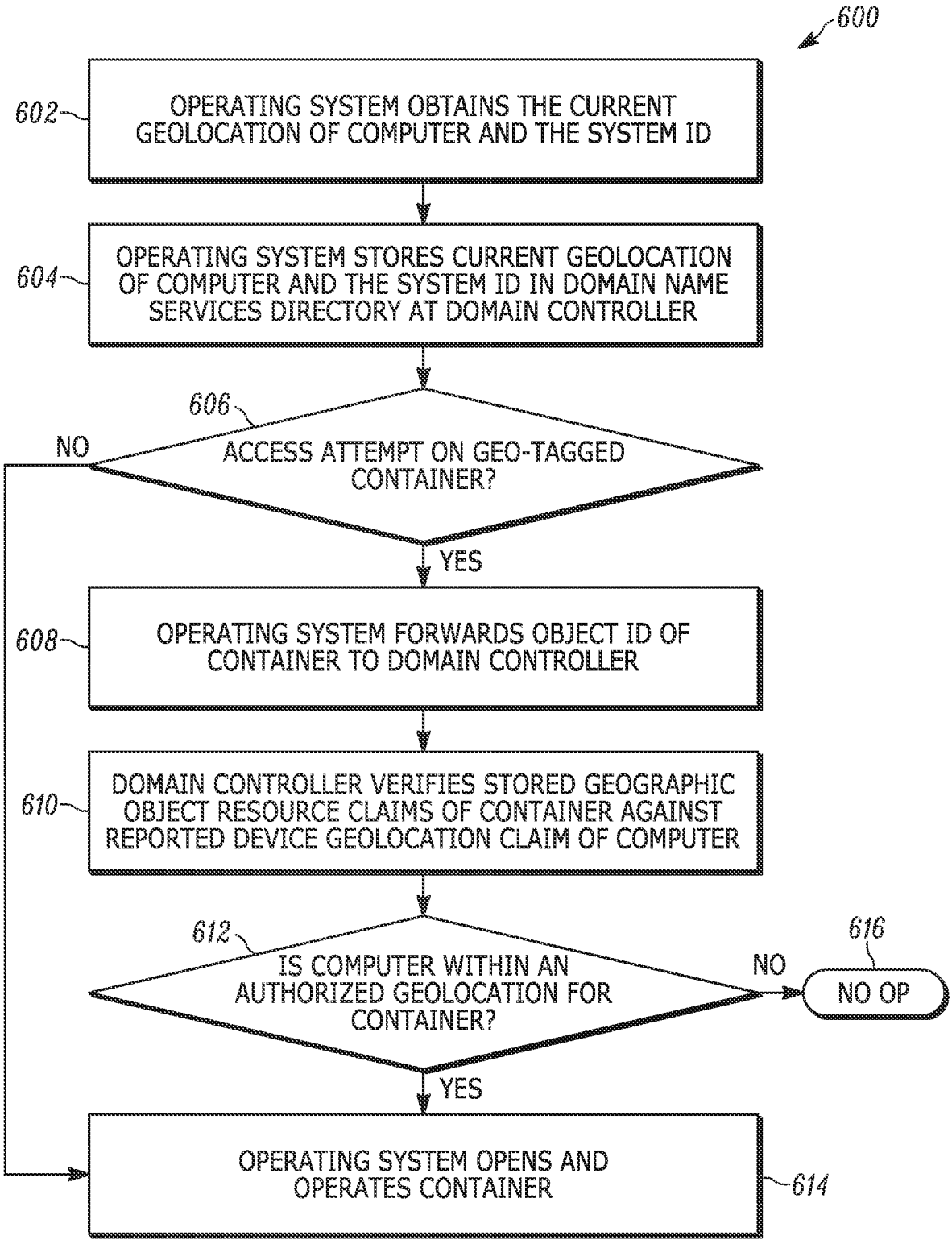

600
602
OPERATING SYSTEM OBTAINS THE CURRENT GEOLOCATION OF COMPUTER AND THE SYSTEM ID
604
OPERATING SYSTEM STORES CURRENT GEOLOCATION OF COMPUTER AND THE SYSTEM ID IN DOMAIN NAME SERVICES DIRECTORY AT DOMAIN CONTROLLER
606
ACCESS ATTEMPT ON GEO-TAGGED CONTAINER?
NO
YES
608
OPERATING SYSTEM FORWARDS OBJECT ID OF CONTAINER TO DOMAIN CONTROLLER
610
DOMAIN CONTROLLER VERIFIES STORED GEOGRAPHIC OBJECT RESOURCE CLAIMS OF CONTAINER AGAINST REPORTED DEVICE GEOLOCATION CLAIM OF COMPUTER
612
IS COMPUTER WITHIN AN AUTHORIZED GEOLOCATION FOR CONTAINER?
NO
616
NO OP
YES
OPERATING SYSTEM OPENS AND OPERATES CONTAINER
614

Figure 6

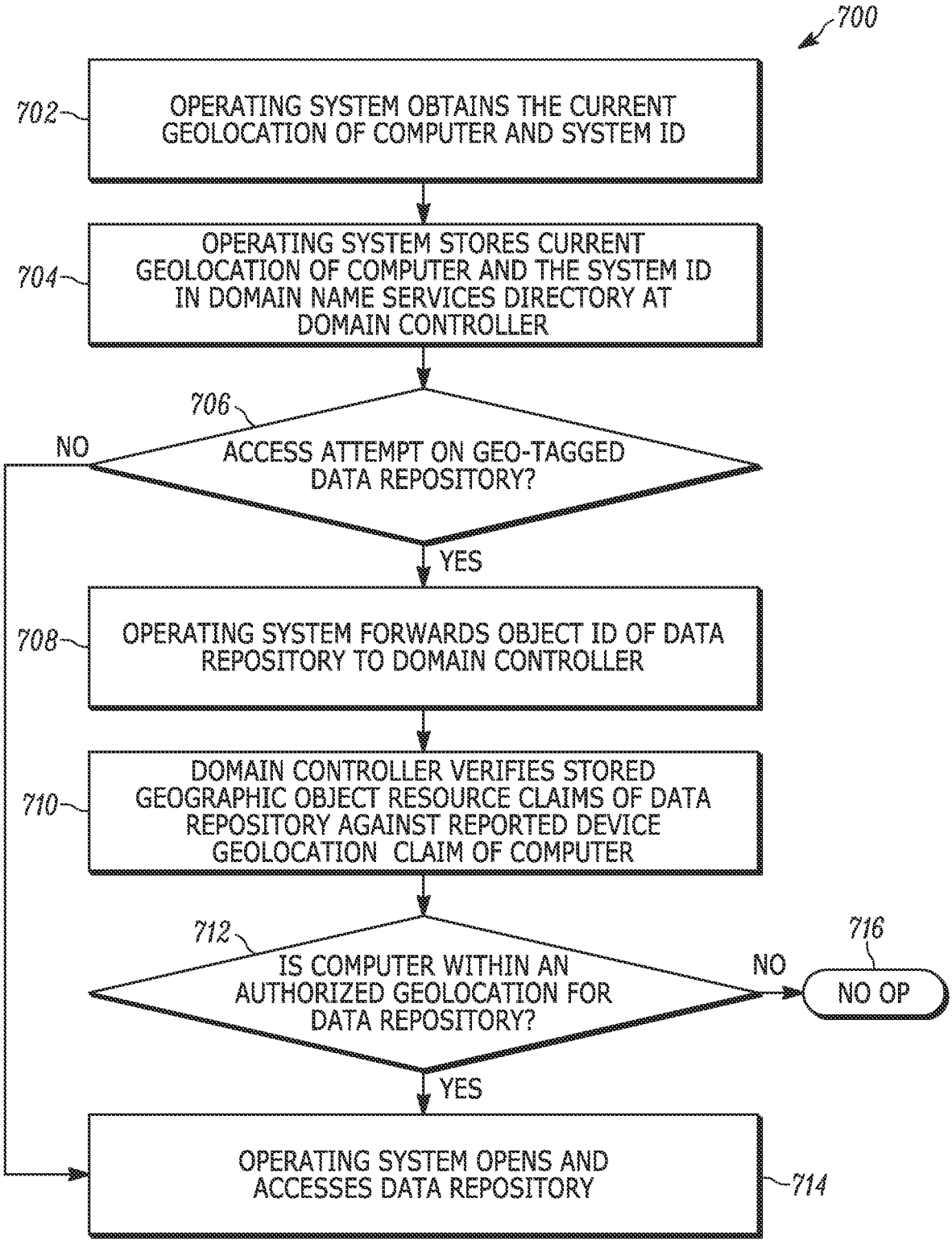

700
702
OPERATING SYSTEM OBTAINS THE CURRENT GEOLOCATION OF COMPUTER AND SYSTEM ID
704
OPERATING SYSTEM STORES CURRENT GEOLOCATION OF COMPUTER AND THE SYSTEM ID IN DOMAIN NAME SERVICES DIRECTORY AT DOMAIN CONTROLLER
706
ACCESS ATTEMPT ON GEO-TAGGED DATA REPOSITORY?
NO
YES
708
OPERATING SYSTEM FORWARDS OBJECT ID OF DATA REPOSITORY TO DOMAIN CONTROLLER
710
DOMAIN CONTROLLER VERIFIES STORED GEOGRAPHIC OBJECT RESOURCE CLAIMS OF DATA REPOSITORY AGAINST REPORTED DEVICE GEOLOCATION CLAIM OF COMPUTER
712
IS COMPUTER WITHIN AN AUTHORIZED GEOLOCATION FOR DATA REPOSITORY?
NO
716
NO OP
YES
OPERATING SYSTEM OPENS AND ACCESSES DATA REPOSITORY
714

Figure 7

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2015/058102

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F21/53 G06F21/64
ADD.

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)
G06F

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	Erin K Banks ET AL: "Trusted Geolocation in the Cloud: Proof of Concept Implementation (Draft)", NIST Interagency Report 7904, 31 December 2012 (2012-12-31), XP055207483, Retrieved from the Internet: URL:http://csrc.nist.gov/publications/drafts/ir7904/draft_nistir_7904.pdf [retrieved on 2015-08-12] page 10 - page 13; figures 16,17 page 24 page 33 ----- -/--	1-20

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

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

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

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

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

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

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

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

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

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
26 January 2016	03/02/2016

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Koblitz, Birger

2

Form PCT/ISA/210 (second sheet) (April 2005)

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	Vmware: "VMware KB: Looking up Managed Object Reference (MoRef) in vCenter Server", , 30 July 2014 (2014-07-30), XP055242684, Retrieved from the Internet: URL:http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=1017126 [retrieved on 2016-01-19] the whole document -----	1-20
A	Vmware: "VMware KB: Editing virtual machine and host custom attributes, notes and annotations", , 29 August 2014 (2014-08-29), XP055242781, Retrieved from the Internet: URL:http://kb.vmware.com/selfservice/microsites/search.do?language=en_US&cmd=displayKC&externalId=1005720 [retrieved on 2016-01-19] the whole document -----	1-20
A	US 2012/159156 A1 (BARHAM PAUL [US] ET AL) 21 June 2012 (2012-06-21) the whole document -----	1-20

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2015/058102

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2012159156 A1	21-06-2012	AR 084211 A1	02-05-2013
		CN 102609662 A	25-07-2012
		EP 2656270 A2	30-10-2013
		JP 2014503909 A	13-02-2014
		KR 20130129224 A	27-11-2013
		TW 201232324 A	01-08-2012
		US 2012159156 A1	21-06-2012
		WO 2012087853 A2	28-06-2012

(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 107111714 A
(43)申请公布日 2017.08.29

(21)申请号 201580058741.8

(22)申请日 2015.10.29

(30)优先权数据
62/073,008 2014.10.30 US

(85)PCT国际申请进入国家阶段日
2017.04.27

(86)PCT国际申请的申请数据
PCT/US2015/058102 2015.10.29

(87)PCT国际申请的公布数据
WO2016/069915 EN 2016.05.06

(71)申请人 新科恩斯卡莱有限责任公司
地址 美国威斯康星州

(72)发明人 索恩·沃巴

(74)专利代理机构 北京安信方达知识产权代理有限公司 11262
代理人 陆建萍 郑霞

(51)Int.Cl.
G06F 21/53(2013.01)
G06F 21/64(2013.01)

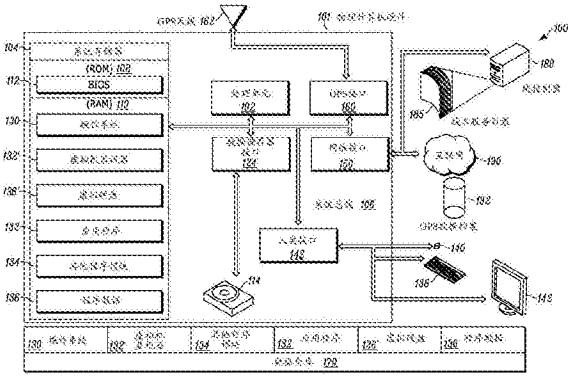
权利要求书3页 说明书9页 附图8页

(54)发明名称

用于资源的地理位置验证的方法和系统

(57)摘要

一种地理位置验证的方法，包括获取操作系统的地理位置，生成关于安装的操作系统的唯一系统ID，以及将操作系统的地理位置和系统ID传输到数据仓库。方法还包括：接收开启与操作系统相关联的计算机对象的部署或授予对其访问的请求，识别计算机对象是否需要地理位置验证，然后识别与计算机对象相关联的对象ID，并将对象ID、操作系统的地理位置和系统ID中的每个发送到域控制器以用于评估。方法还包括搜索数据仓库，以识别与对象ID相关联的一个或多个地理位置对象资源声明，以及将地理位置资源声明与所发送的操作系统的地理位置进行比较。



CN 107111714 A

1.一种方法,包括:

提供操作系统,所述操作系统与至少一个处理器和通信地耦合到所述至少一个处理器的至少一个存储器进行通信,所述存储器具有被储存于其中的计算机可执行指令;

借助于通过所述计算设备对所述计算机可执行指令的执行,创建计算机对象,所述计算机对象包括授权数据部分和程序数据部分,所述授权数据部分能够由应用程序独立于对所述计算机对象的所述程序数据部分的访问来访问;

生成所述计算机对象的唯一对象ID,并将所述唯一对象ID写入所述授权数据部分和所述程序数据部分中;

获取所述计算机对象的一个或多个地理对象资源声明,其中,所述地理对象资源声明包括所述计算机对象被授权进行操作的一个或多个地理位置;以及

将所述唯一对象ID和所述一个或多个地理对象资源声明发送给授权实体。

2.如权利要求1所述的方法,其中,所述计算机可执行指令还由所述操作系统执行以包括:

获取所述计算机对象;以及

仅读取所述授权数据部分而不访问所述程序数据部分。

3.如权利要求2所述的方法,其中,所述计算机可执行指令还由所述操作系统执行,以包括与全球定位卫星接口进行通信,从而识别所述操作系统当前正在进行操作的地理位置。

4.如权利要求3所述的方法,其中,所述计算机可执行指令还由所述操作系统执行以包括:

传送以下内容:a)所述计算机对象的对象ID,b)运行所述操作系统的计算机的设备ID和与所述操作系统相关联的系统ID中的至少一个,以及c)由GPS对所述授权实体识别出的地理位置;以及

响应于所述传送,接收来自所述授权实体的肯定的地理位置验证或否定的地理位置验证,其中,在接收到肯定的地理位置验证时,所述操作系统被授权访问所述计算机对象中的所述程序数据部分,以及其中,在接收到否定的地理位置验证时,所述操作系统不被授权访问所述程序数据部分。

5.如权利要求1所述的方法,其中,所述计算机对象是计算机文件、包含计算机文件的储存设备、从储存设备接收并发出计算机文件的接口和数据库中的至少一个。

6.如权利要求1所述的方法,其中,所述计算机对象是被安装在物理服务器上的虚拟机的虚拟硬盘映像。

7.如权利要求1所述的方法,其中,所述操作系统在物理服务器上被安装和操作。

8.如权利要求1所述的方法,其中,所述操作系统在与物理服务器进行通信的虚拟机上被安装和操作。

9.一种地理位置验证的方法,包括:

获取被安装在计算设备上的操作系统的地理位置,所述计算设备包括处理器和存储器,所述存储器具有储存于其中的计算机可执行指令;

生成关于所安装的操作系统的唯一系统ID;

将所述操作系统的所述地理位置和所述系统ID传输到与所述操作系统进行通信的域

控制器的数据仓库；

在所述计算设备处，接收对开启与所述操作系统相关联的计算机对象的部署或授予对所述计算机对象的访问权限中的至少一个的请求；

识别所述计算机对象是否需要地理位置验证，并且如果需要验证，则识别与所述计算机对象相关联的对象ID，并将所述对象ID、所述操作系统的所述地理位置和所述系统ID中的每个发送到所述域控制器以用于评估；

执行地理位置验证分析，所述地理位置验证分析包括使用所述对象ID搜索所述数据仓库，以识别与所述对象ID相关联的一个或多个地理位置对象资源声明，其中，所述地理对象资源声明包括所述计算机对象被授权进行操作的一个或多个地理位置；以及

将所述地理位置资源声明与所发送的所述操作系统的地理位置进行比较，以提供对所述计算机对象的地理位置验证的确认或否定。

10.如权利要求9所述的方法，其中，所述计算机对象包括授权数据部分和程序数据部分，所述授权数据部分能够由应用程序独立于对所述计算机对象的所述程序数据部分的访问来访问。

11.如权利要求10所述的方法，其中，识别所述计算机对象是否需要地理位置验证不包括访问所述程序数据部分，以及其中，仅地理位置验证的确认授权所述操作系统访问所述程序数据部分。

12.如权利要求10所述的方法，其中，识别所述计算机对象ID不包括访问所述程序数据部分，以及其中，地理位置验证的确认授权所述操作系统访问所述程序数据部分。

13.如权利要求9所述的方法，其中，将所述地理位置资源声明与所发送的所述计算设备的地理位置进行比较包括确定所述计算设备的地理位置是否被识别为位于由所述地理位置资源声明界定的地理边界内。

14.如权利要求10所述的方法，其中，所述计算机对象的地理位置验证的否定防止所述操作系统访问所述计算机对象的所述程序数据部分。

15.如权利要求10所述的方法，其中，所述计算机对象是计算机文件、包含计算机文件的储存设备、从储存设备接收和发出计算机文件的编程接口和数据库查询接口中的至少一个。

16.一种用于地理位置验证的系统，包括：

计算设备，所述计算设备包括处理器和存储器，所述存储器具有储存于其中的计算机可执行指令；

操作系统，所述操作系统被安装在所述计算设备上，其中，所述操作系统包括唯一系统ID以及所述计算设备包括唯一设备ID中的至少一者成立；

全球定位系统接口，所述全球定位系统接口与所述计算设备和所述操作系统中的至少一个进行通信，所述全球定位系统接口能够为所述计算设备和所述操作系统中的至少一个提供地理位置；

网络接口，所述网络接口用于将所述计算设备和所述操作系统中的所述至少一个的所述地理位置以及所述系统ID和所述设备ID中的至少一个传输到与数据仓库进行通信的域控制器；以及

与所述操作系统相关联的计算机对象，所述计算机对象包括唯一对象ID以及一个或多

个地理对象资源声明，所述地理对象资源声明包括在其中所述计算机对象被授权进行操作的一个或多个地理位置边界。

17.如权利要求16所述的系统，其中，所述计算机对象包括授权数据部分和程序数据部分，所述授权数据部分能够独立于对所述程序数据部分的访问而被访问，以及在通过能够与所述操作系统进行通信的授权实体的肯定的地理位置验证之前，所述程序数据部分是不可访问的。

18.如权利要求17所述的系统，其中，地理位置验证能够从与所述操作系统进行通信的域控制器接收，以及其中，当由所述计算设备和所述操作系统中的所述至少一个提供的地理位置是位于由所述地理对象资源声明建立的边界内的地理位置时，肯定的地理位置验证能够由所述操作系统接收。

19.如权利要求18所述的系统，其中，所述计算机对象是计算机文件、包含计算机文件的储存设备、从储存设备接收并发出计算机文件的接口和数据库中的至少一个。

20.如权利要求16所述的系统，其中，所述操作系统是在与所述计算设备进行通信的虚拟机上操作的客户操作系统。

用于资源的地理位置验证的方法和系统

[0001] 相关申请的交叉引用

[0002] 本申请要求2014年10月30日提交的美国临时申请号为62/073,008的优先权，该申请通过引用以其整体并入本文。

[0003] 领域

[0004] 方法和系统大体上涉及各种类型的资源的地理位置验证领域。

[0005] 背景

[0006] 数据中心是应用软件和运行在软件上的客户数据所在的位置。基于云的IT服务的供应商必须保存他们在任何特定时间复制客户数据时的透明度，以保护免于故障或局部灾难。如果数据中心出于任何原因停止运作，若应用软件和在该应用软件上运行的客户数据也可从第二个或可能第三数据中心获得，则客户数据将不会丢失。并且假设它工作足够平顺，客户在这样的故障转移(failover)发生时可能甚至不会被通知。根据特定服务，故障转移可能根本不会导致任何服务中断。

[0007] 全球企业已经利用互联网和基于云计算服务以及数据中心来建立私有通信网络，并从全球技术中获得了效率。由于这样的全球化，近年来，许多国家已经发布了地理位置规则，其限制了公司可如何跨边界(包括通过这些私有网络)处理和传输它们客户的数据。某些实体要求将特定类型的数据(例如政府数据)、员工数据或电信业务数据储存在有限的地理边界内，并且在一些情况下，这样的数据甚至可能不从地理边界的外部进行访问。

[0008] 地理位置由一组坐标指定，该组坐标表示纬度、经度和海拔，它们是地理坐标系的主要元素。地球表面上一点的纬度(φ或phi)是赤道平面与穿过该点并穿过(或接近)地球中心的直线之间的角度。赤道将地球分为北半球和南半球。地球表面上一点的经度(λ或lambda)是从参考子午线到穿过该点的另一个子午线向东或向西的角度。国际公认的参考本初子午线穿过英格兰的格林威治的一点，并确定了适当的东半球和西半球。地球表面上的一点的海拔通常是其相对于海平面的高度；而海拔高度用于海平面以上的点时，诸如飞行中的飞机或轨道上的航天器，深度用于海平面以下的点。

[0009] 地球上的定位的地理位置可根据像Wi-Fi接入点和蜂窝塔的信标获得、根据设备的ID地址获得，或者其可能来自其他来源，诸如全球导航卫星系统(GNSS)或全球定位系统(GPS)设备。地理位置信息的准确性取决于来源，并且与设备、计算机或资源的实际位置可在以下示例性范围内不同：

[0010] • GPS：在大约10米内

[0011] • Wi-Fi：介于大约30米和500米之间

[0012] • 蜂窝塔：介于大约300米和3,000米之间

[0013] • IP地址：介于大约1,000米和5,000米之间

[0014] 在处理信息技术操作时，通常使用一种或多种类型的声明的属性。声明是关于用户、设备、计算机或资源的唯一信息片段。这些是非常常见的属性，其可以作为域名服务目录中的计算机对象的特性来发现，像用户的功能标题、组织部门或办公室位置之类的，是可被定义的声明。数据文件的业务影响分类或计算机的健康状况也是如此。计算机对象或实

体可涉及一个以上的声明，并且声明的任何组合可用于授权对资源的访问。以下示例性类型的声明通常在可商购的域名服务目录中是可用的：

[0015] • 用户声明：与特定用户相关联的属性。

[0016] • 设备声明：与特定计算机对象相关联的属性。

[0017] • 资源属性：被标记用在授权决策中的全局资源特性。

[0018] 声明通常在域名服务目录中进行保护，该域名服务目录在域控制器及其代理上被操作并由其发布。这是为了防止他们被未经授权的人员篡改，并且仅对正确授权的用户、设备和计算机能访问。声明使管理员可以做出关于可包含在用于IT操作的表达、规则和策略中的用户、设备、计算机和资源的精确的组织或企业范围的陈述。

[0019] 简述

[0020] 在至少一些实施例中，用于资源的地理位置验证的方法和系统针对地理位置数据(例如，GPS信号)的使用，以确定计算机对象(诸如虚拟硬盘)是否可被部署，并且其虚拟机是否可由位于特定地理位置的物理服务器操作。此外，在至少一些实施例中，域控制器通过解析操作物理服务器的当前地理位置是否在可以操作虚拟机的地理区域内进行仲裁。

[0021] 示例性实施例包括用于确定物理服务器(包括通用计算机、GPS系统接口、连接电缆和GPS天线)的当前地理位置的方法和系统。为了基于物理服务器的当前地理位置获取用于部署虚拟硬盘的授权，物理服务器采集其当前的地理位置和待部署的虚拟硬盘的标识，并将该信息发送到域控制器。域控制器执行验证过程，并将肯定或否定的评估结果发送到物理服务器，以指示虚拟硬盘中所包含的虚拟机的部署是否可以进行。在至少一些实施例中，操作系统(用于物理服务器和虚拟机二者)能够使用相同的方法和系统进行自我验证。操作系统将其当前的地理位置及其自己的标识转发给域控制器。当接收到令人满意的结果以继续进行时，它可继续超出初始启动阶段的操作。

[0022] 在至少一些实施例中，用于资源的地理位置验证的方法和系统涉及一种方法，该方法包括：提供与至少一个处理器和通信地耦合到该至少一个处理器的至少一个存储器进行通信的操作系统，该存储器具有被储存在其中的计算机可执行指令；借助于通过计算设备执行该计算机可执行指令来创建计算机对象，该计算机对象包括授权数据部分和程序数据部分，该授权数据部分可独立于对该计算机对象的程序数据部分的访问由应用程序访问；生成计算机对象的唯一对象ID，并将该唯一对象ID写入授权数据部分和程序数据部分；获取计算机对象的一个或多个地理对象资源声明，其中，该地理对象资源声明包括计算机对象被授权以进行操作的一个或多个地理位置；以及将唯一对象ID以及一个或多个地理对象资源声明发送给授权实体。

[0023] 在至少一些附加的实施例中，用于资源的地理位置验证的方法和系统涉及一种方法，该方法包括：获取安装在计算设备上的操作系统的地理位置，该计算设备包括处理器和存储器，该存储器具有储存于其中的计算机可执行指令；为所安装的操作系统生成唯一的系统ID；将操作系统的地理位置和系统ID传输到与操作系统进行通信的域控制器的数据仓库；在计算设备处，接收开启与操作系统相关联的计算机对象的部署或授予对其的访问权限中的至少一个的请求；识别计算机对象是否需要地理位置验证，并且如果需要验证，则识别与计算机对象相关联的对象ID，并将对象ID、操作系统的地理位置和系统ID中的每个发送到域控制器以用于评估；执行地理位置验证分析，包括使用对象ID来搜索数据仓库，以识

别与该对象ID相关联的一个或多个地理位置对象资源声明，其中，该地理对象资源声明包括计算机对象被授权以进行操作的一个或多个地理位置；以及将该地理位置资源声明与所发送的操作系统的地理位置进行比较，以提供对计算机对象的地理位置验证的确认或否定。

[0024] 在至少一些进一步的实施例中，用于资源的地理位置验证的方法和系统涉及包括被安装在计算设备上的操作系统的系统。计算设备包括处理器和存储器，该存储器具有储存于其中的计算机可执行指令。操作系统包括唯一系统ID以及计算设备包括唯一设备ID中的至少一项成立。全球定位系统接口与计算设备和操作系统中的至少一个进行通信，并且能够为计算设备和操作系统中的至少一个提供地理位置。网络接口将计算设备和操作系统中的至少一个的地理位置以及系统ID和设备ID中的至少一个传输到与数据仓库进行通信的域控制器。与操作系统相关联的计算机对象包括唯一对象ID以及一个或多个地理对象资源声明。地理对象资源声明包括计算机对象被授权以进行操作的一个或多个地理位置边界。

[0025] 在全面阅读了详细描述和所附权利要求之后，将理解和认识到该方法和系统的其它实施例、方面、特征、目标和优点。

[0026] 附图简述

[0027] 方法和系统的实施例参照附图被公开，并且仅用于说明的目的。方法和系统在其应用中不限于附图中所示的组分的构造或布置的细节。方法和系统能够是其他实施例或以其他各种方式被实践或执行。在附图中：

[0028] 图1图示了示例性计算环境和系统的框图；

[0029] 图2图示了软件和硬件的示例性架构的框图；

[0030] 图3A图示了建筑物内的一个地点的地理区域的圆形界定的边界；

[0031] 图3B图示了建筑物内的一个地点的地理区域的矩形界定的边界；

[0032] 图3C图示了建筑物内的一个地点的地理区域的另一矩形界定的边界；

[0033] 图3D图示了包括一个或多个建筑物的一个地点的地理区域的多边形界定的边界；

[0034] 图3E图示了管辖范围的地理区域的边界，更具体地，怀俄明州(美国)的地理区域的边界；

[0035] 图4图示了用于对要在一个或多个特定地理区域中操作的虚拟硬盘进行标记的示例性方法的流程图；

[0036] 图5图示了被标记为在一个或多个特定地理区域中操作的虚拟机的受控部署的示例性方法的流程图；

[0037] 图6图示了被标记为在一个或多个特定地理区域中操作的应用程序容器的受控部署的示例性方法的流程图；

[0038] 图7图示了表示对被标记以可在一个或多个特定地理区域内访问的数据仓库的受控访问的示例性方法的流程图。

[0039] 详细描述

[0040] 以下的详细描述参照了图示了本发明的示例性实施例的附图。然而，本发明的范围不限于这些实施例。因此，超出附图中所示的那些的实施例，诸如所示实施例的修改版本，可能仍然由所述地理验证的方法和系统所包括。

[0041] 说明书中对“一个实施例”、“实施例”、“示例实施例”等的参考指示所描述的实施例可包括特定的特征、结构或特性，但是每个实施例可不必包括该特定的特征、结构或特性。此外，这样的短语不一定指的是同一实施例。此外，当特定的特征、结构或特性结合实施例进行描述时，认为这是在相关领域的技术人员的知识范围之内，以结合无论是否明确描述的其它实施例实现这样的特征、结构或特性。

[0042] 图1图示了示例性计算环境和系统100的框图，以及图2图示了软件和硬件的示例性架构的框图。在至少一些实施例中，系统100包括计算机101。计算机101可以表示例如在其各自的数据中心中操作的企业或公司服务器，或由商业主机数据中心操作的服务器。然而，该示例并不旨在限制，并且计算机101也可在其他环境中操作，并且相关领域中的技术人员将会认识到，本文中所描述的方法和系统可以由不同于计算机101的各种各样的计算机来执行。计算机101可包括唯一地识别计算机101的设备ID，其中，设备ID可包括许多种类型的标识符，诸如序列号等。另外，操作系统130可被安装在计算机101上，其可包括当操作系统被安装在计算机101上时所创建的系统ID。在每个实例中，当操作系统被安装时，可创建关于其的新的唯一系统ID。

[0043] 参照图1，在至少一些实施例中，计算机101包括多个互连的硬件组件，该多个互连的硬件组件包括但不限于处理单元102、系统存储器104、数据储存接口124、网络接口150、人机接口142和GPS接口160。系统存储器104包括诸如ROM 108和RAM 110的易失性和/或非易失性存储器形式的计算机储存介质。包含有助于在诸如启动期间在计算机101内的元件之间传递信息的基本例程的基本输入/输出系统112(BIOS)通常被储存在ROM 108中。RAM 110通常包含可由处理单元102立即访问和/或当下正在其上被操作的数据和/或程序的模块。通过示例而非限制的方式，图1图示了操作系统130、应用程序132、其他程序模块134、程序数据136、虚拟机监视器132’和虚拟硬盘136’。应用程序132有时在本文中被称为可执行的二进制文件。

[0044] 计算机101还可包括其它可移动/不可移动的易失性/非易失性计算机储存介质。仅通过示例的方式，图1图示了从不可移动的非易失性磁性介质读取或写入其中的硬盘驱动器114。可用在示例性操作环境中的其他可移动/不可移动的易失性/非易失性计算机储存介质包括但不限于光盘，诸如CDROM或数字通用盘、磁带盒、闪存卡、数字录像带、固态RAM、固态ROM等。硬盘驱动器114通常通过数据储存接口124连接到系统总线106。计算机101被配置为从数据仓库读取和向其写入，该数据仓库通过通常连接到系统总线106的网络接口150在内联网和互联网中的至少一个上可到达。在至少一个实施例中，域名服务目录185用作内联网数据仓库，以及GPS数据档案192用作互联网数据仓库。本文中对域名服务目录185的使用的引用被理解为包括除域名服务目录185之外或代替其进行提供的其他数据仓库，包括但不限于GPS数据档案192。以下将更详细地描述计算机101的附加元件。

[0045] 如上所讨论的，用于资源的地理位置验证的方法和系统涉及查明计算机101的当前地理位置。更具体地，在至少一些实施例中，用于资源的地理位置验证的方法和系统包括查明和证明计算机或其操作系统运行在被授权的地理位置内。尽管在至少一个实施例中，全球定位系统(GPS)用作提供期望的地理位置数据的主要来源，而其他实施例可利用其他来源来获得地理位置数据，但特定的地理位置确定能够以若干不同的方式来执行。

[0046] 如图1中所示，包括计算机101中的一个或多个微处理器或微处理器核心的处理单

元102被设计成执行被储存在系统存储器104中的程序指令，以使GPS接口160通信地连接到GPS天线162，并执行其被指定功能。该接口便于GPS信号的接收和当前时间、纬度、经度的传送，以及便于将卫星PRN(伪随机噪声)传输到计算机101的操作系统130。在至少一个实施例中，GPS接口160包括由威斯康星州的伯灵顿市的Sync-n-Scale有限责任公司设计和制造的PCIe附加卡。然而，该示例并旨在限制，并且GPS接口160可包括任何传统的或后续开发的硬件，或者被设计为依赖于可类似的导航辅助系统并执行上述功能的硬件和软件的组合。

[0047] 除了获取特定计算机的地理位置之外，用于资源的地理位置验证的方法和系统还可包括将地理位置信息用作先决条件指示符，以用于启动和操作计算机对象，诸如以下详细讨论的虚拟机、独立的应用程序或容器。现在参照图2，示出了虚拟机监控器132'，其也被称为管理程序。管理程序是仿真能够运行操作系统的另一计算机的特殊类型的应用程序132。被仿真的计算机被称为虚拟机，例如虚拟机101'。虚拟机101'能够运行与计算机操作系统130相同的操作系统的操作系统130'。虽然没有详细讨论，但一个以上的虚拟机101'可经由虚拟机监视器132'来仿真。

[0048] 虚拟机监视器132'可使用虚拟硬盘136'中所储存的指令来创建和启动虚拟机101'。如以下更详细讨论的，计算机101或另一类似能力的设备可用于创建虚拟硬盘136'映像，并将其储存在单独的内联网或互联网的数据仓库中。在至少一些实施例中，虚拟硬盘136'是以遵循特定格式的单一计算机文件的形式所构建的特殊类型的程序数据136。该文件封装了能够承载一个或多个文件系统并支持标准盘和文件读写操作的仿真的储存设备。作为标准计算机文件，虚拟硬盘136'受到其他计算机文件在相同计算机上将遵守的储存、操作和访问策略的约束。这些策略被表达为设备声明和资源属性，并且被保存在诸如域名服务目录185的域名服务目录中，该域名服务目录可以是组织或企业范围的域名服务目录，或者其他类似的数据仓库。在创建时，出于安全的目的，虚拟硬盘136'将使用可用于管理程序132的经典数据保护方法而被加密地保护或以其他方式进行保护。执行该保护以防止对形成计算机对象的虚拟硬盘136'的内容的篡改或未经授权的访问。

[0049] 再次参照图1，在至少一些实施例中，应用程序132可以是自给自足的并且可采取单一可执行的二进制文件或特殊类型的程序模块134(也被称为容器)的形式。容器可包括应用程序运行需要的一切，包括：可执行指令、应用程序运行时库、系统工具和库。类似于其他对象，出于安全的目的，在创建时使用可用于操作系统以防止篡改的经典数据保护方法来对其进行加密保护或以其他方式进行保护。

[0050] 如上所讨论的，在至少一些实施例中，用于地理验证的方法和系统包括使地理区域与各种计算机对象类型相关联。在至少一些实施例中，计算机对象可包括计算机文件、包含计算机文件的储存设备、从储存设备接收和发出计算机文件的接口、具体化计算机文件的数据库、储存设备和接口、以及各种其他组合。

[0051] 提供计算机对象被准许操作的准许位置的地理区域可通过收集地理位置数据点来识别，并且可选地，可通过附加的几何尺寸来识别。参见图3A、图3B、图3C、图3D和图3E，这些数据点的分组可包含以下中的任何一个：a)一对经度和纬度坐标322加半径323(图3A)，b)一对经度和纬度坐标322加辐射长度324和辐射宽度325(图3B)，或c)两对或更多对经度和纬度坐标(图3C、图3D、图3E)的集合。这些地理位置数据点的每个分组都框出周边，诸如假想的圆330、矩形332、334或多边形336。每个周边表示计算机对象可被部署、启动和操作

的地球表面上的实际连续和有界的地理区域。

[0052] 地理区域可以与围绕数据中心建筑物320(图3A)内的服务器机架覆盖区的直接楼层空间一样小,或者与其边界被表示为如图3E中所示的一系列地理坐标对342、344、346和348的管辖范围340一样大。此外,矩形地理区域可根据地理坐标来计算或导出。矩形地理区域周边的四个角可根据如图3B中所示的给定的地理坐标322以及辐射的长度324和宽度325而被计算出,或者如图3C中所示,根据两个给定的地理坐标322'和322"导出,以获取另外两个326和327。

[0053] 这些地理位置数据点的每个分组可被识别为计算机设备(例如,服务器等)、与计算机设备相关联的操作系统或计算机对象的地理位置资源声明(即资源属性)。每个计算机对象可以被分配多个地理位置声明。计算机对象的地理位置声明在本文中可以被称为地理对象资源声明。地理对象资源声明可以被记录并与计算机对象的唯一标识符相关联,该计算机对象的唯一标识符在本文中以下被称为对象ID(以下更详细地讨论)。该信息可以保存在域名服务目录185或类似类型的仓库中。在至少一些实施例中,对域名服务目录185的访问限于一个或多个被授权访问的等级。类似于给计算机对象分配一个或多个地理位置声明,计算机操作系统130或管理程序132'也可被分配至少一个被授权的操作地理区域。每个地理区域建立操作系统130在其内被允许访问类似受限的资源的面积。该地理范围包括相关联的操作系统或管理程序实例(而不是计算机本身)的一个或多个地理位置声明,并且在本文中可被称为地理资源设备声明。出于该目的,可分配多个地理资源设备声明。如同利用地理资源对象声明,地理资源设备声明和相关联的细节在不同的或相同的域名服务目录185内被保护,并且在没有所需授权的情况下不能由计算机的管理员或操作者访问。

[0054] 在创建一个或多个地理位置声明之后,授权用户可添加、减去或替换地理位置声明中的地理位置坐标,并提取它们以用于信息技术(IT)管理目的。这样的改变可使用人机接口142来完成,人机接口142通常将包括诸如监视器138、键盘139和鼠标140的设备。区别于地理位置声明,被分配的地理位置指示符(例如,文件属性),其可通知用户预期准许的地理位置并且可识别计算机对象是否带有标签以用于验证(带有地理标签),可与计算机对象相关联,因此在所有用户抽检时是可见的。分配的地理位置指示符的添加不会影响地理位置声明,因为它们仍然受到保护,因此在抽检的情况下是不可访问的。访问授权和控制的这种分离可例如通过域控制器180结合域名服务目录185或另一个数据仓库来实现。

[0055] 用一个或多个地理对象资源声明(带有地理标签)标记计算机对象可用许多类型的基于计算机的操作系统来执行,并且可在诸如数据中心或部署中心的各个位置进行。示例性标记过程参照图4中的流程图400进行描述。为了开始该过程,在步骤402,形成地理资源对象声明。更具体地,一个或多个位置的地理坐标,诸如被授权操作计算机对象的数据中心的地理坐标,被获取和储存。在至少一些实施例中,地理坐标被储存为与域控制器180相关联的域名服务目录185中的资源属性。在步骤404,为计算机对象生成唯一对象ID,该唯一对象ID将用在验证过程中。在至少一些实施例中,唯一对象ID是全局唯一ID(GUID),尽管可使用其他类型的标识符。

[0056] 在步骤406,请求基于地理位置的授权进行访问或操作的计算机对象由操作系统创建,并且唯一对象ID被嵌入或以其他方式与对象相关联。该任务能够以许多不同的方式来执行。例如,当计算机对象是基于文件的,诸如用于操作虚拟机(例如,虚拟机101')的虚

拟硬盘(例如,虚拟硬盘136')时,虚拟硬盘映像文件可被创建具有一个以上的部分(例如,分区、节段(section)等)。在至少一些实施例中,节段可包括授权数据部分和程序数据部分。程序数据部分可包含例如用于操作虚拟机101'和唯一对象ID的指令。授权数据部分可包含例如关于虚拟机101'的唯一对象ID和地理位置验证要求信息,诸如地理资源对象声明。旨在授权数据部分将独立于访问程序数据部分并且仅由被授权的实体可访问。这将允许在不访问对象的程序内容的情况下进行授权。另外,如上所述,提供可从第二安全部分分开地察看到的第一安全部分的上述过程可用于各种类型的计算机对象。

[0057] 在步骤408,操作系统将计算机对象的唯一对象ID储存在域名服务目录185中。在步骤410,域控制器180将地理资源对象声明绑定到域名服务目录185中的唯一对象ID作为资源属性。在步骤412,操作系统将附加的资源属性和数据保护方法应用于计算机对象以用于被授权的访问目的,并且保存计算机对象供后续部署。如上所述,一个添加的属性可以是分配的地理位置指示符。一旦地理位置带有了地理标签,计算机对象就可被分布到数据中心或其他地点以供后续访问,并且相对于尝试访问带有地理标签的对象的操作系统可被识别为带有地理标签的。

[0058] 现在参照图5、图6和图7,为了将带有地理标签的计算机对象投入到数据中心或其他服务点处的服务中,并且在一些情况下,继续使用带有地理标签的对象,提出访问请求并执行验证。该过程以下参照流程图500、600和700详细地进行讨论。更具体地,讨论了包括示例性虚拟机(图5)、示例性容器(图6)和示例性数据仓库(图7)的各种类型的对象的部署/访问。由于程序性步骤,在至少一些实施例中,对于许多类型的计算机对象是类似的,包括这三种类型的计算机对象,所以步骤将针对三个流程图同时得以解决。

[0059] 从步骤502、602、702开始,当操作系统130开始启动或恢复事件时,并且在其被连续操作期间的时间,操作系统130从其有源GPS接口160(具有连接的GPS天线162)获取其当前正被操作的实际和可证明的地理位置,其在本文中也被称为设备地理位置声明。如果被激活的计算机是虚拟机(例如,101'),则其操作系统130'从其管理程序132'获取地理位置信息。在步骤504、604、704中,操作系统130当前正在操作的实际地理位置对系统管理员和操作者是可用的,并被转发到适当的域控制器180以更新域名服务目录185。发生在计算机对象被部署或以其他方式被验证以供操作之前、期间和之后的该过程可以是连续的。

[0060] 在步骤506、606和706,操作系统130接收开启计算机对象(虚拟硬盘136'、容器134或可执行的二进制文件132)的部署或以其他方式授予对其的访问的请求。如果计算机对象被识别为带有地理标签,诸如具有地理位置指示符,则过程前进到步骤508、608、708,其中操作系统130将计算机对象的唯一对象ID发送到域控制器180以用于评估。在至少一些实施例中,操作系统130还为操作系统130发送系统ID。如果计算机对象未被操作系统识别为带有地理标签的,则该过程前进到步骤514、614、714,其中授予对计算机对象的部署和/或访问。在步骤510、610、710,使用所发送的对象ID,域控制器180搜索域名服务目录185,以识别计算机对象的地理对象资源声明。一旦被识别,域控制器180就启动绑定到在被记录于域名服务目录185中的对象ID的地理对象资源声明与也被储存在域名服务目录185中的操作系统130的报告的设备地理位置声明(地理位置坐标)之间的比较。在至少一些实施例中,比较包括域控制器180使用所发送的设备地理位置声明和地理对象资源声明来解决点在多边形中(PIP)的问题。PIP几何计算解决了平面中的给定点是否位于多边形的边界的内部、外部

或其上的问题。经典几何计算和方法也可用于确定与计算机对象相关联的给定地理位置是否落在地球表面上允许界定的连续地理区域中或外部。这样的计算响应于访问授权请求由域控制器180执行。更具体地，如果在步骤512、612、712，设备地理位置声明被宣告为落在边界(即，区域)内，该边界表示受保护的地理对象资源声明被分配给计算机对象，则域控制器将宣告评估成功，并返回确认地理位置验证的通过结果。如果设备地理位置声明被宣告为落在边界(即，区域)的外部，该边界表示被分配给计算机对象的受保护的地理对象资源声明，则过程移动到步骤516、616、716，并且域控制器将宣告评估失败。在响应中，操作系统130相应地起作用。也就是说，过程移动到步骤514、614、714，其中操作系统130及其管理程序132'被授权继续进行虚拟机101'的部署，运行客户操作系统130'、可执行的二进制文件132或操作容器134。

[0061] 有利地，该方法还可通过防止虚拟硬盘136'在组织内被复制并在特定的物理数据中心地点之外以及以未经授权的方式被操作来提供虚拟硬盘136'的附加保护层。该方法还允许在供流氓员工企图部署和启动虚拟机时被发现并进行缓解，流氓员工利用对用于备用的虚拟硬盘的储存的合法的授权的访问和其他数据管理操作但未被授权使用其内容来部署和启动虚拟机。除了能够证明系统的位置和管辖范围合同义务之外，根据本发明原理的该方法还允许供应商确定最佳有效的数据中心地点，以满足其客户需求，同时操作他们的工作量在诸如电力和通讯的环境和基础设施制约范围内。此外，在至少一些实施例中，该方法可用于查明用户是否在地理区域内操作设备或计算机，并且因此，通过预定策略准许创建数据仓库170并将其附加到操作系统，或访问在相同或另一地理区域中被策划的现有数据仓库。

[0062] 如上所述，虽然本发明的示例性实施例已经结合各种计算设备和系统架构进行了描述，但潜在的概念可应用于任何计算设备或系统，其中期望实现在特定地理的数据中心中的虚拟机的受控的部署和操作。

[0063] 因此，本发明的方法和系统可应用于各种应用和系统。虽然示例性硬件接口、名称和示例在本文中被选择为代表各种选择，但这些硬件接口、名称和示例不旨在限制。本领域中的普通技术人员将认识到，存在多种方式来获得实现由本发明实现的相同、类似或等效的系统和方法的地理位置数据。

[0064] 本文中所描述的各种技术可结合硬件或软件或在适当的情况下结合二者的组合来实现。因此，本发明的方法和系统或其某些方面或部分可采用在诸如CD-ROM、闪存驱动器、硬盘驱动器或任何其他机器可读储存介质的有形介质中具体化的程序代码(即，指令)的形式，其中，当程序代码被加载到诸如计算机的机器中并由其执行时，机器成为用于实践本发明的系统。在程序代码在可编程计算机上执行的情况下，计算设备通常将包括处理器、可由处理器(包括易失性和非易失性存储器和/或储存元件)读取的储存介质、至少一个输入设备、以及至少一个输出设备。优选地，可利用本发明的信号处理服务(例如，通过使用数据处理API等)的一个或多个程序以高级程序性或面向对象的编程语言来实现，以与计算机进行通信。然而，如有必要，程序能够以汇编或机器语言来实现。在任何情况下，语言可以是编译或解释的语言，并与硬件实现相结合。

[0065] 本发明的方法和系统还可经由以在某传输介质上传输的程序代码形式的通信来实践，诸如，在电线或电缆上，通过光纤，或经由任何其他形式的传输，其中，当程序代码被

接收并被加载到机器(诸如EPROM、门阵列、可编程逻辑器件(PLD)、客户端计算机、录像机等)或具有如示例性实施例中所描述的信号处理能力的接收机中并由其执行时，成为用于实践本发明的系统。当程序代码在通用处理器上实现时，该程序代码与处理器结合，以提供操作以调用本发明的功能的唯一系统。另外，与本发明结合使用的任何储存技术可以总是硬件和软件的组合。

[0066] 尽管已经结合各个附图的优选实施例对本发明进行了描述，但应当理解，可使用其他类似的实施例，或者可对所描述的实施例进行修改和添加，以实现本发明的相同功能而不与其背离。此外，应强调的是，考虑各种计算机平台，包括手持设备操作系统和其他应用特定的操作系统，特别是当无线网络设备的数量继续增加时。因此，本发明不应限于任何单一实施例，而是在宽度和范围方面应根据所附权利要求进行解释。

[0067] 虽然本发明已经参照优选实施例进行了描述，但应当理解，本发明不旨在限于以上所阐述的具体实施例。因此，认识到的是，本领域中的那些技术人员将会理解，在不脱离本发明的精神或目的的情况下，可做出某些替换、改变、修改和省略。因此，前面的描述仅仅是示例性的，本发明被视为包括本发明主题的所有合理的等同物，并且不应限制以下权利要求中所阐述的本发明的范围。此外，本文中参照操作方法所描述的步骤并不旨在限制性的，并且可包括变化，诸如附加步骤、移除的步骤和重新排序的步骤。

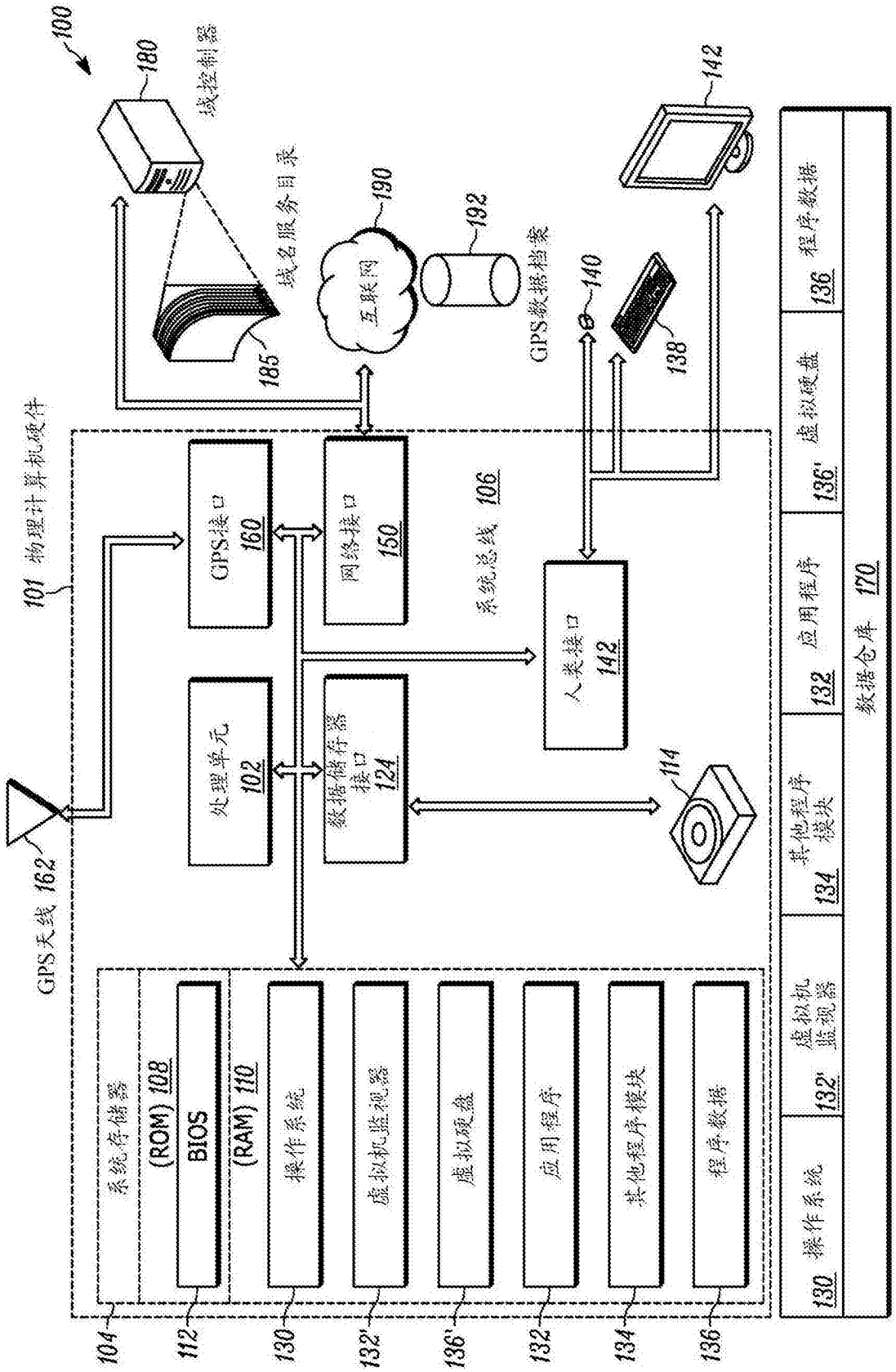

100
180
域控制器
域名服务目录
185
互联网
190
192
GPS数据档案
142
140
138
101 物理计算机硬件
GPS接口 160
网络接口 150
系统总线 106
人类接口 142
处理单元 102
数据储存器接口 124
114
GPS天线 162
系统存储器
(ROM) 108
BIOS
(RAM) 110
操作系统
虚拟机监视器
虚拟硬盘
应用程序
其他程序模块
程序数据
130 操作系统
132' 虚拟机监视器
134 其他程序模块
132 应用程序
136' 虚拟硬盘
136 程序数据
数据仓库 170
104
112
130
132'
136'
132
134
136

图1

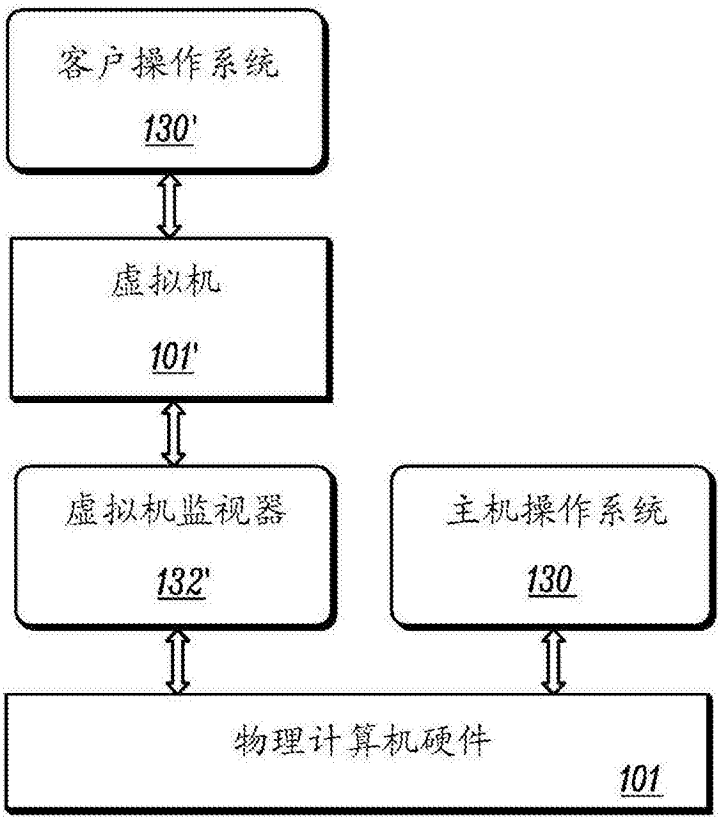

客户操作系统
130'
虚拟机
101'
虚拟机监视器
132'
主机操作系统
130
物理计算机硬件
101

图2

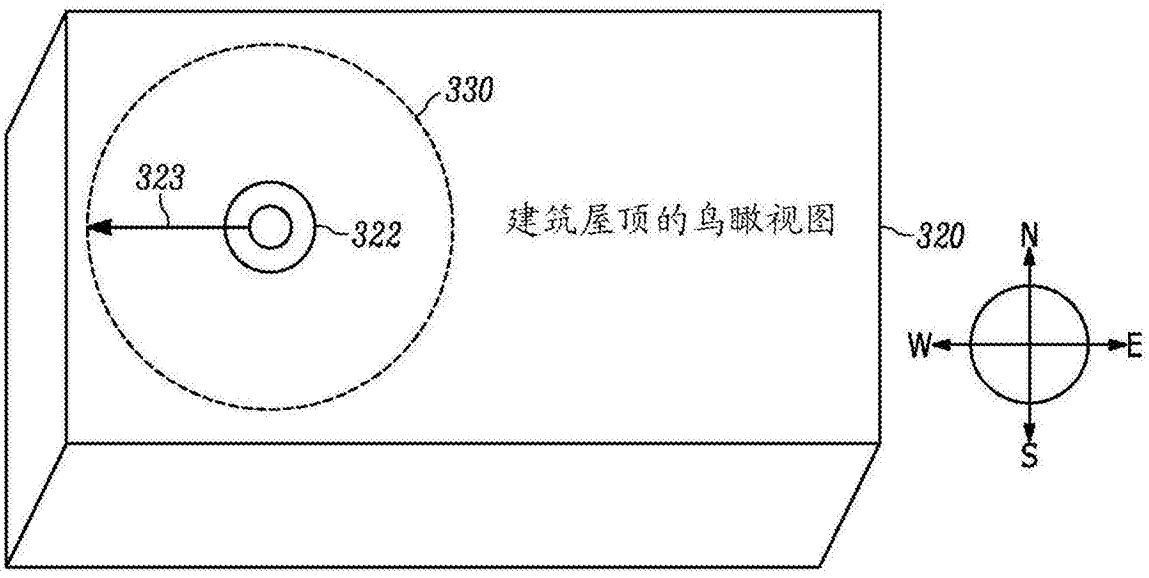

330
323
322
建筑屋顶的鸟瞰视图
320
N
W
E
S

图3A

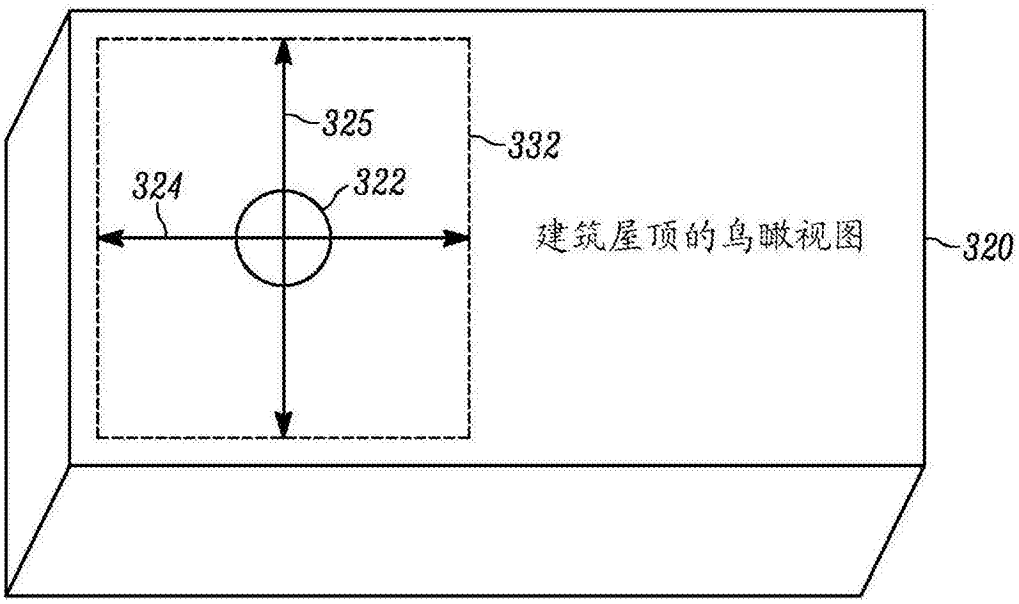

325
332
324
322
建筑屋顶的鸟瞰视图
320

图3B

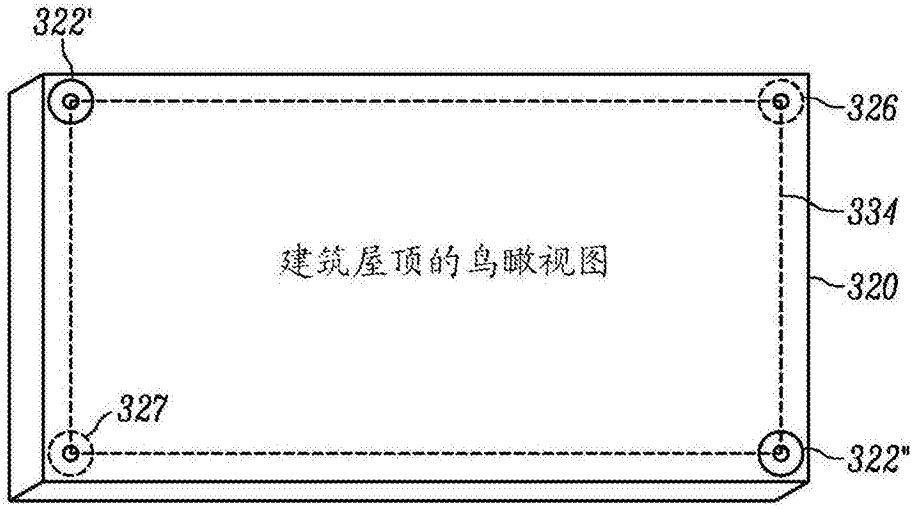

322'
326
334
建筑屋顶的鸟瞰视图
320
327
322"

图3C

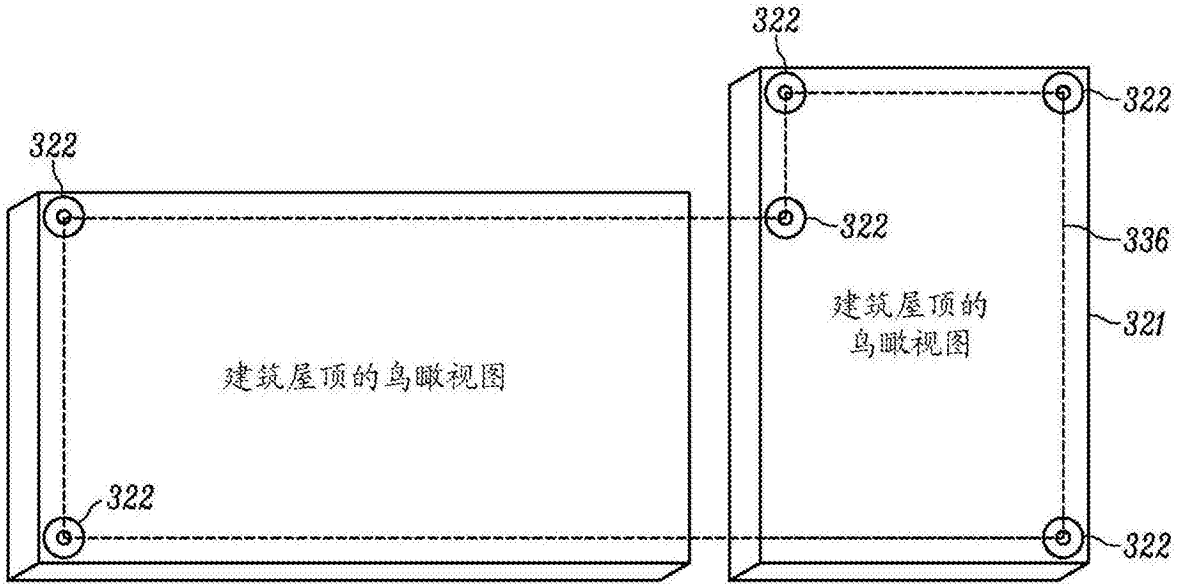

322
322
322
322
322
322
322
336
321
建筑屋顶的鸟瞰视图
建筑屋顶的鸟瞰视图

图3D

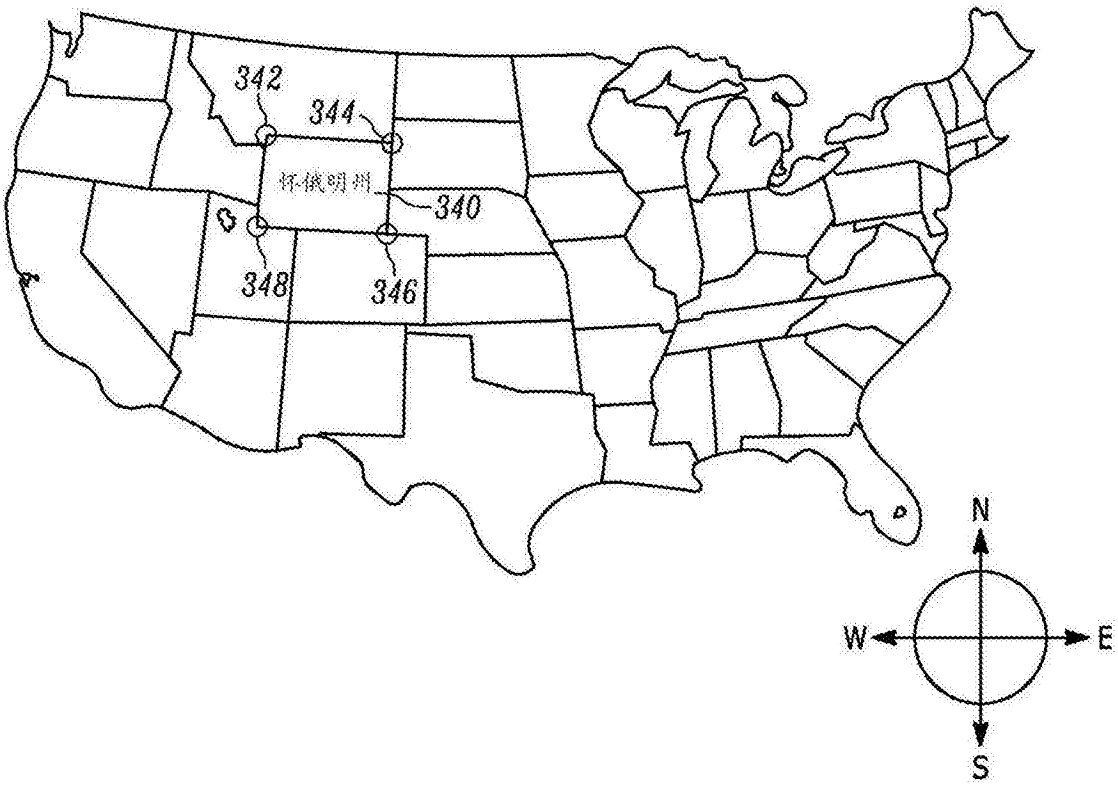

342
344
340
348
346
N
W
E
S

图3E

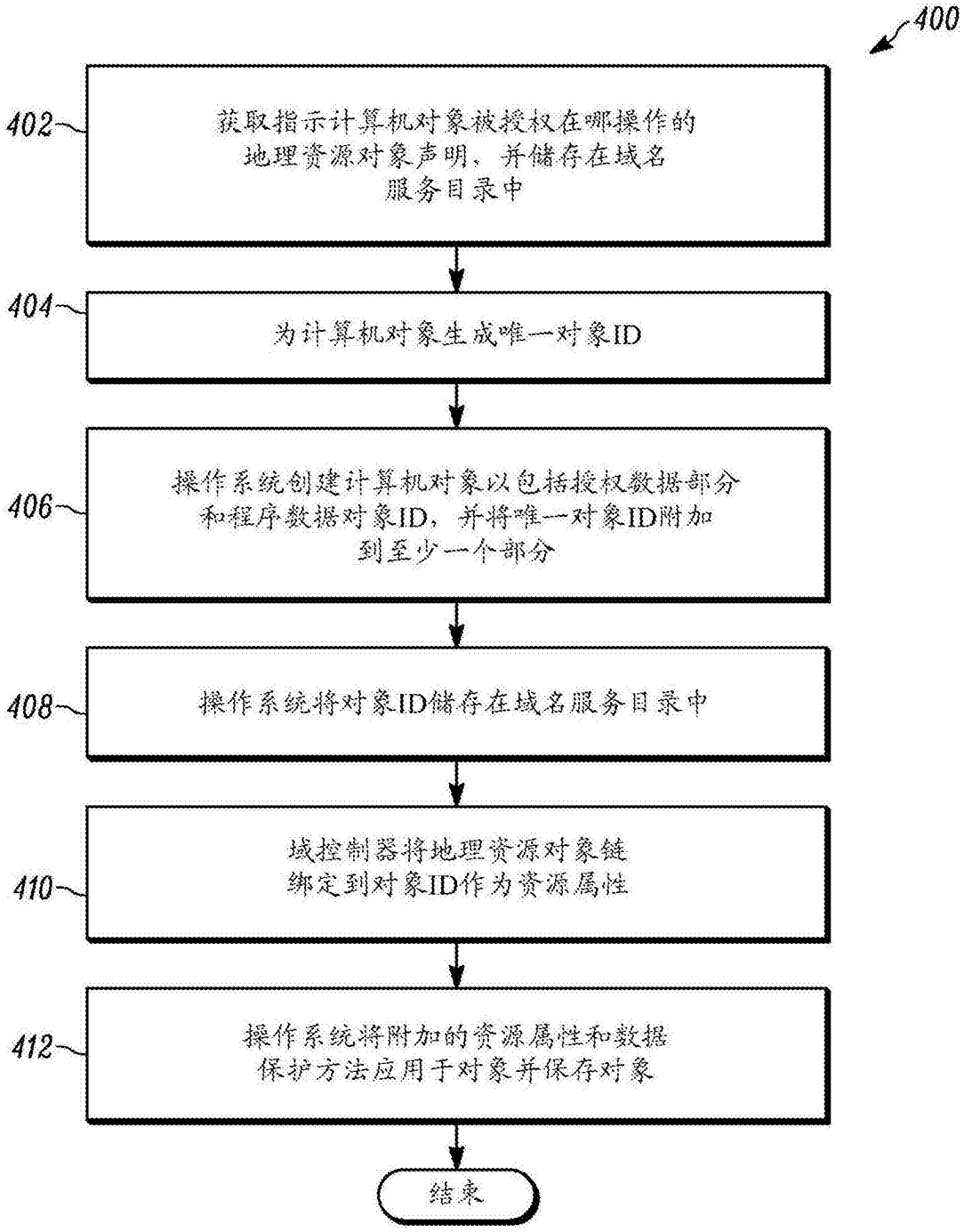

400
402
获取指示计算机对象被授权在哪操作的地理资源对象声明，并储存在域名服务目录中
404
为计算机对象生成唯一对象ID
406
操作系统创建计算机对象以包括授权数据部分和程序数据对象ID，并将唯一对象ID附加到至少一个部分
408
操作系统将对象ID储存在域名服务目录中
410
域控制器将地理资源对象链绑定到对象ID作为资源属性
412
操作系统将附加的资源属性和数据保护方法应用于对象并保存对象
结束

图4

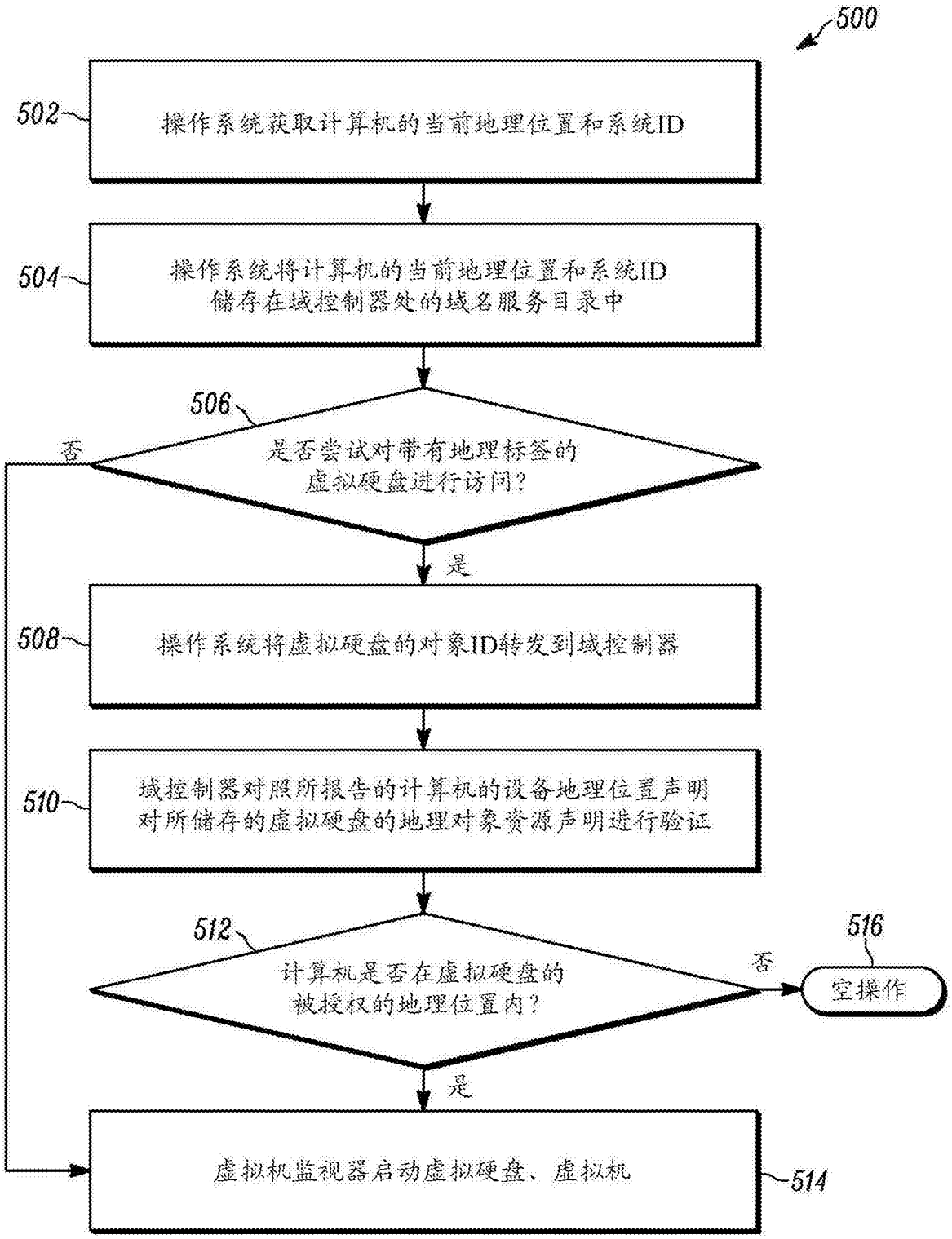

500
502
操作系统获取计算机的当前地理位置和系统ID
504
操作系统将计算机的当前地理位置和系统ID
储存在域控制器处的域名服务目录中
506
是否尝试对带有地理标签的
虚拟硬盘进行访问？
否
是
508
操作系统将虚拟硬盘的对象ID转发到域控制器
510
域控制器对照所报告的计算机的设备地理位置声明
对所储存的虚拟硬盘的地理对象资源声明进行验证
512
计算机是否在虚拟硬盘的
被授权的地理位置内？
否
516
空操作
是
虚拟机监视器启动虚拟硬盘、虚拟机
514

图5

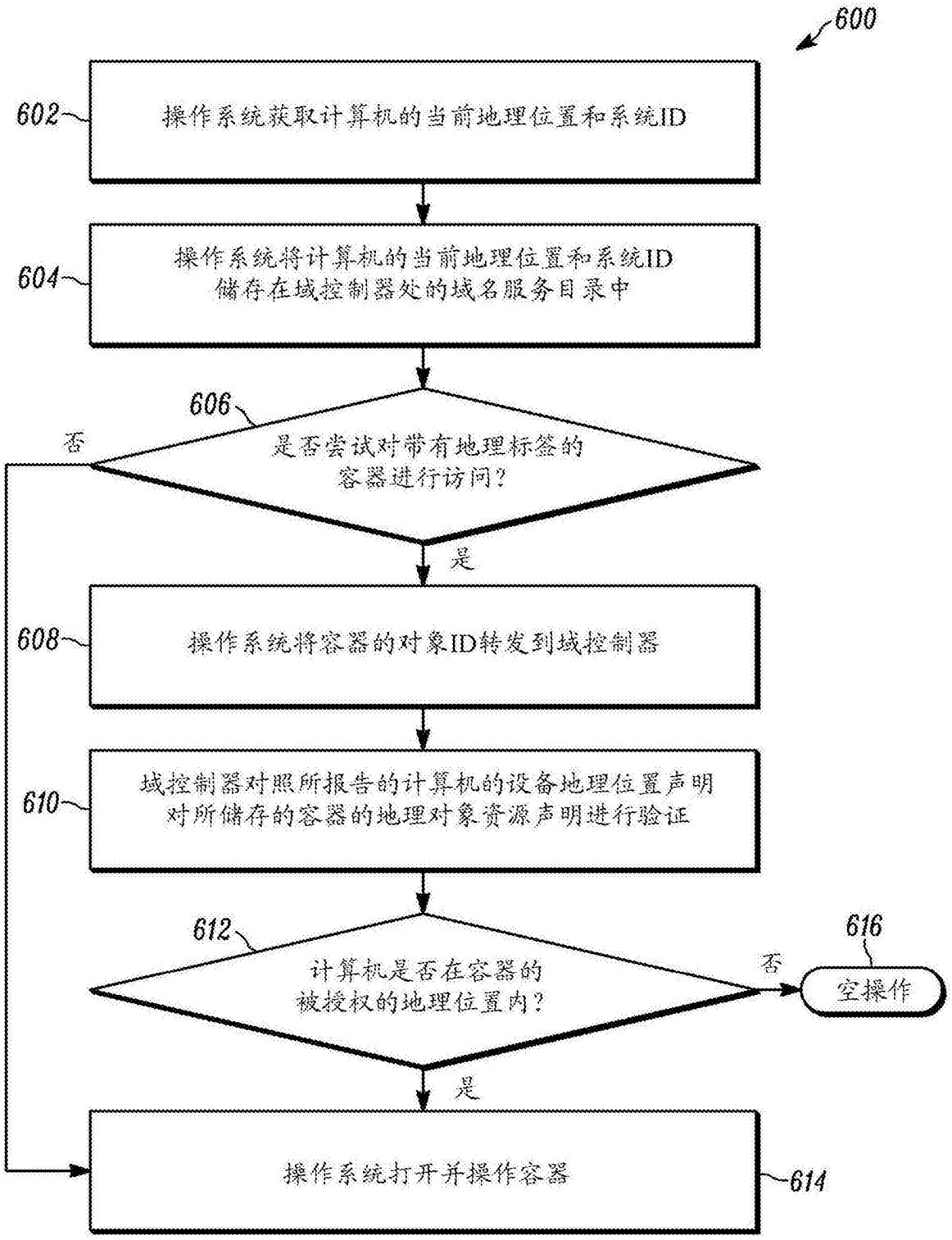

600
602
操作系统获取计算机的当前地理位置和系统ID
604
操作系统将计算机的当前地理位置和系统ID
储存在域控制器处的域名服务目录中
606
是否尝试对带有地理标签的
容器进行访问?
否
是
608
操作系统将容器的对象ID转发到域控制器
610
域控制器对照所报告的计算机的设备地理位置声明
对所储存的容器的地理对象资源声明进行验证
612
计算机是否在容器的
被授权的地理位置内?
否
616
空操作
是
操作系统打开并操作容器
614

图6

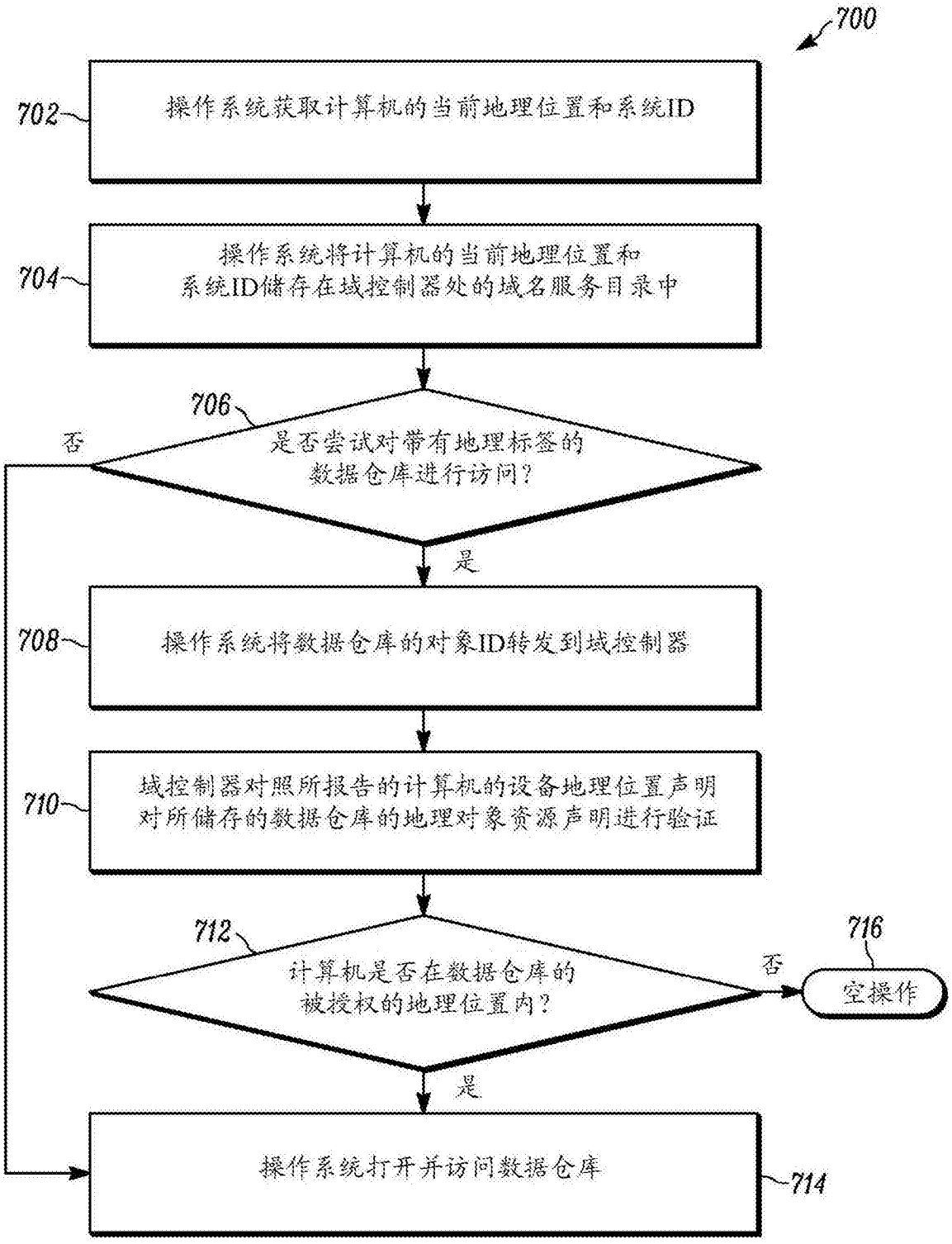

700
702
操作系统获取计算机的当前地理位置和系统ID
704
操作系统将计算机的当前地理位置和
系统ID储存在域控制器处的域名服务目录中
706
是否尝试对带有地理标签的
数据仓库进行访问?
否
是
708
操作系统将数据仓库的对象ID转发到域控制器
710
域控制器对照所报告的计算机的设备地理位置声明
对所储存的数据仓库的地理对象资源声明进行验证
712
计算机是否在数据仓库的
被授权的地理位置内?
否
716
空操作
是
操作系统打开并访问数据仓库
714

图7