



- (51) **International Patent Classification:**
H04L 12/26 (2006.01) *H04L 12/24* (2006.01)
- (21) **International Application Number:**
PCT/CA2014/051267
- (22) **International Filing Date:**
23 December 2014 (23.12.2014)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
61/921,383 27 December 2013 (27.12.2013) US
- (71) **Applicant:** **METAFOR SOFTWARE INC.** [CA/CA];
124 Pender Street East, Suite 210, Vancouver, British
Columbia V6A 1T3 (CA).
- (72) **Inventor:** **BOUBEZ, Toufic**; c/o Metafor Software Inc.,
124 Pender Street East, Suite 210, Vancouver, British
Columbia V6A 1T3 (CA).
- (74) **Agents:** **DE KLEINE, Geoffrey** et al.; Borden Ladner
Gervais LLP, 100 Queen Street, Suite 1300, Ottawa,
Ontario K1P 1J9 (CA).

(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.

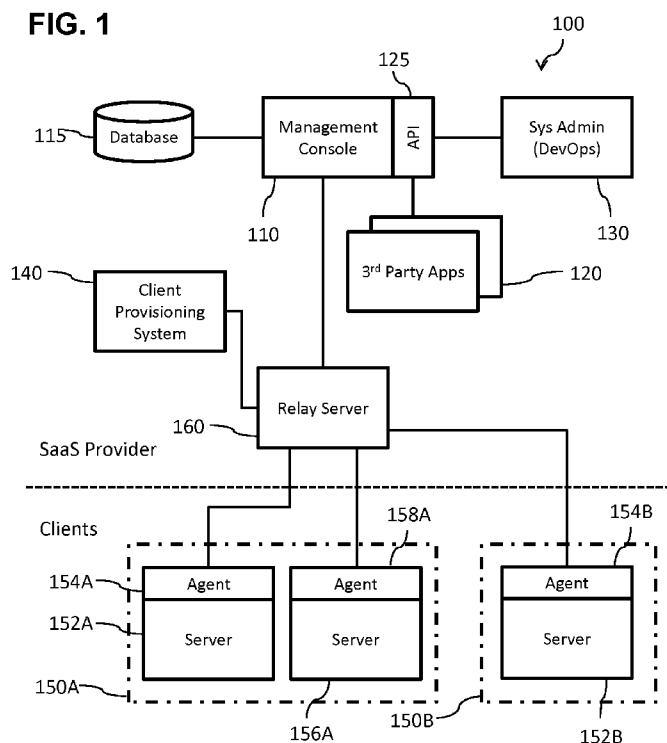
(84) **Designated States** (unless otherwise indicated, for every kind of regional protection available): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, ST, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, 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:** SYSTEM AND METHOD FOR ANOMALY DETECTION IN INFORMATION TECHNOLOGY OPERATIONS

FIG. 1



(57) **Abstract:** An anomaly detection system is able to detect spatial and temporal environment anomalies and spatial and temporal behavior anomalies, and monitor servers for anomalous characteristics of the environment and behavior. If metrics and/or characteristics associated with a given server are beyond a certain threshold, an alert is generated. Among other options, the alert can take the form of a heat map or a cluster cohesiveness report.

- 1 -

SYSTEM AND METHOD FOR ANOMALY DETECTION IN INFORMATION TECHNOLOGY OPERATIONS

Cross-Reference to Related Applications

[0001] This application claims priority benefits from United States Provisional Patent Application serial no. 61/921,383, filed December 27, 2013 and entitled “System and Method for Anomaly Detection in Information Technology Operations”. The ‘383 application is hereby incorporated by reference.

Field of the Invention

[0002] The present invention relates to systems and/or methods for anomaly detection in Information Technology (IT) operations, and more specifically, to systems and/or methods for detection of spatial and temporal manifestations of environment and behavior anomalies in servers, data sets and software applications.

Background of the Invention

[0003] A data center is a facility used to house computing hardware that can include servers, storage systems and telecommunications equipment. Typically a data center comprises multiple servers. Servers can be physical, virtual, or cloud-based machines. A data center, or more generally a computing system, can comprise software such as application software, system software and the like. The software can run on the servers.

[0004] A data center can comprise large clusters of related servers. As data center infrastructure grows, it becomes important to monitor the configuration and health of the servers automatically, and to alert an operator when anomalies occur. The configuration of the data center can include the configuration of hardware and software. The software configuration can include, for example, the configuration of application software and system software running on the servers.

[0005] A data center is a dynamic environment and anomalies can occur frequently in data center operations. Anomalies can be associated with software and/or hardware. It is common for software to be undergoing continuous deployment. The software environment is changing frequently, for example when software is being updated to a new version. Hardware changes are also frequent – with machines being spun up and down, especially in virtual or cloud environments.

[0006] The environment can be chaotic and overwhelming for manual detection methods. It is not practical to sit and watch the operations all the time

- 3 -

for anomalies. There is a need for methods and systems for automated anomaly detection in data center operations.

[0007] It can be hard to detect anomalies and they can cause significant disruption in computer systems and networks. Considerable effort can be spent trying to find anomalies. It is advantageous to have efficient automated ways to find anomalies in a timely fashion.

[0008] Examples of the detection of anomalies by monitoring, analysis or data-mining of system event logs have been discussed. The system and method described herein is related to anomaly detection through static and dynamic analysis of files, packages (such as installed software applications), and metadata.

[0009] Earlier work also discloses threshold-based approaches, for example alerting an operator when a disk is 90% full. The system and method described herein can identify and measure trends, and can anticipate problems before thresholds are triggered.

Summary of the Invention

[0010] A method for detecting a spatial environment anomaly in a system with a plurality of servers includes: monitoring a configuration of each of the plurality of servers and generating an alert associated with a configuration difference selected from the group consisting of a first server of the plurality of servers having a substantively different configuration than other servers of the plurality of servers, and a first server of the plurality of servers having a

- 4 -

different configuration than other servers of the plurality of servers with respect to configuration element designated by a user.

[0011] In some embodiments the configuration is a software configuration. The software configuration can be, among other things, one or more installed software applications or one or more configuration files. In some embodiments at least one of the configuration files specifies the configuration of an installed software application.

[0012] A method for detecting a temporal environment anomaly in a server includes: monitoring a configuration of the server at a first time and a second time; and generating an alert associated with a change in the configuration between the first time and second time wherein the change is more than a predetermined threshold between the first time and the second time and/or related to a configuration element designated by a user.

[0013] In some embodiments the server is one of a plurality of servers in a system.

[0014] In some embodiments the configuration is a software configuration. The software configuration can be, among other things, one or more installed software applications or one or more configuration files.

[0015] A method for detecting a spatial behavior anomaly in a system with a plurality of servers includes: receiving a set of metrics from each of the plurality of servers at one or more times during a time period; segmenting the time period into one or more time windows; calculating one or more statistics for at least one metric of the set of metrics for each of the plurality of servers for each of the

- 5 -

time windows (in some embodiments the statistics calculated are the mean, standard deviation, kurtosis and/or entropy).

[0016] The method further includes concatenating the metrics and the statistics into a corresponding n-dimensional feature vector for each of the plurality of servers (in some embodiments the n-dimensional feature vector is a projection of a higher dimensional space onto n dimensions); calculating one or more spectral characteristics of the metrics; concatenating the spectral characteristics into the n-dimensional feature vector; computing a center of mass in n-dimensional space of the feature vectors; computing a distance measurement from the feature vector of a first server of the plurality of servers to the center of mass; determining a degree of anomaly of the first server based on the distance measurement; monitoring the degree of anomaly of the first server: and generating an alert when the degree of anomaly of the first server exceeds a predetermined threshold.

[0017] In some embodiments the metrics include one or more asynchronous time-stamped log entries, memory usage, CPU usage, traffic and load, and/or synchronous time series metrics. In other or the same embodiments, the metrics are either real-time metrics or replays of metrics collected prior to the start of the time period.

[0018] In other or the same embodiments, the method further includes, after receiving a set of metrics from each of the plurality of servers at one or more times during a time period, the step of transforming the one or more asynchronous time-stamped log entries into one or more synchronous time series metrics.

- 6 -

[0019] In some embodiments the spectral characteristics are wavelet coefficients or Fourier coefficients. In other of the same embodiments, the distance measurement can be a Euclidean distance, L1 (Manhattan) distance, or Canberra distance.

[0020] In some embodiments the projection is derived from a Principal Components Analysis. In other or the same embodiments the set of metrics is received by an agent installed on each of the plurality of servers.

[0021] A method of detecting a temporal behavior anomaly in a server, includes receiving a set of metrics from the server at one or more times during a time period; segmenting the time period into one or more time windows; calculating one or more statistics for at least one metric of the set of metrics for the server for each of the time windows; concatenating the metrics and the statistics into a corresponding n-dimensional feature vector for each of the time windows; calculating one or more spectral characteristics of the metrics, concatenating the spectral characteristics into the n-dimensional feature vector; computing a center of mass in n-dimensional space of the feature vectors; computing a distance measurement from a first feature vector of the feature vectors to the center of mass, the first feature vector corresponding to a first time window; determining a degree of anomaly of the server based on the distance measurement; monitoring the degree of anomaly of the server; and generating an alert when the degree of anomaly of the server exceeds a predetermined threshold.

[0022] In some embodiments the metrics includes one or more asynchronous time-stamped log entries and/or synchronous time series metrics. In other, or the

- 7 -

same embodiments, the set of metrics are real-time metrics or replays of metrics collected prior to the start of the time period. In some embodiments, the one or more asynchronous time-stamped log entries can be converted into one or more synchronous time series metrics.

[0023] In some embodiments the server is one of a plurality of servers in a system.

[0024] A method for detecting anomalies in a system with one or more servers, the method includes: receiving at least one metric corresponding to at least one server of the one or more servers; computing statistics associated with the at least one metric; applying the at least one metric and the statistics to detect at least one anomaly in the system, the at least one anomaly selected from the group consisting of spatial environment anomalies, temporal environment anomalies, spatial behavior anomalies and temporal behavior anomalies; determining the degree of the anomaly, and monitoring the persistence of the anomaly; and generating an alert to an operator of the system. In some embodiments the alert is generated when the degree exceeds a first threshold and the persistence exceeds a second threshold.

[0025] In one embodiment the alert is generated based on the detection of at least one environment anomaly and at least one behavior anomaly.

[0026] A method for detecting anomalies in a system with one or more software applications, includes: receiving at least one metric corresponding to at least one software application of the one or more software applications; computing statistics associated with the at least one metric; applying the at least one metric and the statistics to detect at least one anomaly in the system, the at

- 8 -

least one anomaly selected from the group consisting of spatial environment anomalies, temporal environment anomalies, spatial behavior anomalies and temporal behavior anomalies; determining the degree of the anomaly; monitoring the persistence of the anomaly; and generating an alert to an operator of the system. In some embodiments the alert is generated when the degree exceeds a first threshold and the persistence exceeds a second threshold.

[0027] In some embodiments, the alert is generated based on the detection of at least one environment anomaly and at least one behavior anomaly.

[0028] A method for detecting anomalies in a system with one or more sensors, includes: receiving at least one metric corresponding to at least one sensor of the one or more sensors; computing statistics associated with the at least one metric; applying the at least one metric and the statistics to detect at least one anomaly in the system, the at least one anomaly selected from the group consisting of spatial environment anomalies, temporal environment anomalies, spatial behavior anomalies and temporal behavior anomalies; determining the degree of the anomaly; monitoring the persistence of the anomaly and generating an alert to an operator of the system.

[0029] In some embodiments then alert is generated when the degree exceeds a first threshold and the persistence exceeds a second threshold.

[0030] In some embodiments the alert is generated based on the detection of at least one environment anomaly and at least one behavior anomaly.

[0031] In some embodiments the metric includes sensor data from the one or more sensors.

[0032] The above methods can be implemented by various devices.

Overview of Anomalies

[0033] An anomaly can be an environment anomaly or a behavior anomaly. An environment anomaly can also be known as a static anomaly or a configuration anomaly. A behavior anomaly can also be known as a dynamic anomaly.

[0034] An environment anomaly can be a spatial environment anomaly or a temporal environment anomaly. Similarly, a behavior anomaly can be a spatial behavior anomaly or a temporal behavior anomaly.

[0035] The system and method described herein can provide a number of benefits including, but not limited to, the following functions:

- (a) detection of environment anomalies and the monitoring of servers for anomalous characteristics of the environment:
 - (i) detection of spatial environment anomalies;
 - (ii) detection of temporal environment anomalies;
- (b) detection of behavior anomalies and the monitoring of servers for anomalous behavior:
 - (i) detection of spatial behavior anomalies;
 - (ii) detection of temporal behavior anomalies;
- (c) hybrid detection (environment and behavior); and
- (d) generation of alerts to the operator.

- 10 -

[0036] The system and method described herein can include the following functions:

1. Cohesive analysis in which the detection of spatial behavior anomalies can comprise superimposing one time window from multiple metrics to determine if one of those metrics is anomalous; and
2. Cohesive trending in which cohesive analysis is used in the detection of temporal behavior anomalies by superimposing multiple time windows of one metric in order to determine whether the current time window is anomalous.

Environment Anomaly Detection

[0037] Spatial environment anomalies are examples of environment anomalies and are manifest as differences or anomalies between a set of servers at a particular point in time. Some differences between servers are not regarded as anomalous, for example IP addresses, and these differences can be “whitelisted” and ignored.

[0038] Temporal environment anomalies are other examples of environment anomalies and are manifest as differences or anomalies in the environment of one or more servers between two different points in time. Detection of temporal anomalies can be achieved by taking “snapshots” of the files, packages and metadata on one or more servers, and looking for changes on each server.

Behavior Anomaly Detection

[0039] Spatial behavior anomalies are examples of behavior anomalies and are manifest as differences or anomalies between a set of servers or a set of metrics at a particular point in time.

[0040] Temporal behavior anomalies are other examples of behavior anomalies and are manifest as differences or anomalies in the behavior of one or more servers or one or more metrics between two different points in time.

[0041] Behavior anomaly detection can be achieved by collecting metrics on one or more servers. Suitable metrics can include infrastructure and operational data such as:

- (a) memory usage;
- (b) central processing unit (CPU) usage;
- (c) disk input/output (I/O);
- (d) traffic, and
- (e) load.

[0042] In this application, “traffic” means the amount of data sent and received per second on a server's Network Interface Controller (NIC).

[0043] In this application, “load” means the number of running processes on a server. For the purposes of deriving a load metric, this can be computed, for example, as the exponentially-weighted moving average of the load number over one or more fixed time intervals such as the past minute, 5 minutes, and 15 minutes. The load metric is computed by counting running processes where “running” means running or waiting in the process queue. Therefore, a load metric of 1.0 means the server is fully loaded. A load metric of 0.75, for

- 12 -

example, indicates there is spare capacity. A load metric of 1.25, for example, indicates the server is overloaded.

[0044] Such metrics can also include many types of application-specific data that can be represented as time series data, such as:

- (i) user logins per second;
- (ii) video stream requests per minute;
- (iii) PDF download requests per minute; and
- (iv) sales volume per hour.

[0045] The metrics are dynamic characteristics of the system, and can show change or anomalies spatially and temporally. Spatial anomalies are when the behavior of one server is different than other servers at a point in time. Temporal anomalies are when the behavior of one or more servers is changing with respect to time.

[0046] Metrics can be of two general types: a) synchronous time series metrics, and b) asynchronous time-stamped log entries.

[0047] Synchronous time series metrics can take the form of a series of time stamp and value tuples generated at a regular, synchronous interval. Examples can include the number of user requests per second, and the CPU load per second.

[0048] Asynchronous time-stamped log entries generally correspond to the logging of events, the events able to occur at any time and not necessarily synchronous to other events. Asynchronous time-stamped log entries can be

- 13 -

bursty, meaning that the rate at which asynchronous time-stamped log entries occur can vary as a function of time.

[0049] Asynchronous time-stamped log entries can be transformed by the system into synchronous time series metrics by extracting one or more log entries at regular time intervals and generating a value for each time stamp. The value can be based on the type of log entry.

[0050] For example, for some log entries such as user requests to a server, the system can construct a time series describing the volume of user requests by computing the number of log entries (user requests in this example) within a specified time interval. The volume of user requests can be expressed, for example, as the number of user requests per second.

[0051] In another example, such as packet data logs, the system can compute the average packet size over a specified time interval to generate a synchronous time series of average packet size per second. In yet another example, the system can compute a minimum value over a time interval or a maximum value over a time interval.

[0052] By transforming asynchronous time-stamped log entries into synchronous time series metrics, the system can combine two types of metrics in an analytic engine to provide improved anomaly detection.

Hybrid (Environment and Behavior) Anomaly Detection

[0053] Anomalies can also be detected through a hybrid approach of environment and behavior anomaly detection. For example, anomalies can be detected by observing code changes that lead to changes in behavior or are associated with changes in behavior.

Brief Description of Drawings

[0054] FIG. 1 is a block diagram of an embodiment of an anomaly detection system.

[0055] FIG. 2 is a block diagram of another embodiment of an anomaly detection system.

[0056] FIG. 3 illustrates a representative configuration for detection of a spatial anomaly.

[0057] FIG. 4 illustrates a representative configuration for detection of a temporal anomaly.

[0058] FIGS. 5 and 6 are flowcharts illustrating an embodiment of dynamic anomaly detection.

[0059] FIG. 7 is a flowchart illustrating configuration of an agent on a server in an embodiment of an anomaly detection system.

[0060] FIG. 8 is an example of a chart from a behavior anomaly report generated by an anomaly detection system.

[0061] FIG. 9 is an example of a plot from a behavior anomaly report generated by an anomaly detection system.

[0062] FIG. 10 is an example of a plot illustrating cluster cohesiveness.

[0063] FIG. 11 is a flow chart showing a method for computing differences between deployed instances.

[0064] FIG. 12 is another example of a chart from a behavior anomaly report generated by an anomaly detection system.

- 15 -

[0065] FIG. 13 is another example of a plot from a behavior anomaly report generated by an anomaly detection system.

[0066] FIG. 14 is another example of a plot illustrating cluster cohesiveness.

Detailed Description of Preferred Embodiment(s)

[0067] FIG. 1 is a block diagram of an embodiment of anomaly detection system 100. FIG. 1 comprises a first set of elements associated with a Software as a Service (SaaS) provider and a second set of elements associated with one or more clients of the SaaS provider.

[0068] The first set of elements comprises management console 110 connected to database 115. Management console 110 is connected to third party applications 120 via Applications Programming Interface (API) 125. System administrator 130 (sometimes referred to as DevOps) can communicate with management console 110 via API 125. Management console 110 is connected to client provisioning system 140 and client systems 150A and 150B via relay server 160.

[0069] Client system 150A comprises first server 152A with first installed agent 154A, and second server 156A with second installed agent 158A. Client system 150B comprises server 152B with installed agent 154B.

[0070] Any suitable protocol can be used by relay server 160 to communicate with agents 154A, 158A and 154B. In some embodiments, Extensible Messaging and Presence Protocol (XMPP) can be used as the communications protocol between relay server 160 and agents 154A, 158A and 154B. XMPP facilitates client-server messaging using two open-ended XML streams. XMPP

- 16 -

has been used as a chat protocol, for example for instant messaging. In embodiments of system 100 using XMPP, one benefit of the protocol are security features built into the specifications. An XMPP connection is authenticated using Simple Authentication and Security Layer (SASL) protocol and encrypted using Transport Layer Security (TLS) protocol. Another benefit of XMPP is its presence capability. As an agent is started, the agent announces itself to the server automatically, with no need for user involvement.

[0071] FIG. 2 is a block diagram of another embodiment of anomaly detection system 200. FIG. 2 illustrates an enterprise license deployment of anomaly detection system 200.

[0072] Anomaly detection system 200 comprises management console 210 connected to database 215. Management console 210 is connected to third party applications 220 via Applications Programming Interface (API) 225. System administrator 230 (sometimes referred to as DevOps) can communicate with management console 210 via API 225. Management console 210 is connected to client provisioning system 240 and client system 250 via relay server 260.

[0073] Client system 250 comprises first server 252 with first installed agent 254, and second server 256 with second installed agent 258.

[0074] Any suitable protocol can be used by relay server 260 to communicate with agents 254 and 258. In some embodiments, Extensible Messaging and Presence Protocol (XMPP) can be used as the communications protocol between relay server 260 and agents 254 and 258.

[0075] FIG. 3 illustrates an example configuration 300 for detection of a spatial anomaly. Configuration 300 comprises anomaly detector 310. Anomaly

- 17 -

detector 310 is connected to cluster of servers 320. Cluster 320 comprises N servers, 320-1 through 320-N. In the example shown, and with the exception of server 320-M, servers 320-1 through 320-N each comprise version A of an installed software application. Server 320-M comprises version B of an installed software application. Anomaly detector 310 can be configured to detect 320-M as a *spatial* anomaly.

[0076] Typically, a spatial anomaly can be determined to have occurred if the configuration of one server is substantively different to the configuration on another server. For example, a substantive difference can be a difference that is unexpected, significant or likely to cause difference in the performance of the system. A spatial anomaly can also be determined to have occurred if the configuration is different with respect to a configuration element designated by a user.

[0077] In one embodiment, the configuration can be one or more installed software applications. In another embodiment, the configuration can be one or more configuration files. The configuration files can specify the configuration of a specific application, for example the configuration of an Apache web server, or the configuration of a server's operating system (OS) such as open or closed ports, time zone or OS version.

[0078] FIG. 4 illustrates an example configuration 400 for detection of a temporal anomaly. Configuration 400 comprises anomaly detector 410. Anomaly detector 410 is connected to cluster of servers 420. Cluster 420 comprises N servers, 420-1 through 420-N. The configuration 400 is shown at two times in FIG. 4, at first time t_1 and at second time t_2 occurring after time t_1 .

- 18 -

[0079] At time t_1 , in the example shown, servers 420-1 through 420-N each comprise version A of an installed software application.

[0080] At time t_2 , in the example shown, servers 420-1 through 420-N each comprise version A of an installed software application with the exception of server 420-M. Server 420-M comprises version B of an installed software application. Anomaly detector 410 can be configured to detect 420-M as a *temporal* anomaly.

[0081] Detecting differing software versions on various servers is merely illustrative of one type of the many types of anomalies that can be detected using the anomaly detection system. Its illustration here is in no way meant to be viewed as a limitation.

[0082] FIGS. 5 and 6 are flowcharts illustrating an embodiment of dynamic anomaly detection. At 510 of FIG. 5, in preparation for anomaly detection, one or more agents are configured on one or more servers, such as illustrated in FIG. 1 for example. At 520, a metric is taken. The metric is one of a set of metrics to be taken. The set of metrics can include, for example, memory usage, CPU usage, disk I/O, traffic, and load.

[0083] The time over which the system is observed can be divided into one or more time windows. The time windows can be of equal or unequal duration. The duration of the time windows can be selected based at least in part on the resolution of the data being collected from which the metrics are being determined.

[0084] Metrics can be real-time metrics or can be replays of metrics collected at a previous time, for example from historical data.

- 19 -

[0085] In some embodiments, the metrics are normalized. In some embodiments, the metric data are corrected, for example to compensate for small values which could adversely affect the results of the method.

[0086] At 530, a check is performed to see if all desired metrics belonging to the set of metrics have been taken. If no, then method 500 returns to step 520. If yes, then method 500 proceeds to step 540.

[0087] At 540, a check is performed to see if a time window for metric collection has expired. If no, then method 500 returns to step 520. If yes, then method 500 proceeds to step 550.

[0088] At 550, statistics and/or characteristics are calculated for the metrics. Statistics can include, but are not limited to, mean, standard deviation, kurtosis and entropy. At 560, the statistics for the metrics are transformed into a feature vector in n-dimensional space, and additional spectral and/or spatial features can be added. In some embodiments, wavelet decomposition is used to characterize the behavior of one or more of the servers. In some embodiments, the statistics and a set of wavelet coefficients are concatenated into a single n-dimensional feature vector. In other embodiments, Fourier coefficients are used to characterize the behavior of one or more servers, and are added to, or used in, the n-dimensional feature vector.

[0089] The n-dimensional feature vector can be a concatenation of data selected from the group consisting of i) the values of one or more metrics during a time window, ii) the statistical characteristics of the metric, and iii) the spectral characteristics of the metric. Spectral characteristics can include, for example, Fourier coefficients and wavelet coefficients.

- 20 -

[0090] At 570, a check is performed to see if statistics have been calculated for all desired servers. If no, then method 500 returns to step 550. If yes, then method 500 proceeds to step 610 of FIG. 6.

[0091] At 610 of FIG. 6, method 600 computes the center of mass of the feature vectors in n-dimensional space.

[0092] In an example embodiment, the center of mass can be computed as the average of the values of the feature vector along each dimension. In another example, the center of mass can be computed as the median of the values of the feature vector along each dimension. A benefit of using the median is that the center of mass can be less sensitive to outliers in the set of feature vectors. Other suitable measures can be used to compute the center of mass.

[0093] At 620, a measure of distance from a server to the center of mass is computed. The measure of distance can be one selected from the group consisting of Euclidean distance, L1 (Manhattan) distance and Canberra distance, or another suitable measure of distance. At 630, one or more thresholds are used to determine a degree of anomaly based on the measure of distance.

[0094] In some embodiments, more than one measure of distance is computed, and the results used in combination to provide an aggregate score.

[0095] Adaptive thresholds can be used for determining the degree of anomaly (also known as the anomaly score) based on distance of a server from the center of mass of the cluster. Machine learning can be employed to adjust and configure suitable thresholds. One approach is for the anomaly detection system to send alerts to the operator, and the operator to provide feedback to the system. The system can learn from the feedback and can adjust the threshold(s).

- 21 -

[0096] An adaptive threshold can be computed as the weighted average of one or more thresholds.

[0097] One method to computing a threshold is as follows. The method first computes the standard deviation of the values, and then sorts the data by value so that they are arranged in order from the largest value to the smallest value. In the next step, the method removes the smallest data point and re-computes the standard deviation for the remaining values. The method then checks to see if the re-computed standard deviation has changed significantly from the original standard deviation, for example if the change in standard deviation exceeds a predetermined threshold. If the re-computed standard deviation has changed significantly, then the adaptive threshold is set to the value of the data point last removed from the data set. If the re-computed standard deviation has not changed significantly, then the steps described above are repeated until the adaptive threshold is set. By this method, the adaptive threshold is set to the value of the smallest data point that has a significant influence on the standard deviation of the data set.

[0098] Another method for computing an adaptive threshold is based on an adjusted boxplot method. In a first step, the method divides the data into quartiles, two quartiles on either side of the median. Next, the method computes the lower quartile value, the inter-quartile range, and a gamma value based on a measurement of the degree of skew of the data. The method can then set the adaptive threshold to a value of the sum of the lower quartile value and the product of the gamma value and the inter-quartile range.

- 22 -

[0099] In some embodiments, the system can be configured to compute a True Positive Ratio (TPR) and a False Positive Ratio (FPR) to measure the effectiveness of the method at detecting anomalies. Furthermore, in some embodiments, the system can be configured with a threshold on the number of false positive alerts, depending on how chaotic the environment is. In some embodiments, the system can learn from the environment and adapt to it.

[00100] At 640, a check is performed to see if the degree of anomaly has been determined for all desired servers. If no, then method 600 returns to step 620. If yes, then method 600 proceeds to step 650. At 650, method 600 monitors the persistence of the anomaly. At 660, method 600 generates an alert.

[00101] In some embodiments, a technique can be used to reduce the number of dimensions of the feature vector to a smaller number, for example two dimensions. Principal components analysis (PCA) is an example of a technique suitable for reducing dimensionality. The alert generated by the anomaly detection system can comprise a two-dimensional display of results in the form of a heat map showing spatial and/or temporal anomalies. Anomalies can be displayed as “hotspots” in the heat map. In some embodiments, the alert can comprise a cluster cohesiveness report describing the degree of cohesiveness of the cluster of servers and highlighting at least some anomalous conditions or behavior on one or more servers in the cluster.

[00102] FIG. 7 is a flowchart illustrating configuration of an agent on a server in an embodiment of an anomaly detection system such as anomaly detection system 100 of FIG. 1. At 710, the agent is installed on the server. The server is associated with a client. At 720, the server is provisioned with software using a

- 23 -

gold master. The gold master is the standard against which anomalies can be detected.

[00103] At 730, a management console of the anomaly detection system, such as management console 110 of FIG. 1, opens a chat-based session. At 740, the server wakes up and establishes the session. At 750, the management console sends a unique identifier (ID) to the agent installed on the server. At 760, the management console adds the agent and agent ID to a list of agents and agent IDs maintained at the management console and associated with the client.

[00104] FIG. 8 is an example of chart 800 from a behavior anomaly report generated by an anomaly detection system such as system 100 of FIG. 1. Chart 800 comprises a graphical indication of anomalous behavior, shown as a function of time, for each of four servers (Servers 1, 2, 3 and 4). In the example chart 800, Server 1 and Server 4 exhibit no anomalous behavior. Server 2 exhibits anomalous behavior 820 between times t_3 and t_4 . Server 3 exhibits anomalous behavior 830A through 830G between times t_1 and t_2 . Chart 800 can be color-coded, for example, according to the type and severity of anomaly. Chart 800 can be a graphical indication of when alerts to the operator were generated by the anomaly detection system.

[00105] Alerts generated by the system can be rated according to a determined level of severity, the level of severity determined at least in part based on one or more factors selected from the group consisting of i) the anomaly score, ii) the persistence of the anomaly, and iii) divergence from previous patterns.

[00106] The system can be configured to send a message to a user or an operator of the system in response to an alert. The message can be an email

- 24 -

message, for example. The system can also be configured to trigger other actions in response to an alert. For example, the system can generate an HTTP POST to a throttling mechanism to cause an anomalous server to be placed in quarantine.

[00107] FIG. 9 is an example of a plot from a behavior anomaly report generated by an anomaly detection system such as system 100 of FIG. 1. The plot of FIG. 9 corresponds to chart 800 of FIG. 8, for example. FIG. 9 shows memory usage of four servers (Servers 1, 2, 3 and 4) as a function of time. FIG. 9 indicates that, between times t_1 and t_2 , memory usage of Server 3 falls significantly below its normal value. The drop in memory usage between times t_1 and t_2 is detected by the anomaly detection system as anomalous behavior, as also shown in FIG. 8.

[00108] FIG. 10 is an example of a plot illustrating cluster cohesiveness. The plot of FIG. 10 comprises a first dimension (Dimension 1) along the x-axis, and a second dimension (Dimension 2) along the y-axis. In some embodiments, these two dimensions (Dimension 1 and Dimension 2) are a projection of a much higher-dimensional space, onto the most significant two dimensions using PCA. The behavior of four servers (Servers 1, 2, 3 and 4) is plotted as a function of Dimensions 1 and 2 for a period of time. The behavior of Servers 1, 2 and 4 is clustered around the point (x_c, y_c) . The behavior of Server 3 is clustered around the point (x_3, y_3) .

[00109] In some embodiments, a tree differencing method can be used to show differences between deployed instances of systems and application software. The method computes the difference between two groups of tree-structured data.

- 25 -

[00110] FIG. 11 is a flow chart showing a method 1100 for computing differences between deployed instances.

[00111] The first step is to define and compute hash values for each node. The next step is to use hash values of sub-trees to match the largest sub-trees that are identical between two trees. The number of matches is expanded by considering unmatched parents and children of matched nodes.

[00112] In the first step described above (step 1110 of FIG. 11), a set of attributes are selected for each tree node for computing the hash value of that node. With one traversal of each tree (step 1120), the hash value for each node can be computed using a non-cryptographic hash function suitable for general hash-based lookup (such as MurmurHash). In the next step (step 1130), starting from largest sub-trees, the method matches pairs of nodes having the same hash values.

[00113] In expanding the number of matches (step 1140), the method matches parents of each pair of matched nodes provided they are unpaired, have the same properties and have no children matched to different nodes. In addition, the method matches children of each pair of matched nodes provided they are unpaired and have the same properties.

[00114] In step 1150, the method computes a delta value representing the degree of difference between two trees. The method first computes all the deletions and insertions for unmatched nodes, and the effect of the deletions and insertions. Next, the method computes all of the move operations by finding matched pairs of nodes that have non-matching parents. The method also finds

- 26 -

all matched pairs having different content. In a final step, the method computes the delta value.

[00115] FIG. 12 is another example of a chart from a behavior anomaly report generated by an anomaly detection system such as system 100 of FIG. 1. Chart 1200 comprises a graphical indication of anomalous behavior, shown as a function of time, for each of four entities 1210, 1212, 1214 and 1216. Entities 1210, 1212, 1214 and 1216 are sources of data that can be input to an anomaly detection system.

[00116] In one example embodiment, entities 1210, 1212, 1214 and 1216 can be software applications, and chart 1200 comprises a graphical indication of anomalous behavior for each of the four software applications 1210, 1212, 1214 and 1216.

[00117] In another example embodiment, entities 1210, 1212, 1214 and 1216 can be sensors, and chart 1200 comprises a graphical indication of anomalous behavior for each of the four sensors 1210, 1212, 1214 and 1216. The anomaly detection system can receive data from each of the four sensors 1210, 1212, 1214 and 1216, and can apply the same methods described above to detect anomalous behavior and generate alerts to a user or an operator.

[00118] FIG. 13 is another example of a plot from a behavior anomaly report generated by an anomaly detection system such as system 100 of FIG. 1. The plot of FIG. 13 corresponds to chart 1200 of FIG. 12, for example. FIG. 13 shows a metric for each of the four entities 1210, 1212, 1214 and 1216 as a function of time. FIG. 13 indicates that, between times t_1 and t_2 , the value of the metric of entity 1214 falls significantly below its normal value.

- 27 -

[00119] FIG. 14 is another example of a plot illustrating cluster cohesiveness. The plot of FIG. 14 comprises a first dimension (Dimension 1) along the x-axis, and a second dimension (Dimension 2) along the y-axis. In some embodiments, these two dimensions (Dimension 1 and Dimension 2) are a projection of a much higher-dimensional space, onto the most significant two dimensions using PCA. The behavior of four entities 1210, 1212, 1214 and 1216 is plotted as a function of Dimensions 1 and 2 for a period of time. The behavior of entities 1210, 1212 and 1216 is clustered around the point (x_1, y_1) . The behavior of entity 1214 is clustered around the point (x_2, y_2) .

[00120] In various of the embodiments described above:

- (i) the metrics can be for example memory usage, CPU usage, traffic and/or load; and/or
- (ii) the statistics can be for example mean, standard deviation, kurtosis and/or entropy; and/or
- (iii) the measure of distance can be for example Euclidean distance, L1 (Manhattan) distance and/or Canberra distance, or another suitable measure of distance; and/or
- (iv) the spectral characteristics can be for example Fourier coefficients and/or wavelet coefficients; and/or
- (v) the projection of feature vectors in a higher dimension space onto feature vectors in a lower dimension space can be derived for example from a Principal Components Analysis.

- 28 -

[00121] While particular elements, embodiments and applications of the present invention have been shown and described, it will be understood, that the invention is not limited thereto since modifications can be made without departing from the scope of the present disclosure, particularly in light of the foregoing teachings.

- 29 -

What is claimed is:

1. A method for detecting a spatial environment anomaly in a system comprising a plurality of servers, the method comprising:

- (a) monitoring a configuration of each of said plurality of servers; and
- (b) generating an alert associated with a configuration difference selected from the group consisting of
 - (i) a first server of said plurality of servers having a substantively different configuration than other servers of said plurality of servers, and
 - (ii) a first server of said plurality of servers having a different configuration than other servers of said plurality of servers with respect to a configuration element designated by a user.

2. The method of claim 1, wherein said configuration is a software configuration.

3. The method of claim 2, wherein said software configuration is one or more installed software applications.

4. The method of claim 2, wherein said software configuration is one or more configuration files.

- 30 -

5. The method of claim 4, wherein at least one of said configuration files specifies the configuration of an installed software application.

6. A method for detecting a temporal environment anomaly in a server, the method comprising:

- (a) monitoring a configuration of said server at a first time;
- (b) monitoring said configuration of said server at a second time; and
- (c) generating an alert associated with a change in said configuration between said first time and said second time, said change selected from the group consisting of
 - (i) said change being more than a predetermined threshold between said first time and said second time; and
 - (ii) said change being with respect to a configuration element designated by a user.

7. The method of claim 6, wherein said configuration is a software configuration.

8. The method of claim 7, wherein said software configuration is one or more installed software applications.

9. The method of claim 7, wherein said software configuration is one or more configuration files.

- 31 -

10. The method of any of claims 6 to 9, wherein said server is one of a plurality of servers in a system.

11. A method of detecting a spatial behavior anomaly in a system comprising a plurality of servers, the method comprising:

- (a) receiving a set of metrics from each of said plurality of servers at one or more times during a time period;
- (b) segmenting said time period into one or more time windows;
- (c) calculating one or more statistics for at least one metric of said set of metrics for each of said plurality of servers for each of said time windows;
- (d) concatenating said metrics and said statistics into a corresponding n-dimensional feature vector for each of said plurality of servers;
- (e) computing a center of mass in n-dimensional space of said feature vectors;
- (f) computing a distance measurement from the feature vector of a first server of said plurality of servers to said center of mass;
- (g) determining a degree of anomaly of said first server based on said distance measurement;
- (h) monitoring said degree of anomaly of said first server: and
- (i) generating an alert when said degree of anomaly of said first server exceeds a predetermined threshold.

- 32 -

12. The method of claim 11, wherein said metrics comprise one or more asynchronous time-stamped log entries, said method further comprising, after receiving a set of metrics from each of said plurality of servers at one or more times during a time period, the step of transforming said one or more asynchronous time-stamped log entries into one or more synchronous time series metrics.

13. The method of claim 11, further comprising, before the step of computing said center of mass, the steps of calculating one or more spectral characteristics of said metrics, and concatenating said spectral characteristics into said n-dimensional feature vector.

14. The method of any of claims 11 to 13, wherein said set of metrics is selected from a group consisting of (i) real-time metrics, and (ii) replays of metrics collected prior to the start of said time period.

15. The method of any of claims 11 to 13, wherein said metrics comprise at least one selected from the group consisting of:

- (i) synchronous time series metrics; and
- (ii) asynchronous time-stamped log entries.

16. The method of any of claims 11 to 13, wherein said metrics are selected from the group consisting of memory usage, CPU usage, traffic and load.

- 33 -

17. The method of any of claims 11 to 13, wherein said statistics are selected from the group consisting of mean, standard deviation, kurtosis and entropy.

18. The method of claim 13, where said spectral characteristics are selected from the group consisting of wavelet coefficients and Fourier coefficients.

19. The method of any of claims 11 to 13, wherein said distance measurement is one selected from a group consisting of Euclidean distance, L1 (Manhattan) distance, and Canberra distance.

20. The method of any of claims 11 to 13, wherein said n-dimensional feature vector is a projection of a higher dimensional space onto n dimensions.

21. The method of claim 20, wherein said projection is derived from a Principal Components Analysis.

22. The method of any of claims 11 to 13, wherein said set of metrics is received by an agent installed on each of said plurality of servers.

23. The method of claim 12, further comprising, before the step of computing said center of mass, the steps of calculating one or more spectral

- 34 -

characteristics of said metrics, and concatenating said spectral characteristics into said n-dimensional feature vector.

24. A method of detecting a temporal behavior anomaly in a server, the method comprising:

- (a) receiving a set of metrics from said server at one or more times during a time period;
- (b) segmenting said time period into one or more time windows;
- (b) calculating one or more statistics for at least one metric of said set of metrics for said server for each of said time windows;
- (c) concatenating said metrics and said statistics into a corresponding n-dimensional feature vector for each of said time windows;
- (d) computing a center of mass in n-dimensional space of said feature vectors;
- (e) computing a distance measurement from a first feature vector of said feature vectors to said center of mass, said first feature vector corresponding to a first time window;
- (f) determining a degree of anomaly of said server based on said distance measurement;
- (g) monitoring said degree of anomaly of said server; and
- (h) generating an alert when said degree of anomaly of said server exceeds a predetermined threshold.

- 35 -

25. The method of claim 24, wherein said metrics comprise one or more asynchronous time-stamped log entries, said method further comprising, after receiving a set of metrics from each of said plurality of servers at one or more times during a time period, the step of transforming said one or more asynchronous time-stamped log entries into one or more synchronous time series metrics.

26. The method of claim 24, further comprising, before the step of computing said center of mass, the steps of calculating one or more spectral characteristics of said metrics, and concatenating said spectral characteristics into said n-dimensional feature vector.

27. The method of any of claims 24 to 26, wherein said set of metrics is selected from a group consisting of (i) real-time metrics, and (ii) replays of metrics collected prior to the start of said time period.

28. The method of any of claims 24 to 26, wherein said metrics comprise at least one selected from the group consisting of:

- (i) synchronous time series metrics; and
- (ii) asynchronous time-stamped log entries.

29. The method of claim 25, further comprising, before the step of computing said center of mass, the steps of calculating one or more spectral

- 36 -

characteristics of said metrics, and concatenating said spectral characteristics into said n-dimensional feature vector.

30. The method of claim 24, wherein said server is one of a plurality of servers in a system.

31. A method for detecting anomalies in a system comprising one or more servers, the method comprising:

- (a) receiving at least one metric corresponding to at least one server of the one or more servers;
- (b) computing statistics associated with said at least one metric;
- (c) applying said at least one metric and said statistics to detect at least one anomaly in said system, said at least one anomaly selected from the group consisting of spatial environment anomalies, temporal environment anomalies, spatial behavior anomalies and temporal behavior anomalies; and
- (f) generating an alert to an operator of said system.

32. The method of claim 31, wherein said alert is generated based on the detection of at least one environment anomaly and at least one behavior anomaly.

- 37 -

33. The method of any of claims 31 and 32, further comprising, after detecting said anomaly, the steps of determining the degree of said anomaly, and monitoring the persistence of said anomaly, and wherein said alert is generated when said degree exceeds a first threshold and said persistence exceeds a second threshold.

34. A method for detecting anomalies in a system comprising one or more software applications, the method comprising:

- (a) receiving at least one metric corresponding to at least one software application of the one or more software applications;
- (b) computing statistics associated with said at least one metric;
- (c) applying said at least one metric and said statistics to detect at least one anomaly in said system, said at least one anomaly selected from the group consisting of spatial environment anomalies, temporal environment anomalies, spatial behavior anomalies and temporal behavior anomalies; and
- (f) generating an alert to an operator of said system.

35. The method of claim 34, wherein said alert is generated based on the detection of at least one environment anomaly and at least one behavior anomaly.

- 38 -

36. The method of any of claims 34 and 35, further comprising, after detecting said anomaly, the steps of determining the degree of said anomaly, and monitoring the persistence of said anomaly, and wherein said alert is generated when said degree exceeds a first threshold and said persistence exceeds a second threshold.

37. A method for detecting anomalies in a system comprising one or more sensors, the method comprising:

- (a) receiving at least one metric corresponding to at least one sensor of the one or more sensors;
- (b) computing statistics associated with said at least one metric;
- (c) applying said at least one metric and said statistics to detect at least one anomaly in said system, said at least one anomaly selected from the group consisting of spatial environment anomalies, temporal environment anomalies, spatial behavior anomalies and temporal behavior anomalies; and
- (f) generating an alert to an operator of said system.

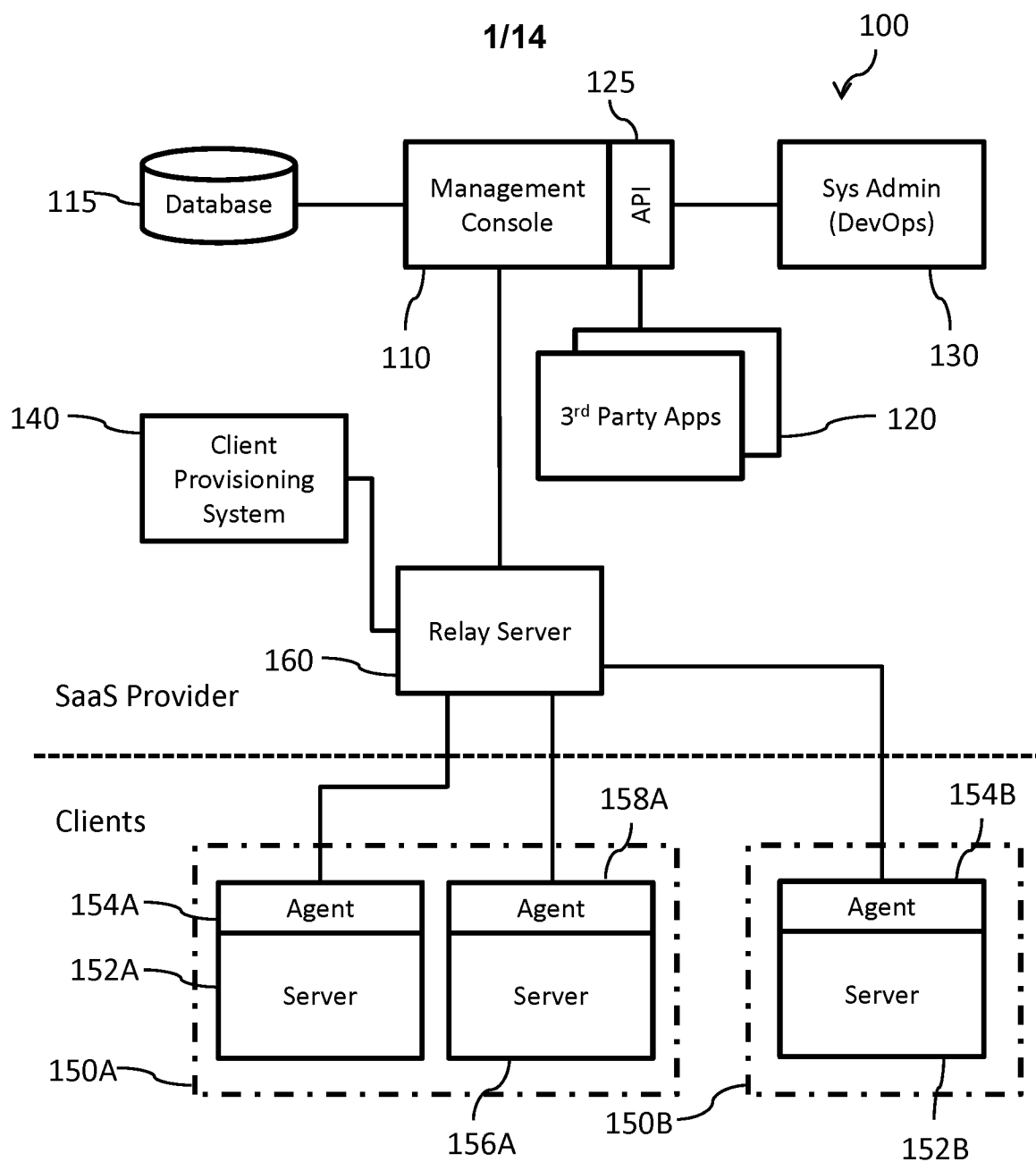
38. The method of claim 37, wherein said alert is generated based on the detection of at least one environment anomaly and at least one behavior anomaly.

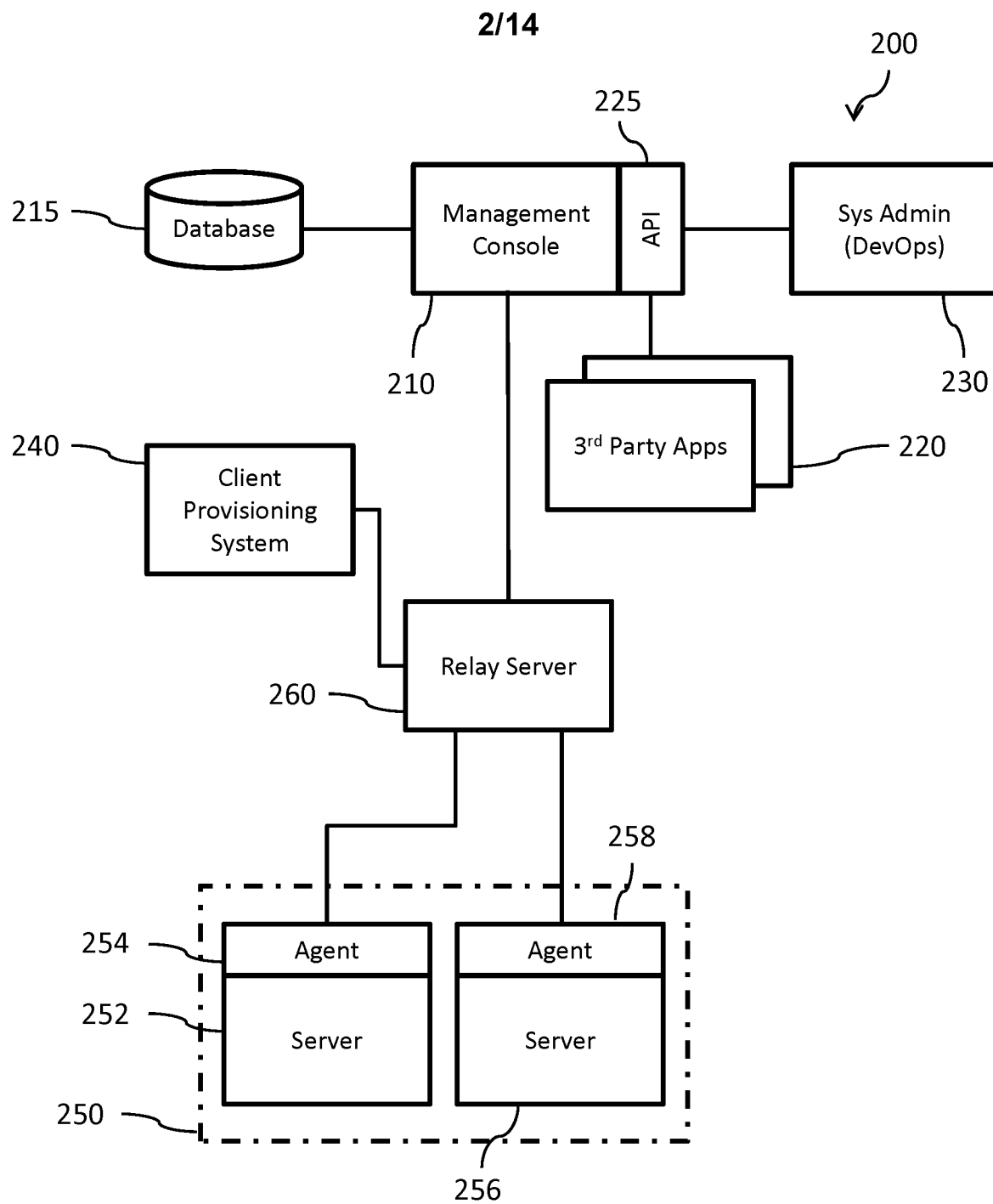
- 39 -

39. The method of any of claims 37 and 38, further comprising, after detecting said anomaly, the steps of determining the degree of said anomaly, and monitoring the persistence of said anomaly, and wherein said alert is generated when said degree exceeds a first threshold and said persistence exceeds a second threshold.

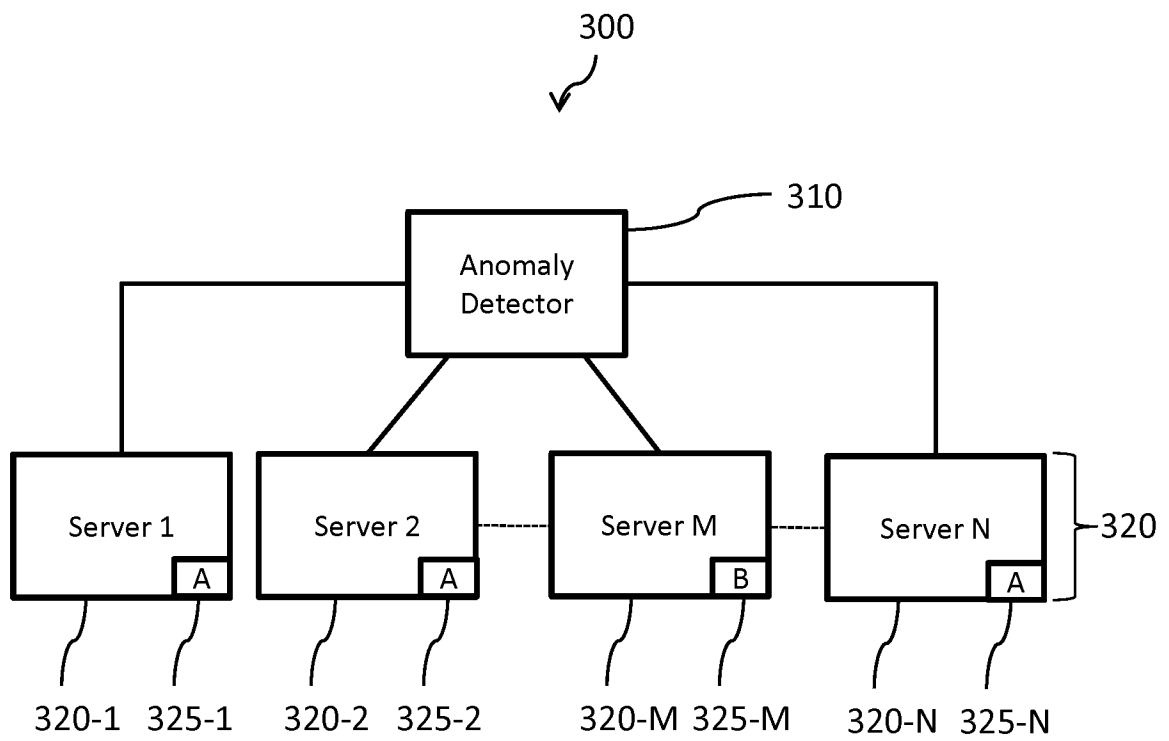
40. The method of claim 37, wherein said at least one metric comprises sensor data from said one or more sensors.

41. A device for implementing the method of any of claims 1, 6, 11, 24, 31, 34 and 37.

**FIG. 1**

**FIG. 2**

3/14

**FIG. 3**

4/14

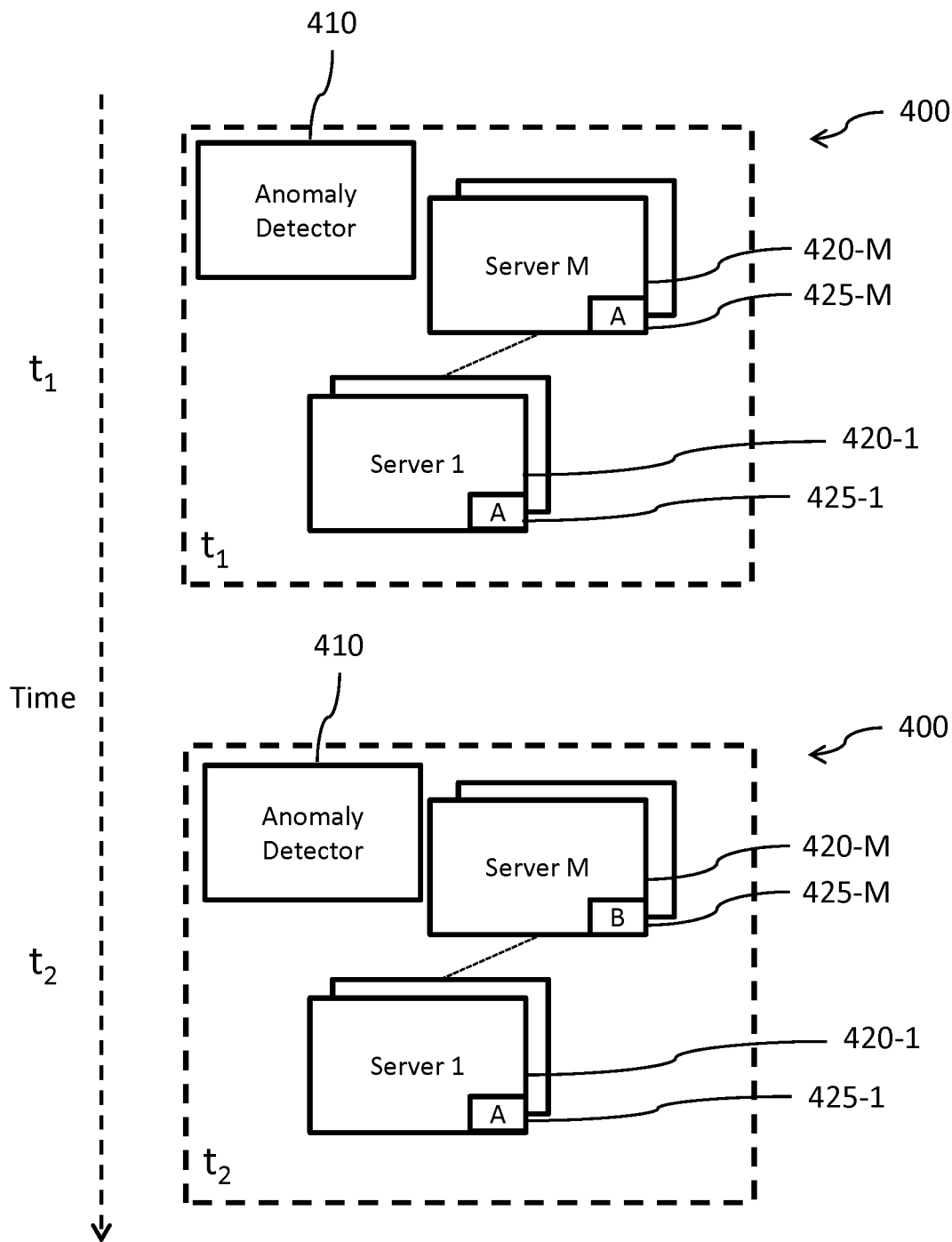
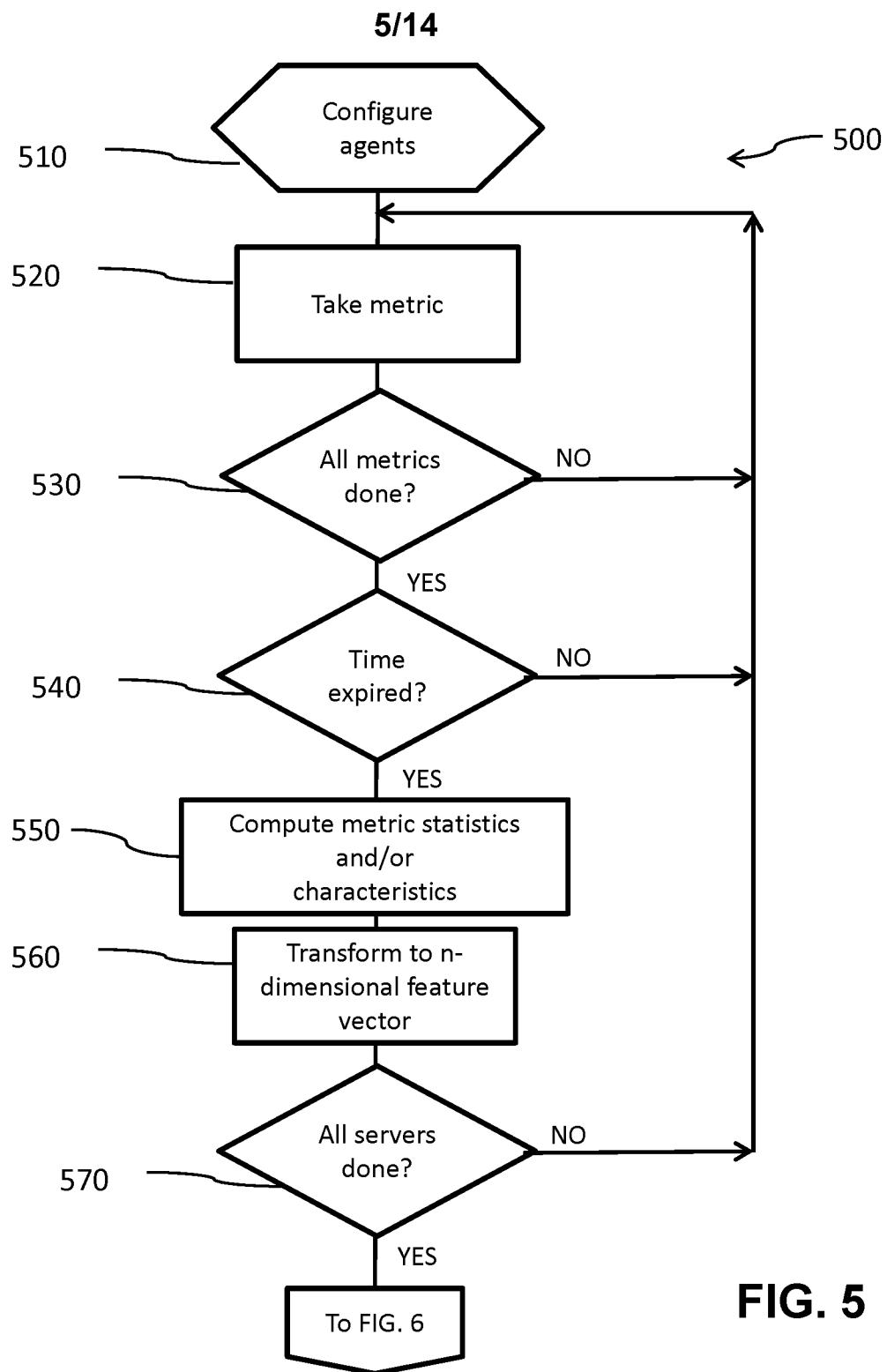


FIG. 4

**FIG. 5**

6/14

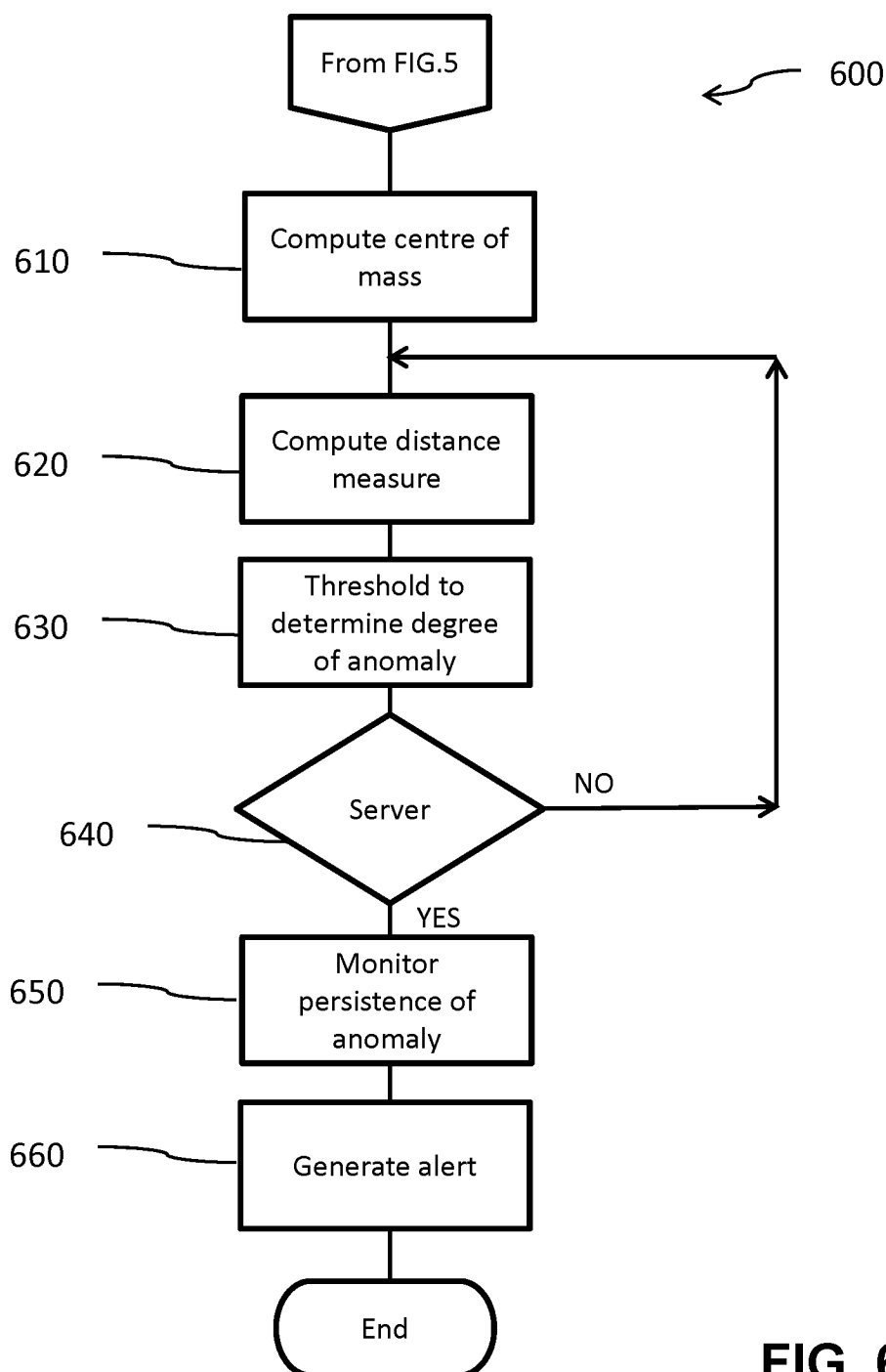


FIG. 6

7/14

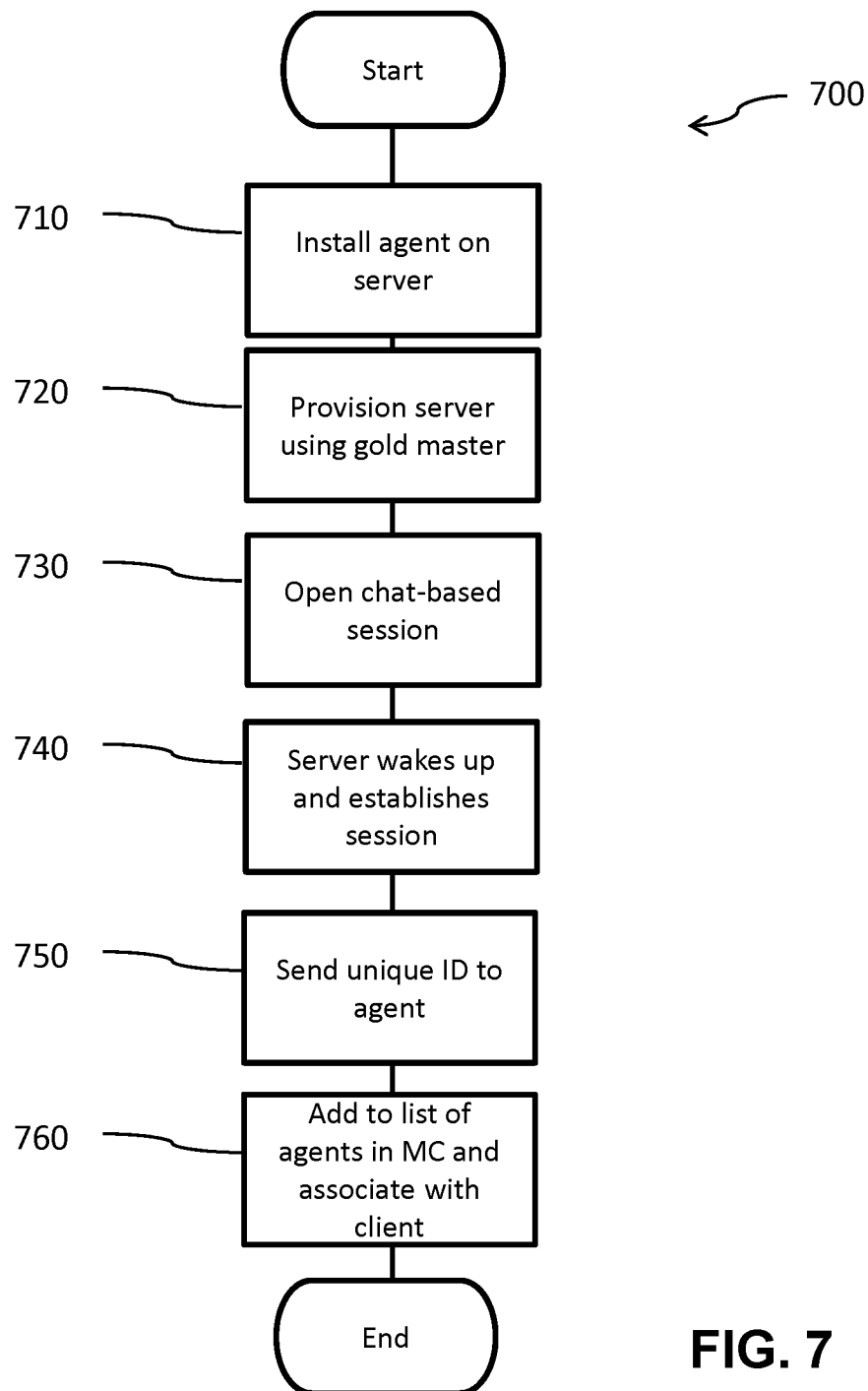


FIG. 7

8/14

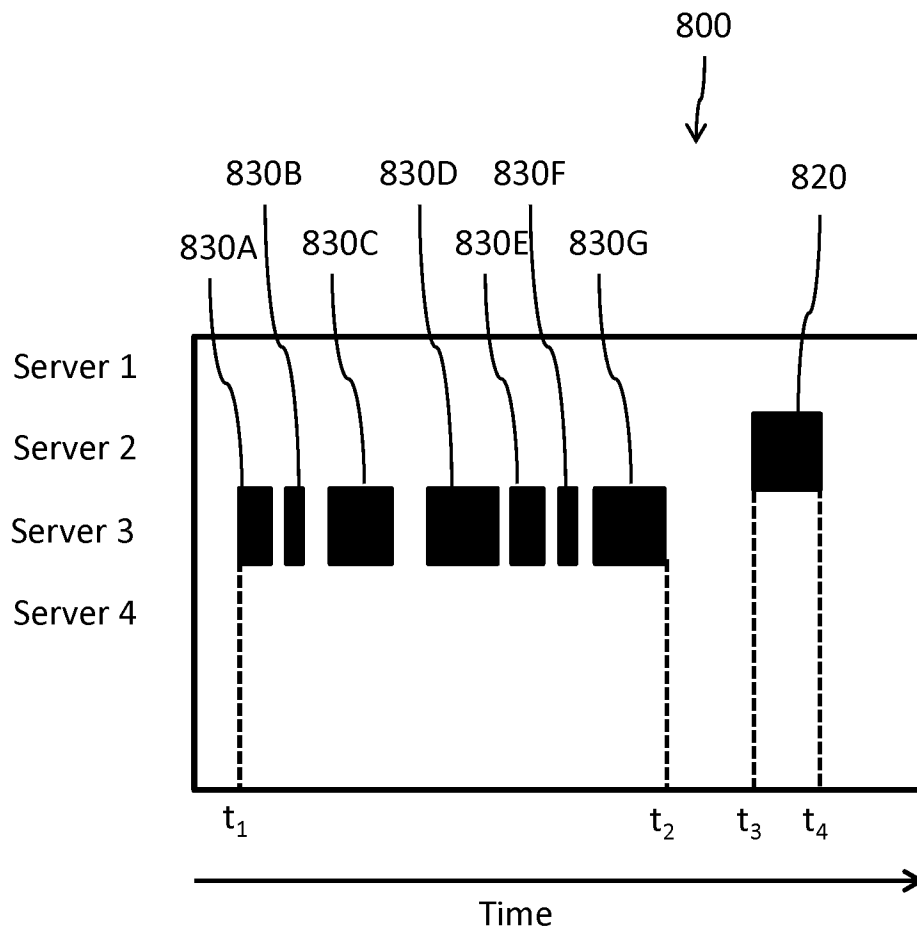
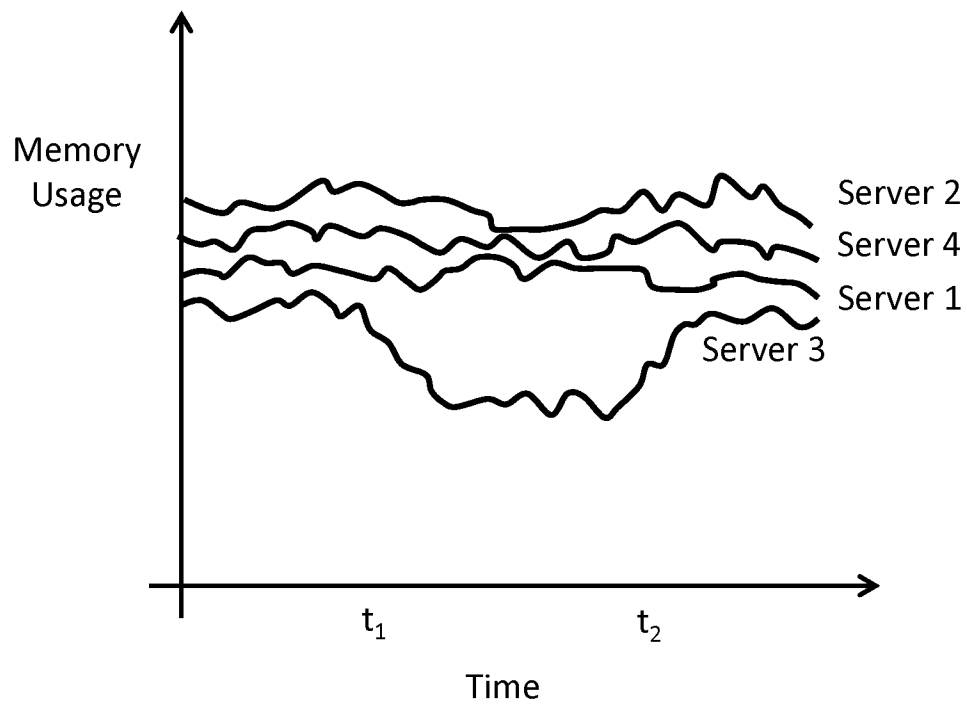
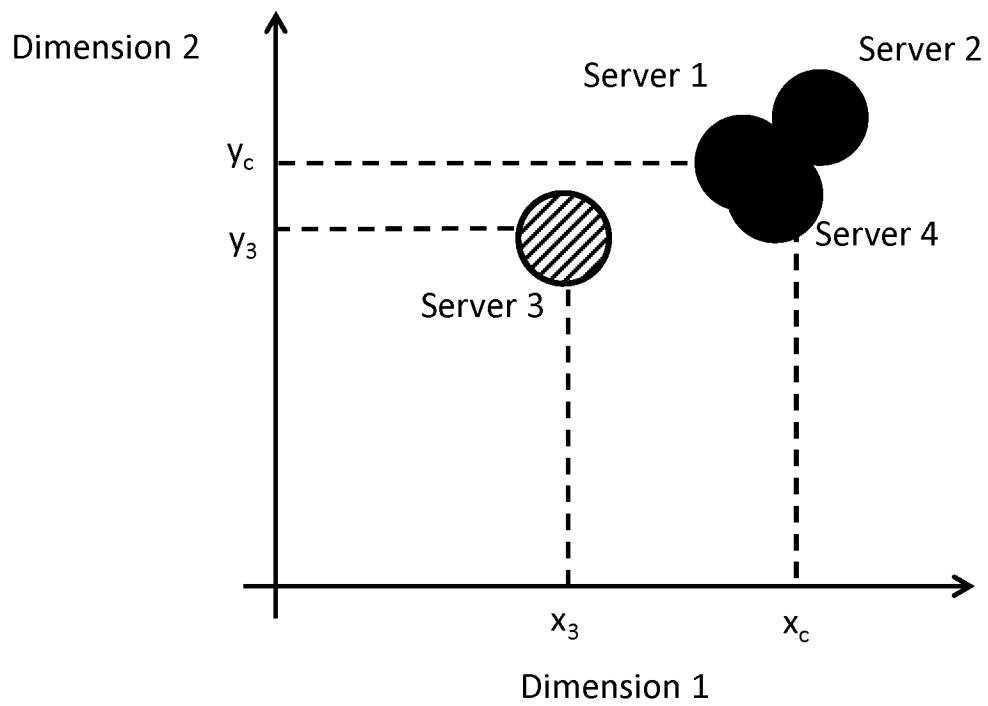


FIG. 8

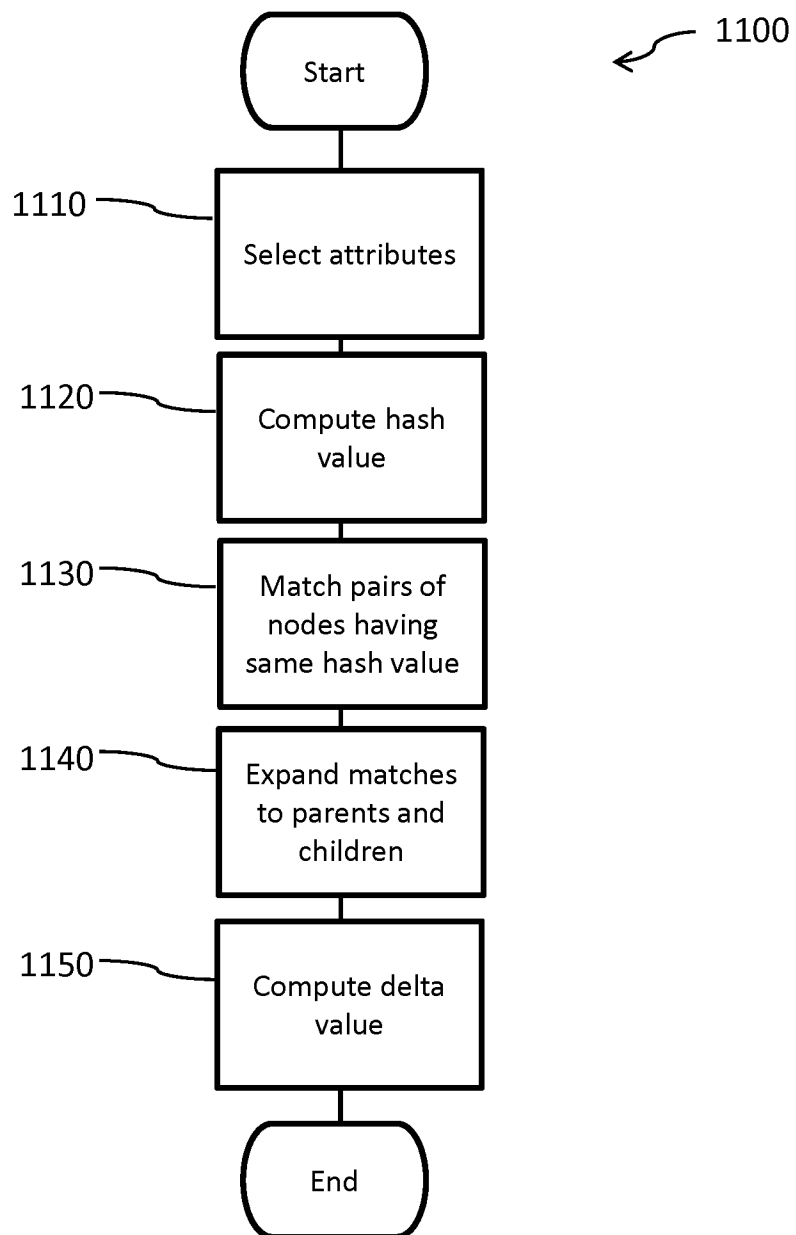
9/14

**FIG. 9**

10/14

**FIG. 10**

11/14

**FIG. 11**

12/14

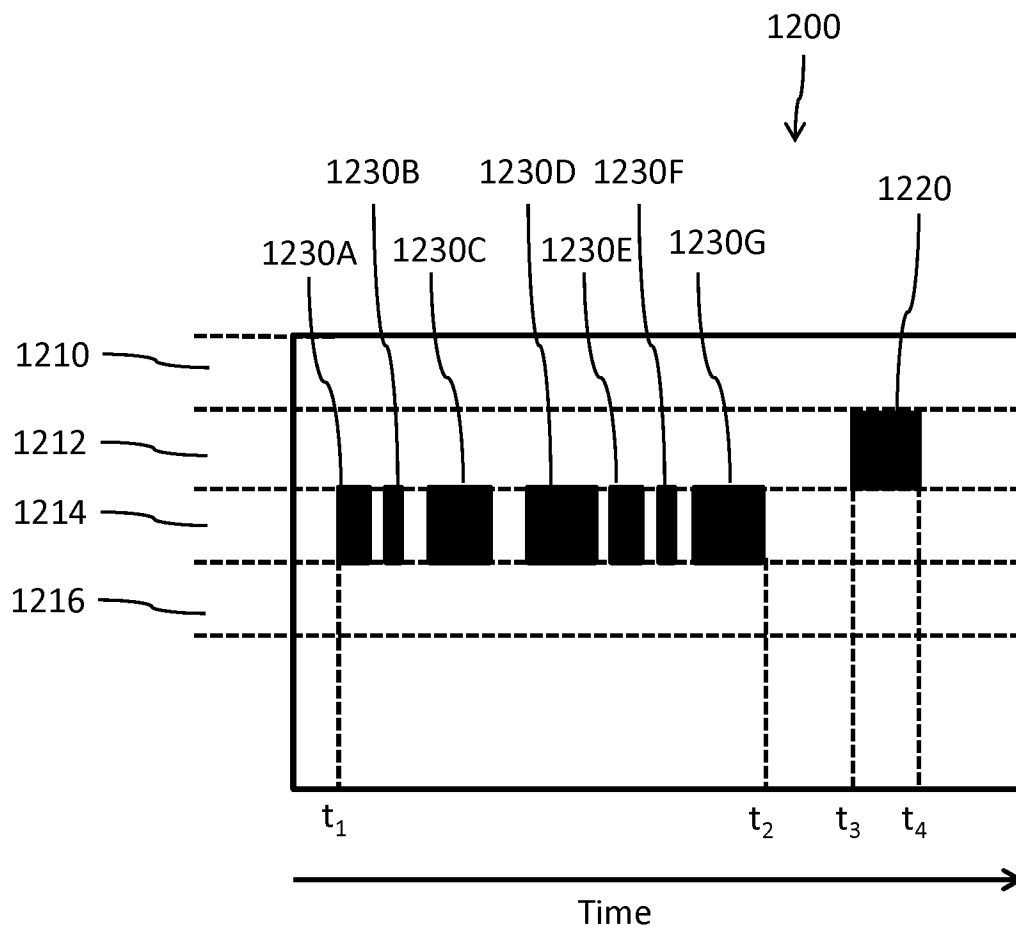
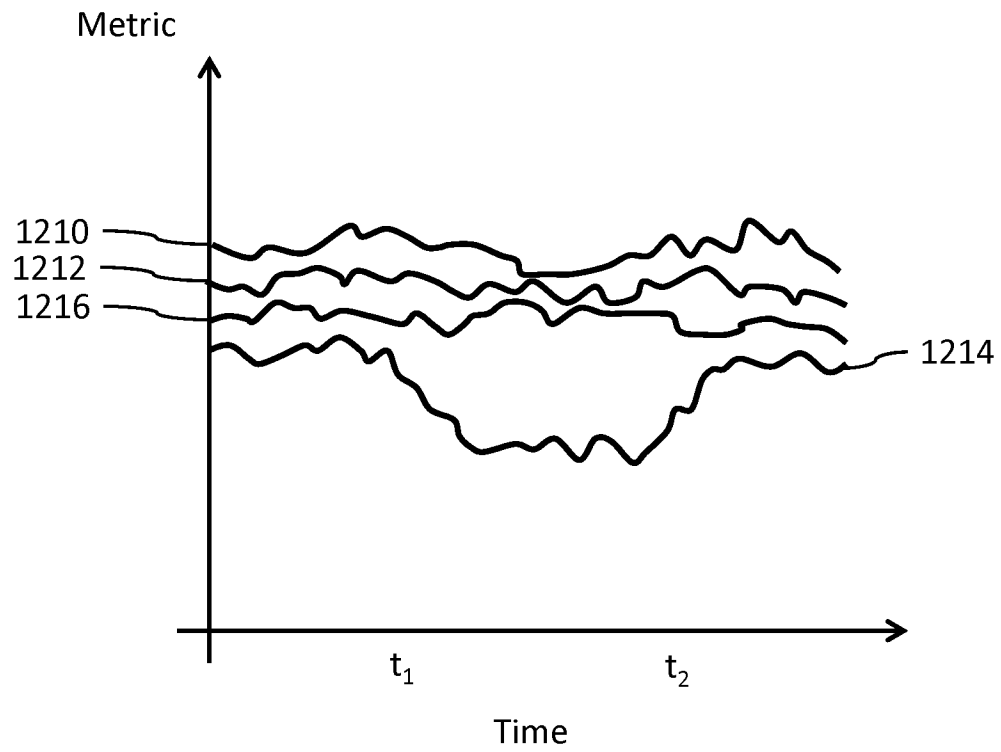


FIG. 12

13/14

**FIG. 13**

14/14

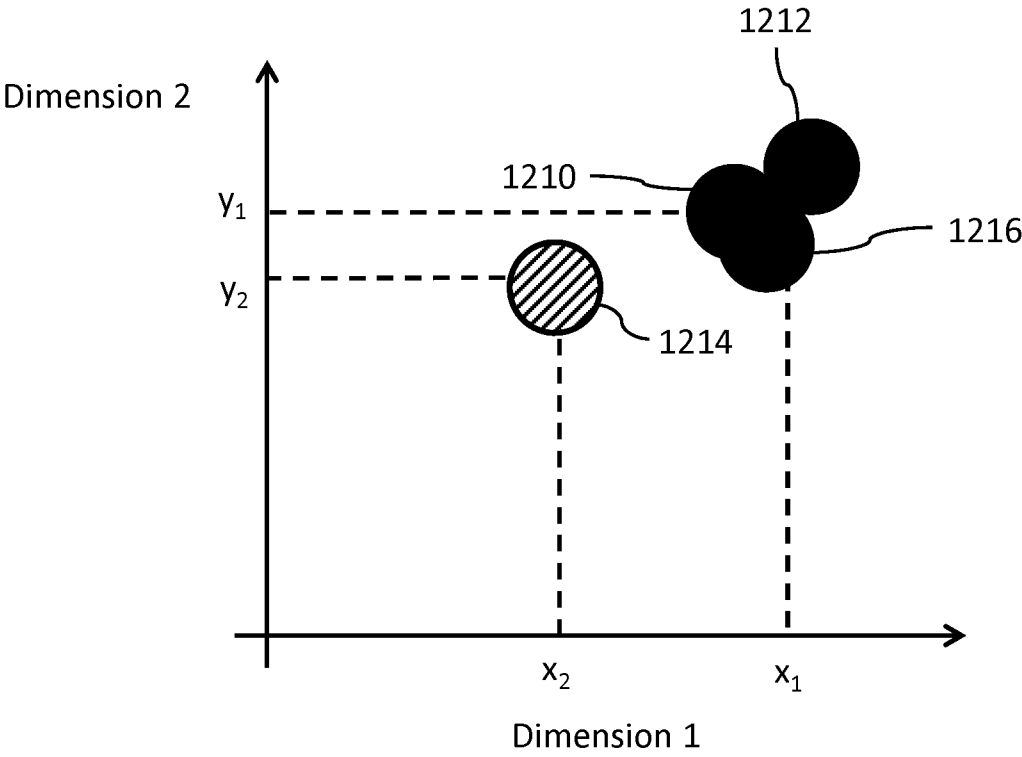


FIG. 14

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CA2014/051267

A. CLASSIFICATION OF SUBJECT MATTER
IPC: **H04L 12/26** (2006.01), **H04L 12/24** (2006.01)

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
IPC: H04L 12/26 (2006.01), H04L 12/24 (2006.01)

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

Electronic database(s) consulted during the international search (name of database(s) and, where practicable, search terms used)
Databases: Questel-Orbit (Fampat), Canadian Patents Database, IEEEExplore, Google
Keywords: detect anomaly/difference/mismatch/inconsistency, monitor server/node/cluster, configuration, generate alert/alarm, spatial/temporal, version, static, environment, threshold, software/application/file, compare, first/second time

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X Y	US 2005/0210465 A1 (SASAKI et al.) 22 September 2005 (22-09-2005) *abstract; paragraphs [0003]-[0010], [0048], [0065], [0073]; fig. 1*	1-5, 41 6-10
X Y	US 2012/0192008 A1 (SUZUKI) 26 July 2012 (26-07-2012) *abstract; paragraphs [0006]-[0018], [0024], [0058]-[0074]; figs. 8, 9, 20-27*	1-5, 41 6-10
Y	US 2009/0260000 A1 (PILANT et al.) 15 October 2009 (15-10-2009) *abstract; paragraphs [0011], [0017]*	6-10
Y	US 2009/0077171 A1 (KASHA) 19 March 2009 (19-03-2009) *abstract; paragraphs [0002]-[0008], [0022]-[0024]*	6-10
A	US 2005/0022003 A1 (OLIPHANT) 27 January 2005 (27-01-2005) *abstract; paragraphs [0005]-[0016]*	1-10, 41

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* "A" "E" "L" "O" "P"	Special categories of cited documents: document defining the general state of the art which is not considered to be of particular relevance earlier application or patent but published on or after the international filing date document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified) document referring to an oral disclosure, use, exhibition or other means document published prior to the international filing date but later than the priority date claimed	"T" "X" "Y" "&"	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 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 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
27 March 2015 (27-03-2015)

Date of mailing of the international search report
09 April 2015 (09-04-2015)

Name and mailing address of the ISA/CA
Canadian Intellectual Property Office
Place du Portage I, C114 - 1st Floor, Box PCT
50 Victoria Street
Gatineau, Quebec K1A 0C9
Facsimile No.: 001-819-953-2476

Authorized officer

Daniela Savin (819) 635-6286

INTERNATIONAL SEARCH REPORT

International application No.

PCT/CA2014/051267**Box No. II Observations where certain claims were found unsearchable (Continuation of item 2 of the first sheet)**

This international search report has not been established in respect of certain claims under Article 17(2)(a) for the following reasons:

1. ☐ Claim Nos.:
because they relate to subject matter not required to be searched by this Authority, namely:

2. ☐ Claim Nos.:
because they relate to parts of the international application that do not comply with the prescribed requirements to such an extent that no meaningful international search can be carried out, specifically:

3. ☐ Claim Nos.:
because they are dependent claims and are not drafted in accordance with the second and third sentences of Rule 6.4(a).

Box No. III Observations where unity of invention is lacking (Continuation of item 3 of first sheet)

This International Searching Authority found multiple inventions in this international application, as follows:

Group A: Claims 1-10 and 41 are directed to methods for detecting a spatial or temporal environment anomaly, in which a configuration of a server is monitored and an alert is generated when a configuration difference is detected; and

Group B: Claims 11-41 are directed to methods of detecting a spatial or temporal behaviour anomaly, in which a set of metrics is received from a server during a time period, statistics are calculated for at least one metric of the set of metrics, a degree of anomaly is determined based on the calculated statistics, the degree of anomaly is monitored, and an alert is generated when the degree of anomaly exceeds a predetermined threshold.

1. ☐ As all required additional search fees were timely paid by the applicant, this international search report covers all searchable claims.
2. ☐ As all searchable claims could be searched without effort justifying additional fees, this Authority did not invite payment of additional fees.
3. ☐ As only some of the required additional search fees were timely paid by the applicant, this international search report covers only those claims for which fees were paid, specifically claim Nos.:

4. ☒ No required additional search fees were timely paid by the applicant. Consequently, this international search report is restricted to the invention first mentioned in the claims; it is covered by claim Nos.:

Group A: Claims 1-10 and 41

Remark on Protest

- ☐ The additional search fees were accompanied by the applicant's protest and, where applicable, the payment of a protest fee.
- ☐ The additional search fees were accompanied by the applicant's protest but the applicable protest fee was not paid within the time limit specified in the invitation.
- ☐ No protest accompanied the payment of additional search fees.

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/CA2014/051267

Patent Document Cited in Search Report	Publication Date	Patent Family Member(s)	Publication Date
US2005210465A1	22 September 2005 (22-09-2005)	US7590670B2 JP2005266916A JP4354314B2	15 September 2009 (15-09-2009) 29 September 2005 (29-09-2005) 28 October 2009 (28-10-2009)
US2012192008A1	26 July 2012 (26-07-2012)	US8650444B2 CN102597966A EP2498186A1 JP5532053B2 KR20120052415A KR101436033B1 WO2011055436A1	11 February 2014 (11-02-2014) 18 July 2012 (18-07-2012) 12 September 2012 (12-09-2012) 25 June 2014 (25-06-2014) 23 May 2012 (23-05-2012) 01 September 2014 (01-09-2014) 12 May 2011 (12-05-2011)
US2009260000A1	15 October 2009 (15-10-2009)	None	
US2009077171A1	19 March 2009 (19-03-2009)	US8185621B2	22 May 2012 (22-05-2012)
US2005022003A1	27 January 2005 (27-01-2005)	US2006259593A2 US2007112941A2 US2005044389A1 US2006259779A2 US2007113100A2 US8266699B2 US2015040232A1 US8984644B2 US2005005129A1 US2006259775A2 US2007118756A2 US2005005159A1 US2006259972A2 US2005005162A1 US2006259946A2 US2007113265A2 US2005005171A1 US2006230441A2 US2007113272A2 US2007256132A2 US2014109230A1 US2015033287A1 US2015033323A1 US2015033348A1 US2015033349A1 US2015033350A1 US2015033351A1 US2015033352A1 US2015033353A1 US2015040230A1 US2015040231A1 US2015040233A1	16 November 2006 (16-11-2006) 17 May 2007 (17-05-2007) 24 February 2005 (24-02-2005) 16 November 2006 (16-11-2006) 17 May 2007 (17-05-2007) 11 September 2012 (11-09-2012) 05 February 2015 (05-02-2015) 17 March 2015 (17-03-2015) 06 January 2005 (06-01-2005) 16 November 2006 (16-11-2006) 24 May 2007 (24-05-2007) 06 January 2005 (06-01-2005) 16 November 2006 (16-11-2006) 06 January 2005 (06-01-2005) 16 November 2006 (16-11-2006) 17 May 2007 (17-05-2007) 06 January 2005 (06-01-2005) 12 October 2006 (12-10-2006) 17 May 2007 (17-05-2007) 01 November 2007 (01-11-2007) 17 April 2014 (17-04-2014) 29 January 2015 (29-01-2015) 29 January 2015 (29-01-2015) 29 January 2015 (29-01-2015) 29 January 2015 (29-01-2015) 29 January 2015 (29-01-2015) 29 January 2015 (29-01-2015) 29 January 2015 (29-01-2015) 29 January 2015 (29-01-2015) 05 February 2015 (05-02-2015) 05 February 2015 (05-02-2015) 05 February 2015 (05-02-2015)