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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
5 November 2015 (05.11.2015)

WIPO | PCT

(10) International Publication Number
WO 2015/167496 A1

(51) International Patent Classification:
G06F 17/00 (2006.01)

(21) International Application Number:
PCT/US2014/036055

(22) International Filing Date:
30 April 2014 (30.04.2014)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.** [US/US]; 11445 Compaq Center Drive W., Houston, Texas 77070 (US).

(72) Inventors: **GMACH, Daniel Juergen**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **AUYOUNG, Alvin**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **BLOCK, Robert**; 1160 Enterprise Way, Sunnyvale, California 94089 (US). **RADHAKRISHNAN, Jayaram Kallapalayam**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **PRAMANIK, Suranjan**; 1160 Enterprise Way, Sunnyvale, California 94089 (US). **STEPHEN, Julian James**; 1501 Page Mill Road, Palo Alto, California 94304-1100 (US). **SINGLA, Anurag**; 1160 Enterprise Way, Sunnyvale, California 94089 (US).

(74) Agents: **HAQ, M. Aamir** et al.; Hewlett-Packard Company, Intellectual Property Administration, Mail Stop 35 3404 E. Harmony Road, Fort Collins, Colorado 80528 (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, LT, 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, SZ, TZ,

[Continued on next page]

(54) Title: SELECTING FROM COMPUTING NODES FOR CORRELATING EVENTS

(57) Abstract: In response to determining that an event matches a condition of a rule, a given one of a plurality of computing nodes is selected to send the event, based on one or both of an attribute of the event and an identifier of the rule. Information of the event is sent to the given computing node to perform correlation of the event with another event.

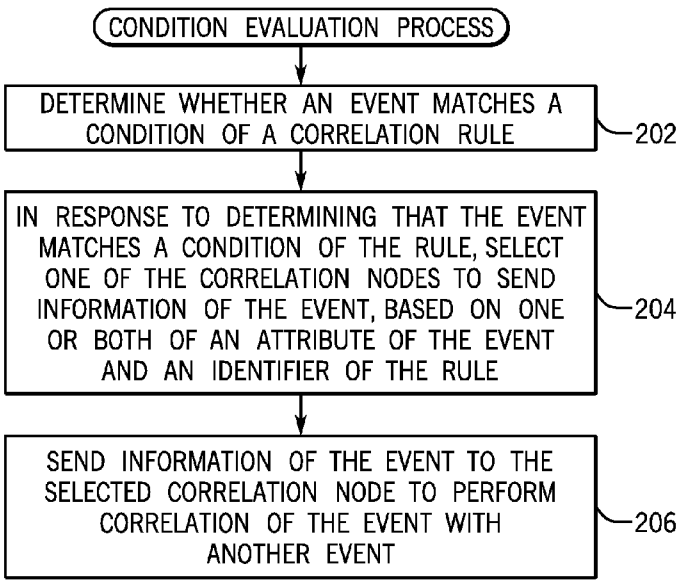

FIG. 2A

WO 2015/167496 A1

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).

Declarations under Rule 4.17:

— *as to the identity of the inventor (Rule 4.17(i))*

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

Published:

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

SELECTING FROM COMPUTING NODES FOR CORRELATING EVENTS

Background

[0001] Events can be monitored in a distributed arrangement of devices that can be coupled over one or multiple networks. The events are represented by data collected by various sources in the distributed arrangement of electronic devices. The monitored events can be analyzed for various purposes. As an example, the events can be analyzed to identify an attack on a network or an electronic device, such that real-time countermeasures can be invoked to thwart the attack or mitigate the damage caused by the attack.

Brief Description Of The Drawings

[0002] Some implementations are described with respect to the following figures.

[0003] Fig. 1 is a block diagram of an example arrangement for performing distributed event processing, according to some implementations.

[0004] Figs. 2A and 2B are flow diagrams of condition evaluation processes, according to various implementations.

[0005] Fig. 3 is a flow diagram of a correlation process, according to some implementations.

[0006] Fig. 4 is a block diagram of an example arrangement for performing distributed event processing, according to further implementations.

[0007] Fig. 5 is a block diagram of an example computing node, according to some implementations.

Detailed Description

[0008] An event analysis system can receive events from multiple sources and can perform analysis of the received events for various purposes. An event can refer to an activity represented by data collected at a source. In some examples, a

source can include a monitoring agent. A monitoring agent can be implemented as machine-readable instructions executing on an electronic device, or alternatively, a monitoring agent can be a standalone device coupled to a network or an electronic device that is being monitored.

[0009] An event can be triggered if monitored data satisfies a specified condition. The monitored data can relate to a login attempt, and the event may be triggered if the login attempt fails. Another example event is an event based on monitoring usage of a resource at an electronic device, such as a server computer, a communication switch or router, a storage subsystem, and so forth. If the monitored resource usage exceeds a specified threshold, then a corresponding event can be triggered. Data representing the event can be in the form of a message, an alert, or any other type of notification that can be sent from a monitoring agent to an event analysis system. Although specific examples are provided above, it is noted that there can be many other examples of events.

[0010] Analysis of events can be performed for one or some combination of the following purposes: to provide insight into activities within a distributed arrangement of electronic devices, to detect intrusion by an unauthorized entity (human or code), to determine compliance, to perform risk management, and so forth. Intrusion detection can be performed to identify intrusion of an unauthorized entity into a network or an electronic device. Compliance determination can be performed to determine whether an activity or data within a network or an electronic device satisfies a specified policy or government regulation. Risk management can be performed to ascertain a risk level associated with activities of an enterprise (*e.g.* a business concern, an educational organization, a government agency, etc.).

[0011] The event analysis system can employ a rules engine to determine whether a pattern of events (*e.g.* events that are within a certain time window, events that can be joined together, etc.) satisfies one or multiple rules. Traditionally, a rules engine of an event analysis system is not easily scalable to handle increasing workload (due to an increased number of events or event sources). To handle increased workload, an enterprise may have to invest in high-end hardware

equipment (*e.g.* high-end server computers), which may be expensive. Alternatively, an enterprise can implement filters to restrict the ingested input stream of events, to avoid overloading the rules engine. However, restricting the ingested input stream of events can lead to deteriorated performance of an event analysis system.

[0012] In accordance with some implementations, an event analysis system is implemented with distributed computing nodes that can be easily scalable to accommodate increased event processing workload, such as due to an increased number of events. A computing node can refer to a computer, a collection of computers, a processor, or a collection of processors. To increase the capacity of the event analysis system, scale-out data processing can be implemented using the distributed computing nodes. Scale-out data processing refers to processing that can be scaled outwardly by simply adding more computing nodes; in this manner, existing computing nodes do not have to be upgraded with higher-end equipment. Techniques or mechanisms to provide the scale-out data processing are discussed further below.

[0013] As the number or source of events to be processed is increased, the number of distributed computing nodes can be increased to handle the increased event processing workload.

[0014] As shown in Fig. 1, an event analysis system 100 receives events 102 from one or multiple event sources 104. The event sources 104 can include monitoring agents, as noted above. The rules engine of the event analysis system 100 can be implemented as machine-readable instructions that are executed in a distributed manner by multiple computing nodes (106 and 108).

[0015] The rules engine can analyze events in an input data stream for detecting interesting events or patterns, based on one or multiple rules. In some examples, a rule can include multiple parts: (1) at least one condition that is to be met, and (2) at least one action that is to be triggered if an incoming event matches the condition. Examples of conditions can include a simple condition, a join condition, a time

aggregation condition, a condition based on lookup of data lists, or any other condition.

[0016] A first event can include an event from a log of a Hypertext Transfer Protocol (HTTP) server. Another event can include an event from a log of a Domain Name System (DNS) server. Yet another event can be an event from a firewall. More generally, different events may originate from different sources. In further examples, other events can include events relating to financial activities of an enterprise, events relating to sales activities of an enterprise, events relating to human resources activities, and so forth.

[0017] Conditions specified in a rule can range in complexity. For example, a simple condition can perform a string match of an attribute (or attributes) of an event with a specified target value (or values). A specific example of a string match condition is “IPaddress==10.10.10.10,” which attempts to match an Internet Protocol (IP) address of an event with a target value (10.10.10.10). In other examples, more complex conditions can be specified, such as conditions that employ expressions, conditions based on lookup of data lists, and so forth.

[0018] Some rules can also specify correlations between events. A correlation can refer to either time aggregating events or joining events, or both. Correlating events can refer to discovering a relationship among the events to determine the significance of such relationship, so that an action can be taken based on the correlated events.

[0019] For example, a rule may specify that a condition is satisfied only if a minimum number of relevant events are detected within a specific time window. To determine if this rule is satisfied, the relevant events occurring within the specific time window are collected or aggregated (a process referred to as time aggregation).

[0020] In a specific example, an event analysis system can use a rule to identify potential attackers on a server from a stream of HTTP log events. The conditions of the rule can be as follows: "identify any sequence of 10 malformed HTTP

requests from the same IP address made within a one-minute window." The time aggregation performed for this rule would collect events relating to malformed HTTP requests from the same IP address within the one-minute window. The event analysis system can then determine whether there are at least 10 such events in the one-minute window to determine if the rule is satisfied. If the rule is satisfied, then the event analysis system can trigger an action specified in the rule. As an example, the action can be to add the IP address to a blacklist of IP addresses that are blocked from accessing a network or electronic device.

[0021] A rules engine of an event analysis system can also perform a join of events, based on a join rule. A join rule correlates different events. The different events may be generated by a single source or by multiple sources.

[0022] The evaluation of conditions in rules and the correlation of events can be distributed across the computing nodes of the event analysis system 100. The event analysis system 100 includes condition evaluation nodes 106 that can perform the evaluation of conditions in rules. The event analysis system 100 also includes correlation nodes 108 for performing the correlation of events.

[0023] In accordance with some implementations, both the condition evaluation tasks and the correlation tasks can be distributed across multiple computing nodes. In other words, a first set of computing nodes can be used to perform distributed condition evaluations, and a second set of computing nodes can be used to perform correlation. Note that the first set and the second set of computing nodes can be different sets of computing nodes, or can be a common set of computing nodes. In other words, the condition evaluation nodes 106 can be different from the correlation nodes 108, or alternatively, they can be the same computing nodes.

[0024] The events102 are received by respective condition evaluation nodes 106. Each condition evaluation node 106 includes condition evaluation module 110, which can be implemented as machine-readable instructions executable in the respective condition evaluation node 106. The condition evaluation module 110 evaluates received events against rules 112 stored in the respective condition

evaluation node 106. Each condition evaluation node 106 also stores data lists 114, which can include shared global state information to be evaluated against one or multiple rules 112. Examples of the data lists 112 include a blacklist of IP addresses, a whitelist of IP addresses, an event counter (to count a number of events detected by the event analysis system), or any data structure containing information that can be used in evaluating a condition of a rule.

[0025] At least some of the data lists 114 can include dynamic data that can change, such as in response to an action performed when a specific rule is satisfied by a received event (or events). Since evaluation of the rules considers dynamic data, such rules can be referred to dynamic rules. As discussed further below, the data lists 114 are maintained synchronized across the various nodes 106 and 108 such that the nodes 106 and 108 have access to consistent data lists.

[0026] If an event satisfies a rule 112, a condition evaluation module 110 can determine if the action triggered by the rule 112 can be performed locally at the condition evaluation node 106. If the action can be performed locally, then information of such an event would not have to be forwarded to a correlation node 108.

[0027] On the other hand, if the condition evaluation module 110 determines that an event satisfies a correlation rule (*e.g.* a time aggregation rule or a join rule), then the correlation action triggered by the correlation rule cannot be performed locally at the respective condition evaluation node 106, in which case the condition evaluation module 110 forwards information of the event to a respective correlation node 108. Note that if the event satisfies multiple rules that involve correlation (time aggregation and/or event joining), then the condition evaluation module 110 can forward information of the event to multiple respective correlation nodes 108, where each correlation node performs a respective one of the correlations specified by the multiple rules. The information of events forwarded to correlation nodes 108 are referred to as 107 in Fig. 1. The information of an event 107 forwarded to a correlation node 108 includes meta information relating to the rule(s) that was (or were) matched to the respective event. For example, the information of an event

107 can include partial match information, which includes a subset of the event's attributes combined with the identifier(s) of the rule(s) partially matched by the event. By sending just information of the event 107 rather than the entire event from a condition evaluation node 106 to a correlation node 108, more efficient usage of the communication bandwidth between the condition evaluation nodes 106 and the correlation nodes 108 is achieved.

[0028] Although not shown in Fig. 1, the correlation nodes 108 can also send messages back to the condition evaluation nodes 106. Also, condition evaluation nodes 106 can send messages to other condition evaluation nodes 106, and correlation nodes 108 can send messages to other correlation nodes 108.

[0029] Each correlation node 108 includes a correlation module 116, which can be implemented as machine-readable instructions executable in the respective correlation node 108. Each correlation node 108 also stores rules 118 and the data lists 114. The rules 118 stored at the correlation node 108 are correlation rules. In other examples, the rules stored at each correlation node 108 can be the same rules 112 stored at the condition evaluation nodes 106. Each correlation module 116 performs correlation of events forwarded from condition evaluation nodes 106, based on the rules 118 stored at the correlation node 108.

[0030] Fig. 2A is a flow diagram of a condition evaluation process that can be performed by a condition evaluation module 110, in accordance with some implementations. The condition evaluation module 110 determines (at 202) whether an event received by the condition evaluation module 110 matches a condition of a correlation rule (from among the rules 112 in Fig. 1) that relates to correlating (*e.g.* time aggregating or event joining) of events.

[0031] Note that the matching of the event to a condition of the rule can be a partial match of the event to the rule. A rule can include multiple sets of conditions. An event can be compared against each of the multiple sets of conditions in the rule. A partial match refers to a situation where the event matches less than all sets of conditions in the rule. In some cases, the multiple sets of conditions can relate to

different events. Thus, any given event may partially match just a subset of the conditions.

[0032] In response to determining that the event matches a condition of the correlation rule, the condition evaluation module 110 selects (at 204) one of the correlation nodes 108 to send information of the event (107 in Fig. 1), based on an attribute of the event and/or an identifier of the rule. Different values of the attribute (or of multiple attributes) may map to different correlation nodes 108. In some implementations, the correlation node 108 can be selected based on an identifier of the correlation rule, such that different rules would map to different correlation nodes 108. In other examples, selection of one of the correlation nodes 108 is based on both the attribute(s) of the event and the identifier of the rule.

[0033] The condition evaluation module 110 then sends (at 206) the information of the event to the selected correlation node 108 to perform correlation of the event with another event.

[0034] To enable the distributed processing of joins and aggregations, information of events that potentially can be joined or aggregated together are forwarded to the same correlation node 108. In some examples, a function (*e.g.* a hash function) is applied to one or multiple attributes of the event and/or to the identifier of the correlation rule. The function produces an output value (*e.g.* hash value). Different values output by the function cause different correlation nodes 108 to be selected.

[0035] Fig. 2B is a flow diagram of a condition evaluation process that can be performed by a condition evaluation module 110, in accordance with further implementations. The condition evaluation module 110 receives (at 220) an event. The condition evaluation module 110 determines (at 222) whether the event matches a condition of a rule (from among rules 112 in Fig. 1) associated with an action that can be performed locally at the respective condition evaluation node 106 (such rule is referred to as a “local rule” below). If the event does not match any local rule, then the condition evaluation process proceeds to task 226.

[0036] However, if the event matches a local rule, then the condition evaluation module 110 performs (at 224) an action specified by the local rule.

[0037] The condition evaluation module 110 further determines (at 226) whether the event matches a condition of a correlation rule (from among the rules 112 in Fig. 1). If the event does not match any correlation rule, then the condition evaluation process stops.

[0038] However, if the event matches a correlation rule, the condition evaluation module 110 selects (at 228) one of the correlation nodes 108 to send information of the event (107 in Fig. 1), based on an attribute(s) of the event and/or an identifier of the correlation rule. The condition evaluation module 110 generates (at 230) information of the event (107 in Fig. 1) to send to the selected correlation node 108. As noted above, the information of the event (107) can include partial match information, which includes a subset of the event's attributes combined with the identifier(s) of the rule(s) partially matched by the event.

[0039] The condition evaluation module 110 then sends (at 232) the information of the event to the selected correlation node 108 to perform correlation of the event with another event.

[0040] Note that the event can match multiple local rules and/or correlation rules, in which case tasks 222 and 224 can be repeated for each match to a respective local rule, and tasks 226, 228, and 230 can be repeated for each match to a respective correlation rule.

[0041] If an event matches multiple correlation rules, then information of the event may be forwarded to multiple correlation nodes 108 (since the function applied to the event attribute(s) and the different rule identifiers would produce multiple output values, which potentially may map to multiple correlation nodes 108). In this case, the information of the event generated for a match to a first correlation rule may differ from information of the event generated for a match to a second

correlation rule, so that different correlation nodes 108 would receive different information of the event.

[0042] Fig. 3 is a flow diagram of a correlation process that can be performed by a correlation module 116, in accordance with some implementations. The correlation module 116 receives (at 302) information of events from one or multiple condition evaluation nodes 106.

[0043] In response to the received information of events, the correlation module 116 performs (at 304) correlation of the events, which can include time aggregating and/or joining the events, as specified by the correlation rule(s) identified in the received information of events. For example, if the correlation rule is a time aggregation rule, then the correlation module 116 determines whether the events fall within a specified time interval. If so, the events can be time aggregated.

[0044] On the other hand, if the correlation rule is a join rule, then the correlation module 116 can determine if the events satisfy respective sets of conditions in the correlation rule (where each set of conditions corresponds to a respective event). If the different events satisfy the respective different sets of condition of the join rule, then the events of can be joined.

[0045] A join rule connects different events that have certain attributes in common. An example join rule can include a first set of conditions that relate to an event from an intrusion detection system, which is directed to a specific resource on a specific port. A second set of conditions of the join rule may specify an event associated with a firewall that is directed to a specific resource on a specific port. Attributes of the different events that are compared for purposes of joining the events can include any or some combination of the following: a source address, a target address, a source port, a target port, and so forth. If the values of the attributes being compared match, then the different events can be joined.

[0046] The correlation module 116 can also send (at 306) update information to nodes (condition evaluation nodes 106 and correlation nodes 108) to update one or

multiple data lists 114, in the case where the correlation performed by the correlation module 116 results in an update of dynamic data in the one or multiple data lists.

[0047] Fig. 4 is a block diagram of an example arrangement that includes an event analysis system 100-1 according to further implementations. The event analysis system 100-1 includes condition evaluation nodes 106-1 and correlation nodes 108-1. Each of the condition evaluation nodes 106-1 includes a respective condition evaluation module 110, which is similar to or the same as the condition evaluation module 110 discussed in connection with Fig. 1. Also, each of the correlation nodes 108-1 includes a correlation module 116 that is the same as or similar to the correlation module 116 discussed above in connection with Fig. 1.

[0048] Each condition evaluation module 106-1 further includes event receivers 402 for receiving events from event sources 104. The event analysis system 100-1 can include a load balancer 404 for distributing events received from the event sources 104 across the event evaluation nodes 106. The distribution of events across the condition evaluation nodes 106 can be performed to balance the workload of the condition evaluation nodes 106-1.

[0049] Each condition evaluation node 106-1 also stores rules 112 and data lists 114. If a condition evaluation module 110 determines that an event can be locally processed by the respective condition evaluation node 106-1, then the condition evaluation module 110 can trigger an action (406) corresponding to the rule that is satisfied by the event. If the action causes an update of dynamic data included in the data lists 114, then the update (408) can be sent to a state manager 410, for performing an update of one or multiple data lists 114.

[0050] If the condition evaluation module 110 determines that an event matches a correlation rule, then information of the event is forwarded to a selected one of the correlation nodes 108-1. The correlation module 116 in each correlation node 108-1 can perform correlation of events received from one or multiple condition evaluation nodes 106-1. In response to events satisfying a correlation rule (in the rules 118), the correlation module 116 can trigger an action (410). If an update of dynamic data

in the data lists 114 is to be performed, then the update (412) is sent to a state manager 414 in the correlation node 108-1, for updating one or multiple data lists 114.

[0051] Although not shown in Fig. 4, the correlation nodes 108-1 can also send messages back to the condition evaluation nodes 106-1. Also, condition evaluation nodes 106-1 can send messages to other condition evaluation nodes 106-1, and correlation nodes 108-1 can send messages to other correlation nodes 108-1.

[0052] A challenge in performing event processing using distributed computing nodes is that the rules engine of the event analysis system may maintain shared global state information (in the form of the data lists 114, for example) that is used for the event processing. The event processing can use the shared global state information to determine whether one or multiple rules are satisfied. The shared global state information is shared among the computing nodes of the event analysis system (100 or 100-1). To maintain consistency of such shared global state information across the multiple computing nodes, any update (data insertion, data deletion, or data modification) of the shared global state information results in the update being broadcast to all other computing nodes, such as by using the state managers 410 and/or 414 of Fig. 4. The broadcast can be performed atomically using a consensus protocol, which guarantees a consistent ordering event update broadcasts among the condition evaluation nodes and the correlation nodes. Using this protocol, every node applies updates in the same order, thereby ensuring strong consistency of the shared global state information even in the case of node failure.

[0053] Fig. 5 is a block diagram of an example computing node 500, which can be used for implementing a condition evaluation node 106 or 106-1, or a correlation node 108 or 108-1. The computing node 500 includes one or multiple processors 502, which can be coupled to a network interface 504 to allow the computing node 500 to communicate over a network, such as to communicate with another computing node. A processor can include a microprocessor, microcontroller,

processor module or subsystem, programmable integrated circuit, programmable gate array, or another control or computing device.

[0054] The computing node 500 also includes a non-transitory machine-readable or computer-readable storage medium (or storage media) 506, which can store machine-readable instructions 508. The machine-readable instructions 508 can include the condition evaluation module 110 or correlation module 116, in some examples. The storage medium (or storage media) 506 can also store the rules 112 or 118 and the data lists 114.

[0055] The storage medium (or storage media) 506 can be implemented with any or some combination of different forms of memory including semiconductor memory devices such as dynamic or static random access memories (DRAMs or SRAMs), non-volatile memories (*e.g.* memristor memories, phase change memories, etc.), erasable and programmable read-only memories (EPROMs), electrically erasable and programmable read-only memories (EEPROMs) and flash memories; magnetic disks such as fixed, floppy and removable disks; other magnetic media including tape; optical media such as compact disks (CDs) or digital video disks (DVDs); or other types of storage devices. Note that the instructions discussed above can be provided on one computer-readable or machine-readable storage medium, or alternatively, can be provided on multiple computer-readable or machine-readable storage media distributed in a large system having possibly plural nodes. Such computer-readable or machine-readable storage medium or media is (are) considered to be part of an article (or article of manufacture). An article or article of manufacture can refer to any manufactured single component or multiple components. The storage medium or media can be located either in the machine running the machine-readable instructions, or located at a remote site from which machine-readable instructions can be downloaded over a network for execution.

[0056] In the foregoing description, numerous details are set forth to provide an understanding of the subject disclosed herein. However, implementations may be practiced without some of these details. Other implementations may include

modifications and variations from the details discussed above. It is intended that the appended claims cover such modifications and variations.

What is claimed is:

1. A method comprising:

determining, by a system, whether an event matches a condition of a rule relating to correlating of events;

in response to determining that the event matches a condition of the rule, selecting, by the system, a given one of a plurality of computing nodes to send information of the event, based on one or both of an attribute of the event and an identifier of the rule; and

sending, by the system, the information of the event to the given computing node to perform correlation of the event with another event.

2. The method of claim 1, wherein determining whether the event matches the condition of the rule comprises evaluating the event with respect to the rule using dynamically changing data.

3. The method of claim 2, further comprising:

updating the dynamically changing data in response to an action performed as a result of the correlation of the event with another event.

4. The method of claim 3, wherein the determining, the selecting, and the sending are performed by a first one of the plurality of computing nodes, the method further comprising:

determining whether the event matches a condition of a second rule that specifies a local action to be performed at the first computing node;

in response to determining that event matches a condition of the second rule, performing the local action at the first computing node; and

updating the dynamically changing data in response to the action performed at the first computing node.

5. The method of claim 1, wherein the rule relates to joining of different events.

6. The method of claim 1, wherein the rule relates to aggregating events within a specified time interval.

7. The method of claim 1, wherein the determining, the selecting, and the sending are performed by a first computing node of the system, the method further comprising:

determining, by a second computing node, whether a second event matches a condition of the rule;

in response to determining that the second event matches a condition of the rule, selecting, by the second computing node, the given computing node to send information of the second event, based on one or both of an attribute of the second event and the identifier of the rule; and

sending, by the second computing node, the information of the second event to the given computing node to perform correlation of the second event and the event sent by the first computing node.

8. The method of claim 1, further comprising:

determining, by the system, whether the event matches a condition of a second rule relating to correlating of events;

in response to determining that the event matches a condition of the second rule, selecting, by the system, another one of the plurality of computing nodes based on one or both of an attribute of the event and an identifier of the second rule; and

sending, by the system, information of the event to the another computing node to perform correlation of the event with another event according to the second rule.

9. A system comprising:

a plurality of first computing nodes to receive events and to evaluate the events with respect to a rule; and

a plurality of second computing nodes to correlate events,

wherein a given one of the plurality of first computing nodes is to:

determine that the events received by the given first computing node match a condition of the rule;

in response to the determining, select one of the plurality of second computing nodes, based on one or both of an attribute of the events received by the given first computing node and an identifier of the rule; and

send the events to the selected second computing node to perform correlation of the sent events according to the rule.

10. The system of claim 9, wherein the determining that the events received by the given first computing node match the condition of the rule is a determination that the events received by the given first computing node partially satisfy the rule.

11. The system of claim 9, wherein different values associated with one or both of the attribute and the identifier of the rule map to different ones of the second computing nodes.

12. The system of claim 9, wherein the given first computing node is to perform the selecting by applying a function on one or both of the attribute of the events received by the given first computing node and the identifier of the rule, the applied function producing a value that maps to the one of the plurality of second computing nodes.

13. The system of claim 9, wherein the aggregation comprises a join of different events according to the rule.

14. An article comprising at least one non-transitory machine-readable storage medium storing instructions that upon execution cause a system to:

determine whether events match a condition of a first rule relating to correlating of events;

in response to determining that the events match a condition of the first rule, select a first one of a plurality of computing nodes to send information of the events, based on one or both of an attribute of the events and an identifier of the first rule;

send the information of the events to the first computing node to perform correlation of the events according to the first rule;

determine whether the events match a condition of a second rule relating to correlating of events;

in response to determining that the events match a condition of the second rule, select a second one of a plurality of computing nodes, based on one or both of an attribute of the events and an identifier of the second rule; and

send information of the events to the second computing node to perform correlation of the events according to the second rule.

15. The article of claim 14, wherein the first rule includes a first condition relating to a first event, and a second condition relating to a second event, and wherein the correlation according to the first rule comprises joining the first and second events based on the first condition and the second condition being satisfied.

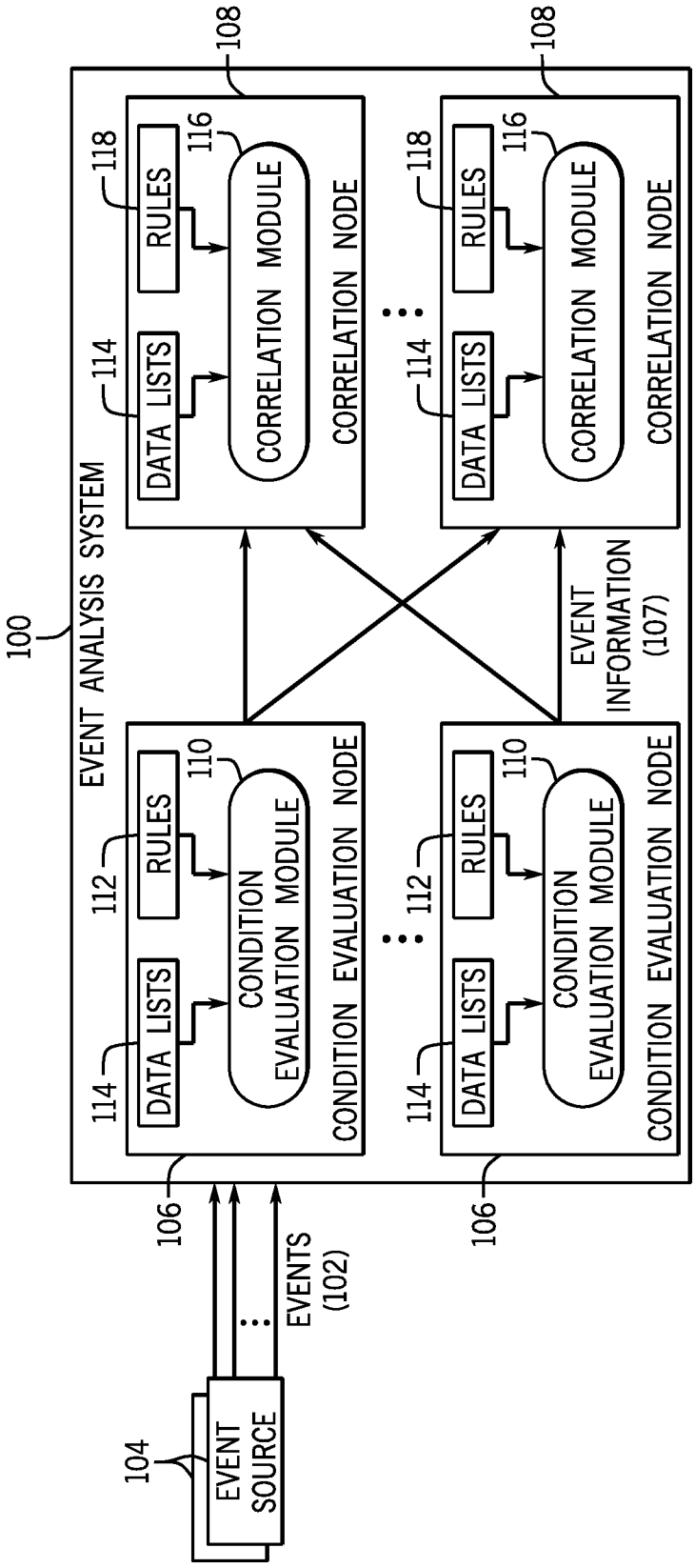
100
EVENT ANALYSIS SYSTEM
104
EVENT SOURCE
EVENTS (102)
106
106
CONDITION EVALUATION NODE
CONDITION EVALUATION NODE
112
RULES
110
CONDITION EVALUATION MODULE
114
DATA LISTS
EVENT INFORMATION (107)
108
108
CORRELATION NODE
CORRELATION NODE
118
RULES
116
CORRELATION MODULE
114
DATA LISTS

FIG. 1

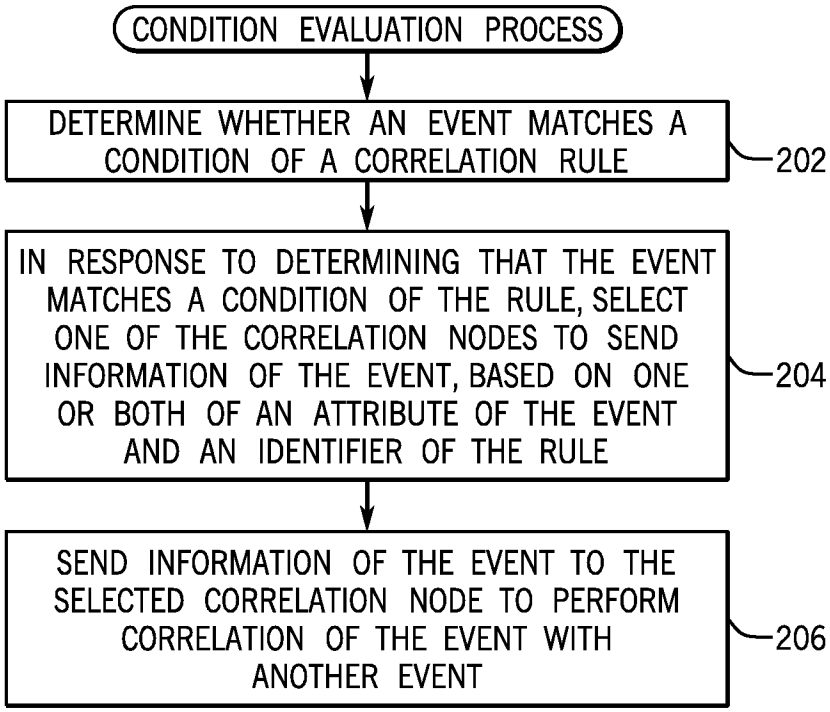
CONDITION EVALUATION PROCESS
DETERMINE WHETHER AN EVENT MATCHES A CONDITION OF A CORRELATION RULE
202
IN RESPONSE TO DETERMINING THAT THE EVENT MATCHES A CONDITION OF THE RULE, SELECT ONE OF THE CORRELATION NODES TO SEND INFORMATION OF THE EVENT, BASED ON ONE OR BOTH OF AN ATTRIBUTE OF THE EVENT AND AN IDENTIFIER OF THE RULE
204
SEND INFORMATION OF THE EVENT TO THE SELECTED CORRELATION NODE TO PERFORM CORRELATION OF THE EVENT WITH ANOTHER EVENT
206

FIG. 2A

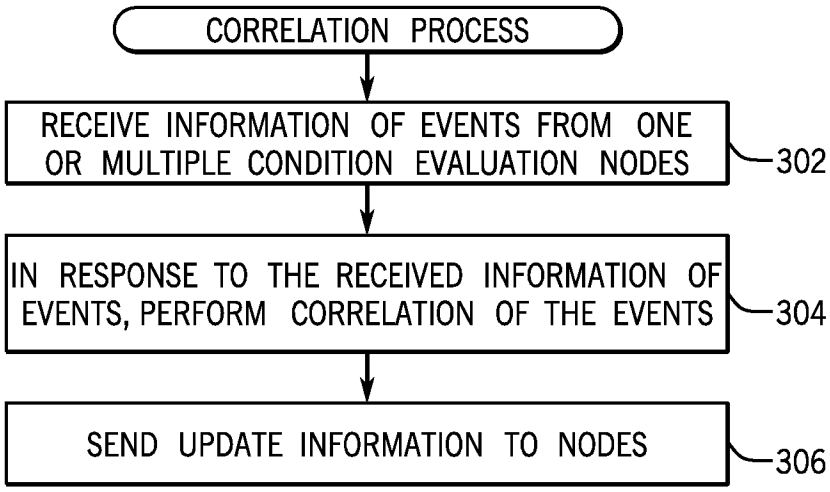
CORRELATION PROCESS
RECEIVE INFORMATION OF EVENTS FROM ONE OR MULTIPLE CONDITION EVALUATION NODES
302
IN RESPONSE TO THE RECEIVED INFORMATION OF EVENTS, PERFORM CORRELATION OF THE EVENTS
304
SEND UPDATE INFORMATION TO NODES
306

FIG. 3

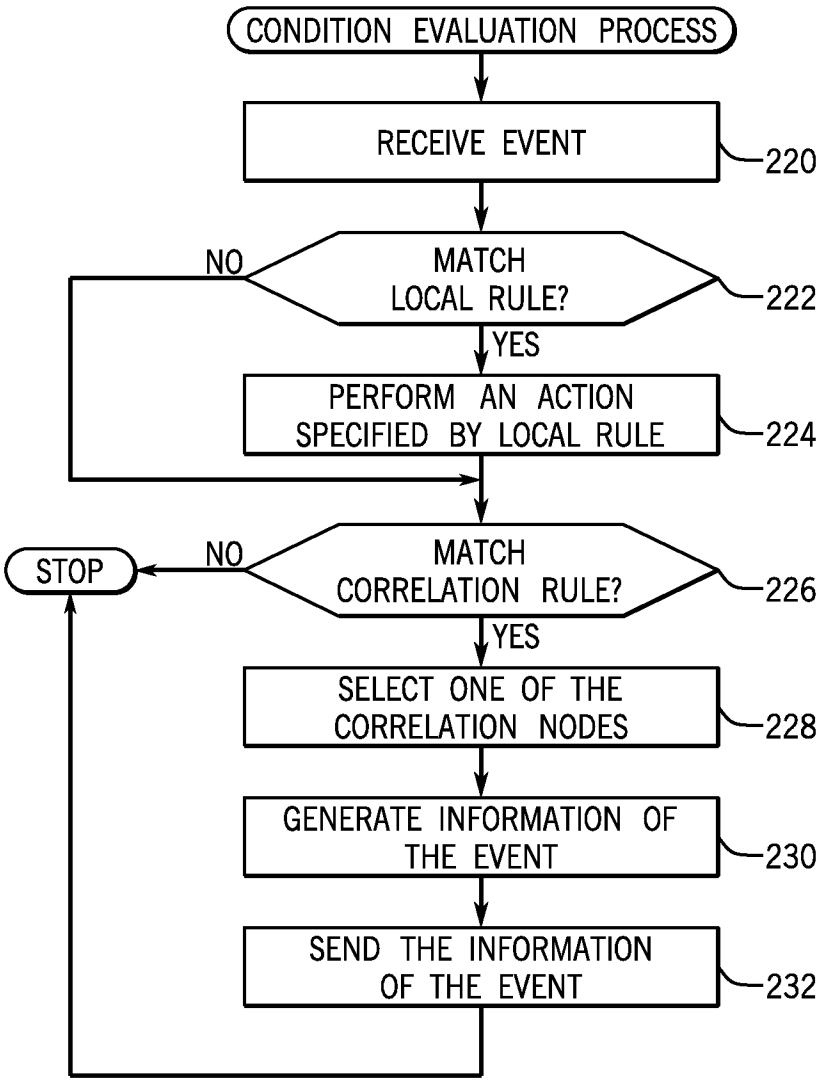
CONDITION EVALUATION PROCESS
RECEIVE EVENT
220
NO
MATCH LOCAL RULE?
222
YES
PERFORM AN ACTION SPECIFIED BY LOCAL RULE
224
STOP
NO
MATCH CORRELATION RULE?
226
YES
SELECT ONE OF THE CORRELATION NODES
228
GENERATE INFORMATION OF THE EVENT
230
SEND THE INFORMATION OF THE EVENT
232

FIG. 2B

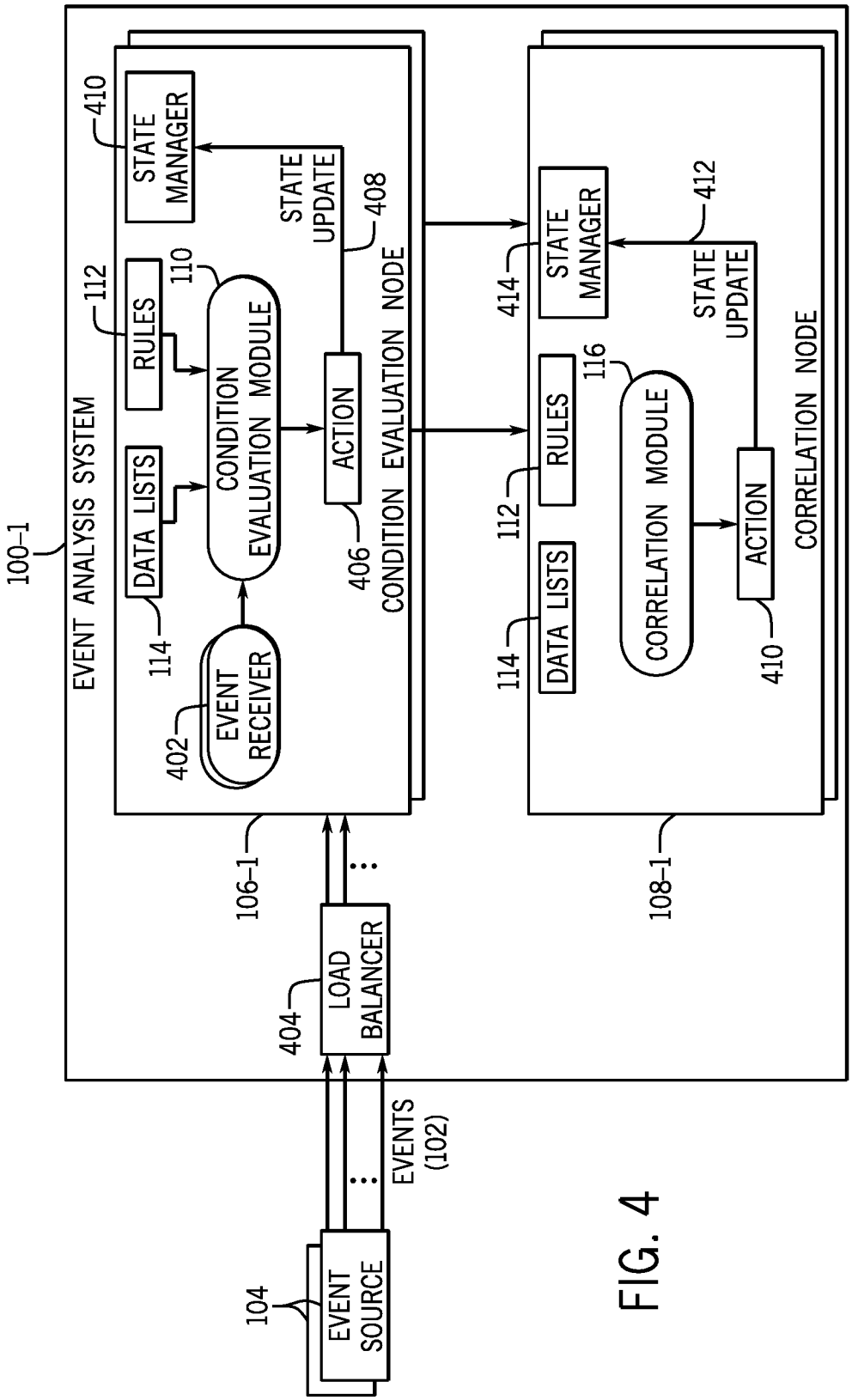
EVENT ANALYSIS SYSTEM
100-1
CONDITION EVALUATION NODE
406
STATE MANAGER
410
STATE UPDATE
408
RULES
112
CONDITION EVALUATION MODULE
110
DATA LISTS
114
EVENT RECEIVER
402
ACTION
106-1
LOAD BALANCER
404
EVENT SOURCE
104
EVENTS (102)
CORRELATION NODE
108-1
STATE MANAGER
414
STATE UPDATE
412
RULES
112
CORRELATION MODULE
116
DATA LISTS
114
ACTION
410

FIG. 4

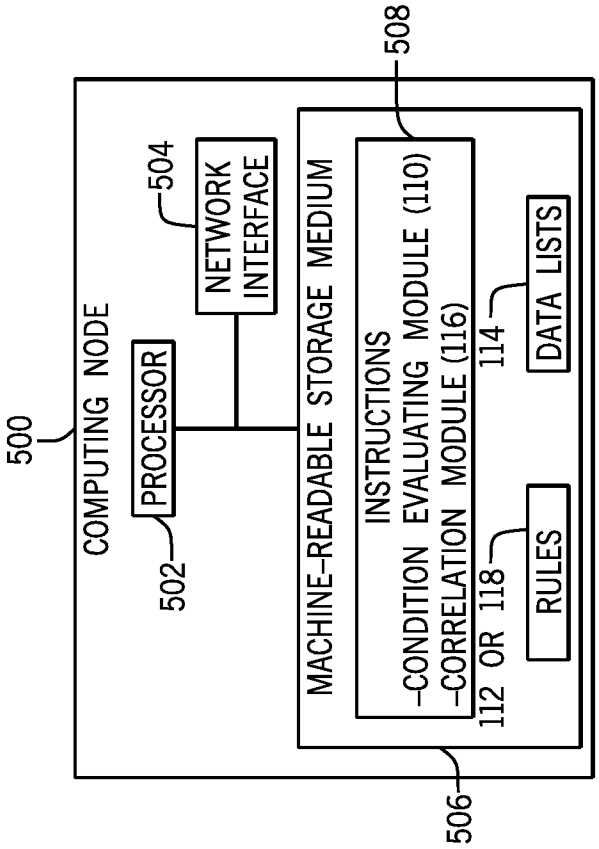
500
COMPUTING NODE
PROCESSOR
502
NETWORK INTERFACE
504
MACHINE-READABLE STORAGE MEDIUM
508
INSTRUCTIONS
-CONDITION EVALUATING MODULE (110)
-CORRELATION MODULE (116)
112 OR 118
RULES
114
DATA LISTS
506

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/036055

A. CLASSIFICATION OF SUBJECT MATTER

G06F 17/00(2006.01)i

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 17/00; G06F 15/173; G06F 3/00; G06F 9/44; G06F 17/30; G06F 9/46; G06F 11/20; G06F 21/00

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: event, correlation, network management, and similar terms

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2010-0223628 A1 (JOSHUA H. ROSENBLUTH et al.) 02 September 2010 See paragraphs [0072] and [0109]; claim 1; and figure 2.	1-15
A	US 6,061,723 A (WALKER ANTHONY et al.) 09 May 2000 See column 10, lines 21-25 and 39-42; column 13, lines 29-40; and figures 4 and 6.	1-15
A	WO 2000-039674 A1 (COMPUTER ASSOCIATES THINK, INC.) 06 July 2000 See page 2, lines 4-24; page 14, lines 20-24; and figure 3.	1-15
A	US 2011-0047262 A1 (DANIEL JOSEPH MARTIN et al.) 24 February 2011 See paragraphs [0007] and [0032]-[0034]; and figure 3.	1-15
A	US 2008-0244741 A1 (ERIC GUSTAFSON et al.) 02 October 2008 See paragraphs [0007] and [0051]-[0052]; and figure 6.	1-15

☐ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
19 January 2015 (19.01.2015)	**20 January 2015 (20.01.2015)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 302-701,
Republic of Korea
Facsimile No. ++82 42 472 3473

Authorized officer
NHO, Ji Myong
Telephone No. +82-42-481-8528

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/US2014/036055

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2010-0223628 A1	02/09/2010	US 7730494 B1	01/06/2010
		US 8307374 B2	06/11/2012
US 06061723 A	09/05/2000	EP 0909056 A2	14/04/1999
		EP 0909056 A3	22/09/1999
		EP 0909056 B1	28/02/2007
		JP 11-184781 A	09/07/1999
		JP 3556842 B2	25/08/2004
WO 00-39674 A1	06/07/2000	AT 363095 T	15/06/2007
		AU 2395200 A	31/07/2000
		AU 760999 B2	29/05/2003
		BR 9916697 A	25/09/2001
		CA 2356672 A1	06/07/2000
		CN 1126033 C	29/10/2003
		CN 1332867 A	23/01/2002
		DE 69936152 D1	05/07/2007
		DE 69936152 T2	24/01/2008
		EP 1192535 A1	03/04/2002
		EP 1192535 A4	11/06/2003
		EP 1192535 B1	23/05/2007
		IL 143940 A	03/06/2007
		JP 2002-533828 A	08/10/2002
		KR 10-0512231 B1	05/09/2005
		US 6446136 B1	03/09/2002
		ZA 200105265 A	26/11/2002
US 2011-0047262 A1	24/02/2011	TW 201132049 A	16/09/2011
		US 8255525 B2	28/08/2012
		WO 2011-020765 A1	24/02/2011
US 2008-0244741 A1	02/10/2008	CA 2629723 A1	24/05/2007
		EP 1949235 A2	30/07/2008
		EP 1949235 A4	03/09/2014
		JP 2009-516266 A	16/04/2009
		US 8046833 B2	25/10/2011
		WO 2007-058952 A2	24/05/2007
		WO 2007-058952 A3	07/05/2009