



- (51) International Patent Classification:
H04L 29/08 (2006.01)
- (21) International Application Number:
PCT/US2015/026064
- (22) International Filing Date:
16 April 2015 (16.04.2015)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
6750/CHE/2014 30 December 2014 (30.12.2014) IN
- (71) Applicant: **HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP** [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).
- (72) Inventors: **TIWARI, Hari Sahaya**; Sy. No. 192, Whitefield Road, Mahadevapura Post, Bangalore 560048 (IN). **KULKARNI, Manish Ramesh**; Sy. No. 192, Whitefield Road, Mahadevapura Post, Bangalore 560048 (IN). **NAIDU, Bhakthavatsala K.**; Sy. No. 192, Whitefield Road, Mahadevapura Post, Bangalore 560048 (IN). **ERTL, Markus**; Alttrottstrasse 31, 69190 Walldorf (DE).
- (74) Agents: **SHOOKMAN, Jeb A.** et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road, Mail Stop 79, Fort Collins, CO 80528 (US).

[Continued on next page]

(54) Title: CLUSTER ARBITRATION

(57) Abstract: Approaches for arbitrating between node groups within a cluster are described. In one example, non-transitory machine-readable storage medium encoded with instructions executable by a processing resource is to analyze a received message associated with a node group from among a plurality of node groups based on at least one configurable policy. The message may include information associated with operation of the node group. Based on the analyzing of the received message a control indication is generated for use in controlling operations of one or more nodes.

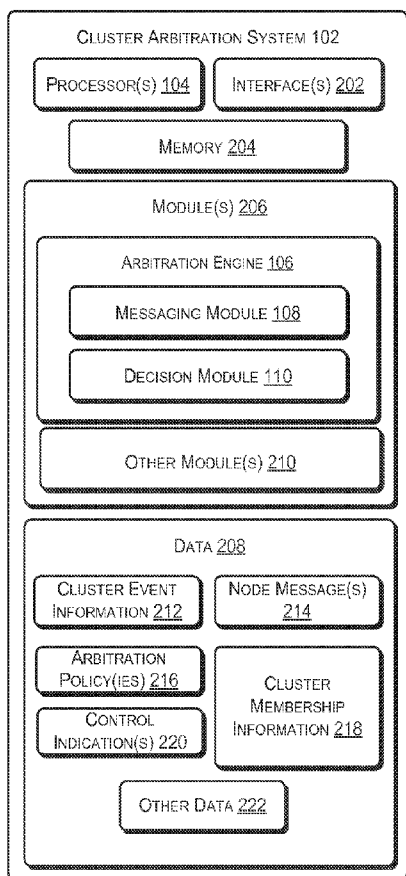


FIG. 2



(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).

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))

CLUSTER ARBITRATION

BACKGROUND

[0001] Multiple applications may be hosted on an application server. Generally, such applications may be supported by a group of computing devices or servers, collectively referred to as cluster. Clustering allows multiple servers to run the same application. Such architectures allow the applications to be handled seamlessly and efficiently between the groups of computing devices. In case one of the computing devices hosting the application is down, the application workload may be switched or moved to another computing device within the cluster. Each of the nodes may further be coupled to data storage, for accessing or creating data by the applications.

[0002] Each computing device of a cluster may be considered as a node. While existing as a member within the cluster, each node is able to communicate with other nodes within the cluster. In some cases, for example, due to a node failure or due to issues with a communication channel between the nodes, the communication between the nodes may go down. In such cases, one or more groups of nodes may be gracefully shut down to avoid data access by different instances of application executing on remaining node groups.

BRIEF DESCRIPTION OF THE DRAWINGS

[0003] The following detailed description references the drawings, wherein:

[0004] FIG. 1A is a block diagram of an example system for arbitration of a cluster;

[0005] FIG. 1B is a communication network environment implementing an example system for cluster arbitration;

[0006] FIG. 2 is a block diagram of another example system for cluster arbitration;

[0007] FIGS. 3A-3B are communication network environment implementing an example system for cluster arbitration;

[0008] FIG. 4 is a flowchart of an example method for arbitration between node groups of a cluster;

[0009] FIG. 5 is a flowchart of another example method for arbitration between node groups of a cluster; and

[0010] FIG. 6 is a block diagram of an example network environment implementing a non-transitory computer-readable medium, for arbitration between node groups of a cluster.

DETAILED DESCRIPTION

[0011] A plurality of computing devices within a networked computing environment may be logically organized within a group, referred to as a cluster. The computing devices may be further coupled to a data storage device. The data storage may include data, either utilized or created, by the applications running on the computing devices. Although the applications may be executing over a definite number of computing devices, a redundant set of computing devices may also be provided for failovers. During a failover, a running application may be switched onto the redundant set of computing device to ensure that the applications are available for use with minimum downtime.

[0012] The cluster may include a plurality of computing devices, each acting as a node. The nodes may be implemented as any computing-based device, such as servers, desktops, etc. Nodes may also be implemented as virtual machines or as computational resources within a cloud-computing based environment. Within the cluster, one or more nodes may be logically organized to form node groups. In some cases, the cluster may span across a geographic region, as a result of which the node groups (and in turn the nodes) may be physically separated by a distance. In such instances, the nodes groups may be in communication with other node groups through a communication channel. Through the communication channel, the nodes and the node groups may coordinate with each other, for affecting read or write, or both operations through the applications onto the data storage.

[0013] The clusters may experience certain cluster-related events which may affect its operation, and may also affect the applications which may be running on the nodes within the clusters. Generally, during a cluster-related event, one or more nodes may no longer be in communication with the other nodes of the

cluster. For example, a communication link between two node groups may become temporarily unavailable or may be down. As a result, node groups which were previously connected to other node groups may no longer be in communication with each other, resulting into a condition referred to as a partitioned network. In case of a partitioned network, instances of applications executing on two or more disparate node groups may separately attempt to read or write data or both from the data storage. This may result in data inconsistency.

[0014] Generally, on occurrence of a partitioned network, one of the node groups may be retained as part of the cluster, whereas one or more node groups which have been affected as a result of the partitioned network may be gracefully shut down and removed from the cluster. Prior to the graceful shutdown, the application workloads may be transferred to the node groups which have been retained, to ensure the high-availability and continuity of the applications being hosted by the cluster.

[0015] Determining which of the node groups is to be gracefully shut down, involves analyzing a message received from the node groups. For example, node group from which the message was received first may be retained, whereas another node group may be allowed to be gracefully shut down. However, such determinations do not consider the computational capabilities of the node groups, nor may consider the criticality or characteristics of workloads which may be executing on the node groups. In such a case, node groups which may be handling critical workloads may be forced to gracefully shut down owing to the first-come, first-served approach. However, reinitiating or restarting the critical workloads and corresponding application onto another node group may involve considerable time during which the application would be unavailable. This may in turn affect the operation and the efficiency of the cluster, and also the availability of the application to users.

[0016] Approaches for arbitrating between node groups within a cluster are described. While arbitrating, upon occurrence of a cluster-related event, one or more node groups within a cluster may be determined to be retained within the cluster, while another set of node groups may be determined to be gracefully

shut down. In one example, a cluster may include a plurality of node groups. Each of the node groups may be implemented at different sites across different geographic locations. The node groups in turn may include one or more nodes which are in communication with each other. The different node groups may also be in communication with each other over a communication network. As a result, each of the nodes of every node group within the cluster would be in communication, either directly or indirectly, with every other node within the cluster.

[0017] Approaches for arbitrating between node groups within the cluster may include determination of an occurrence of a cluster-related event. A cluster-related event may include, but is not limited to, a partitioned network or a node failure. Any cluster-related event may also be considered as an event which affects the communication between the nodes within the cluster. For example, on occurrence of a partitioned network, the communication link between two node groups may be affected. In such a case, not all nodes within the cluster would be able to communicate with at least one other node within the cluster.

[0018] For the node groups which have been affected by the cluster-related event, one coordinator node may be selected for each corresponding node group. The coordinator node for any given node group may further obtain node-related information from each of the nodes within the node group under consideration. The node-related information may include, but is not limited to computational resources, state of the nodes, state of executing applications, and analytical information depicting historical, operational and functional data.

[0019] The coordinator node may communicate the node-related information of each node within its given node group to an arbitration engine. The arbitration engine on receiving the node-related information determines which of the node groups within the cluster are to be retained and for which node groups a graceful shutdown is to be initiated. The arbitration engine may analyze the node-related information based on one or more configurable policies. The policies therefore may be considered as the basis on which one or more node groups may be selected for gracefully shutting down, within the cluster. In an example, the arbitration engine may generate a control indication based on the

analysis of the node-related information. The control indication may indicate whether the given node group is to be retained within the cluster or is to be shut down. The control indication may then be communicated to the different node groups within the cluster.

[0020] Once determined, shutdown process for the determined node groups may be initiated. On shutting down, the determined node groups may be removed from the cluster, and workload of each of the removed groups may be shifted to the node groups which are retained pursuant to the cluster-related event.

[0021] The above approaches may also be implemented within a distributed computing environment employing a plurality of clusters. Each cluster may include a plurality of node groups, with each of the node groups further including one or more nodes. The arbitration engine may be implemented on any computing device, such as a server, in communication with each of the node groups within the cluster. Nodes may also be implemented as virtual machines or as computational entities within a cloud-based computing environment.

[0022] The present subject matter provides a managed approach for controlling the operations of the node groups within the cluster, in the event of a cluster-related event. As explained in the preceding paragraphs, the selection of the candidate node groups for shutting down is based on predefined configurable policies. Further, as per the present subject matter, analytical information corresponding to historical operational data of the node and applications executing on the node, are also considered to determine which of the node groups are to be retained and which are to be gracefully shut down. Accordingly, the process by which the failover is to be managed may be controlled and guided to better suit any specific need. The present subject matter ensures that node groups handling critical workloads are not shut down but retained, thereby resulting in better efficiency of the cluster and higher availability of the hosted applications. Moreover, waiting time periods for initiating applications within highly complex distributed environment may also be avoided based on such configurable policies.

[0023] These and other examples are further described herein with reference

to FIGS. 1-6. It is to be noted that the description and figures relate to example implementations, and is not to be construed as a limitation to the present subject matter. It is thus understood that various arrangements may be devised that, although not explicitly described or shown herein, embody the principles of the present subject matter. Moreover, all statements herein reciting principles, aspects, and embodiments of the present subject matter, as well as specific examples, are intended to encompass equivalents thereof.

[0024] FIG. 1A provides a block diagram of an example cluster arbitration system 102 for arbitrating between node groups within a cluster. The cluster arbitration system 102 may be implemented, for example, within a distributed computing environment having one or more clusters. Each of the clusters may include a plurality of node groups as its member. The node groups may in turn further include at least one node. In the present example as illustrated in FIG. 1A, the cluster arbitration system 102 may further include processor(s) 104, an arbitration engine 106. The arbitration engine 106 further includes a messaging module 108 and a decision module 110.

[0025] The processor(s) 104 may be implemented as microprocessors, microcomputers, microcontrollers, digital signal processors, central processing units, state machines, logic circuitries, and/or any devices that manipulate signals based on operational instructions. The functions of various elements shown in figures, including a functional block labelled as "processor(s)", may be provided through the use of dedicated hardware as well as hardware capable of executing machine readable instructions. The cluster arbitration system 102 may implement arbitration between the node groups within a cluster. Arbitration between the node groups is carried out on occurrence of a cluster-related event. In one example, on occurrence of a cluster-related event, the messaging module 108 may receive information associated with nodes within a node group.

[0026] The information associated with the nodes, once received, is analyzed by the decision module 110. The decision module 110 within the arbitration engine 106 may analyze the information associated with each of the nodes based on one or more configurable policies. Based on the configurable policies, the decision module 110 may determine one or more node groups within the

cluster which would be retained within the cluster. Accordingly, the other remaining node groups within the clusters would be removed as members of the cluster, and would be gracefully shut down. The application workloads may be moved to the node groups which were retained to provide for seamless availability of the applications, being hosted by the cluster.

[0027] The aspects as outlined above are further described in conjunction with FIG. 1B. FIG. 1B illustrates a network environment 100 for arbitrating between node groups within a cluster, as per an example of the present subject matter. The network environment 100 may be a cluster, or portion thereof. The cluster may further include a plurality of node groups. Within the network environment 100, the cluster arbitration system 102 may be used for arbitrating between node groups within a cluster. In the present example, the cluster arbitration system 102 may further include an arbitration engine 106 for arbitrating between node groups, such as node groups 112-1, 2. Each of the node groups 112-1, 2 further includes a plurality of nodes. In the present example, node group 112-1 includes nodes 114-1, 2,..., N (collectively referred to as nodes 114). Similarly, the node group 112-2 includes nodes 116-1,2,...,N (collectively referred to as nodes 116). Although FIG. 1B depicts only two node groups 112-1, 2, the cluster within the environment may also include other node groups 112 without deviating from the scope of the present subject matter.

[0028] The node groups 112-1, 2 may be further in communication with each other through a networked communication channel 118. The channel 114 may be implemented as any computing network allowing the node groups 112-1, 2 to communicate with each other. Once the node groups 112-1, 2 are communicatively connected, each of the nodes 114, 116 are in communication with each other as well. The nodes 114, 116, when in communication, can synchronize their operations to affect read and/or write operations onto a data storage (not shown in FIG. 1B). The nodes 114, 116 may be implemented as one or more computing-based devices, such as a server.

[0029] The nodes 114, 116 provide for failover in the event of a cluster-related event. A cluster-related event may be considered as any event as a result of which any of nodes 114 within the node group 112-1 are no longer in

communication with any one or more nodes 116 of the node group 112-2. The communication between the nodes 114, 116 of the respective node groups 112-1, 2 may be affected by a variety of conditions. For example, the network communication channel 118 during the course of operations may become temporarily unavailable, or may have gone down. This may result in a split between the node group 112-1 and the node group 112-2, a condition referred to as a partitioned network. In such a scenario, the connectivity between the node group 112-1 and the node group 112-2 would be lost. As a result, none of the nodes 114 within the node group 112-1 would be in communication with any of the nodes 116 within the node group 112-2.

[0030] Within the environment 100, the cluster arbitration system 102 may determine the occurrence of such a cluster-related event. For each of the node groups 112-1, 2, one node from among the nodes 114 is selected as a coordinator node. For example, within the node group 112-1, a node 114-2 may be elected as a coordinator node. Similarly, for the node group 112-2, node 116-2 may be elected as the coordinator node. For ease of explanation, the coordinator node elected within node group 112-1 and the node group 112-2, are referred as the coordinator node 114-2 and coordinator node 116-2, respectively.

[0031] Within the node group 112-1 and the node group 112-2, the respective coordinator nodes, i.e., coordinator node 114-2 and coordinator node 116-2, obtain node-related information from each of the nodes within the node groups 112-1, 2. The node-related information may pertain to functional and operational attributes of the nodes from which the information is obtained. The information may indicate computational resources, state of the nodes, state of executing applications, and analytical information depicting historical operational and functional data. It should be noted that the information may be used for generally representing the computational and network related capabilities of the respective nodes.

[0032] Besides this, the information may also indicate the previous operational occurrence corresponding to the respective nodes. For example, the analytical information may provide the operational history of node, such as

number of failure encountered in the past, description of faults experienced, and so on. It should be noted that the types of node-related information are exemplary, and in no manner should be construed as limiting the scope of the present subject matter. Returning to the coordinator node, the coordinator node for each node group obtains the node-related information for the corresponding node. For example, for node group 112-1 the coordinator node 114-2 obtains the node-related information from each of the nodes 114. Similarly, the coordinator node 116-2 obtains the node-related information from each of the nodes 116.

[0033] The node-related information is subsequently communicated to the cluster arbitration system 102. The node-related information received from the node groups 112-1, 2 is then analyzed and processed by the arbitration engine 106. In one example, the node-related information may be analyzed by the decision module 110. The arbitration engine 106 may process the node-related information based on one or more predefined policies or rules. In one example, the policies may provide the manner in which the node groups within a cluster are to be managed. The policies may specify certain system requirements which are to be conformed with, based on which the arbitration engine 106 may determine which of the node groups 112-1, 2 is to be shut down and which is to be retained as part of the cluster.

[0034] To manage the operation of the node groups 112-1, 2, the arbitration engine 106 may further generate a control indication for each of the node groups 112-1, 2. The control indication may subsequently be communicated to the node groups 112-1, 2. In one example, the control indication may be a message for each of the node groups 112-1, 2 to indicate whether any node group is to be retained or is to be removed from the cluster and gracefully shut down. The operation of the node groups 112-1, 2 may then be subsequently carried out in accordance with the control indication. For example, if the control indication for node group 112-2 indicates that it is to be removed from the cluster, the graceful shutdown of the node group 112-2 may be initiated. Similarly, the arbitration engine 106 may also determine that node group 112-1 is to be retained. In the present example, the initiation of the node shutdown may be performed by other computing devices (not shown in FIG. 1B), such as

a node manager. Similarly, a load balancing server (also not shown in FIG. 1B) may transfer the load from the node group 112-2 to any other node group. In one example, the load balancing may be implemented by any one or more nodes 114, 116 within the node groups 112-1, 2.

[0035] Although the present example has been explained in the context of a partitioned network, arbitrating between different node groups in the event of any other cluster-related event would also be within the scope of the present subject matter. Furthermore, the node groups 112-1, 2 have been indicated as having equal number of nodes 114, 116. This is only indicative and is not to be considered as a limitation onto the present subject matter.

[0036] It should be noted that the present explanation has been provided with respect to only two node groups 112-1, 2 interacting with each other. The same subject matter can also be implemented using other multiple node groups, each of which may be communicatively connected to the cluster arbitration system 102, without deviating from the scope of the present subject matter. These and other aspects of the present subject matter are provided with additional details in conjunction with the description for FIGS. 2-3.

[0037] FIG. 2 illustrates an example cluster arbitration system 102 for arbitrating between node groups within a cluster. The cluster arbitration system 102 may be implemented as, but is not limited to, a server, a workstation, a computer, and the like. The cluster arbitration system 102 further includes interface(s) 202 and memory 204. The interface(s) 202 may include a variety of interfaces, for example, interfaces for data input and output devices, referred to as I/O devices, storage devices, network devices, and the like. The interface(s) 202 facilitate communication between the system 102 and various computing devices connected in the network environment 100, including, but not limited to, nodes 114, 116 within the node groups 112-1, 2.

[0038] The interface(s) 202 may include a variety of interfaces, for example, interfaces for data input and output devices, referred to as I/O devices, storage devices, network devices, and the like, for communicatively associating the cluster arbitration system 102 either one or more nodes groups, such as node groups 112-1, 2. The interface(s) 202 may be implemented as either hardware

or software. The memory 204 may store one or more executable instructions, which may be fetched and executed so as to implement the arbitration between node groups within a cluster. An example of such an instruction includes, but is not limited to, firmware for the cluster arbitration system 102 for implementing multiple functionalities. The memory 204 may be non-transitory computer-readable medium including, for example, volatile memory, such as RAM, or non-volatile memory such as EPROM, and flash memory.

[0039] The cluster arbitration system 102 may include module(s) 206. The module(s) 206, amongst other things, include routines, programs, objects, components, and data structures, which perform particular tasks or implement particular abstract data types for providing access control for objects. The module(s) 206 may be implemented by hardware, by computer-readable instructions executed by a processing unit, or by a combination thereof. In one implementation, the module(s) 206 includes the arbitration engine 106. The arbitration engine 106 includes a messaging module 108 and the decision module 110.

[0040] The cluster arbitration system 102 may further include other module(s) 206 for implementing functionalities that supplement applications or functions performed by the cluster arbitration system 102. Amongst other things, the data 208 includes cluster event information 212, node message(s) 214, arbitration policies 216, cluster membership information 218, control indication 220 and other data 222. The module(s) 206 may be implemented as a combination of hardware and programming (e.g., programmable instructions) to implement one or more functionalities of the module(s) 206.

[0041] In examples described herein, such combinations of hardware and programming may be implemented in a number of different ways. For example, the programming for the module(s) 206 may be processor executable instructions stored on a non-transitory machine-readable storage medium and the hardware for the module(s) 206 may include a processing resource (e.g., one or more processors), to execute such instructions. In the present examples, the machine-readable storage medium may store instructions that, when executed by the processing resource, implement module(s) 206 or their

associated functionalities. In such examples, the cluster arbitration system 102 may include the machine-readable storage medium storing the instructions and the processing resource to execute the instructions, or the machine-readable storage medium may be separate but accessible to cluster arbitration system 102 and the processing resource. In other examples, module(s) 206 may be implemented by electronic circuitry.

[0042] In operation, the cluster arbitration system 102 may determine whether a cluster-related event has occurred. In one example, the cluster arbitration system 102 may receive an indication which depicts that a cluster-related event has occurred. As also explained in the preceding portions, each of the nodes within the node groups 112-1, 2 are in communication (direct or indirect) with each other.

[0043] It is to be noted that as a result of the cluster-related event, one or more nodes within node groups (such as node groups 112-1, 2) may no longer be in communication with any other node, either within the same node group, or any other node group. Examples of a cluster-related event may include, but are not limited to, a partitioning of a network and node failure. Any other type of cluster-related event would also be within the scope of the present subject matter.

[0044] On the occurrence of the cluster-related event, any one from amongst each of the nodes 114, 116 may be elected as a coordinator node. To elect a coordinator node, each of the nodes 114, 116 within the node groups 112-1, 2 may be used for electing a coordinator node within the respective node groups 112-1, 2. In the present example (such as that illustrated in FIG. 1B), node 114-2 for node group 112-1, and node 116-2 for node group 112-2 may be considered as being the coordinator nodes, respectively. For the purposes of the following description, the coordinator nodes would be referred to as coordinator node 114-2 and coordinator node 116-2.

[0045] The coordinator nodes, i.e., coordinator node 114-2 and coordinator node 116-2, may further obtain information of each of the nodes within their respective node groups. For example, the coordinator node 114-2 may poll each of the nodes present within the node group 112-1 for node related

information. The node-related information, in one example, may indicate the functional and operational characteristics of the node. The node-related information may also include analytical information indicating historical operational details of the respective nodes. As an example, the historical operational details may provide information associated with node failure in predetermined time intervals, number of times the node experienced downtime, etc. In another example, the node-related information may indicate computational resources, state of executing applications, criticality or characteristics of workloads, and complexity of workloads. The different types of node-related information are indicative, and are not intended for defining the scope of the present subject matter.

[0046] For the node group 112-1, the node-related information is gathered by the coordinator node 114-2 and communicated to the cluster arbitration system 102. In a similar manner, node-related information of the node group 112-2 is gathered by the coordinator node 116-2 and communicated to the cluster arbitration system 102. In one example, the node-related information may be communicated as messages to the cluster arbitration system 102. The node-related information provided by the respective coordinator node 114-2 and the coordinator node 116-2, are received by the messaging module 108 in the arbitration engine 106.

[0047] Once received by the cluster arbitration system 102, the messaging module 108 may store the node-related information from the different node groups 112-1, 2 in the form of node message(s) 214. In one example, the node message(s) 214 may be stored as a mapping between a node identifier and their functional and operational characteristics.

[0048] The decision module 110 of the arbitration engine 106 may then analyze the node message(s) 214 to determine the manner in which the different node groups, such as node groups 112-1, 2, are to be managed in the event of a cluster-related event. The analysis of the node message(s) 214 is based on the arbitration policies 216. The arbitration policies 216 may prescribe one or more conditions which are to be conformed with for selecting the manner based on which the different node groups (such as node groups 112-1, 2) are to

be arbitrated.

[0049] For example, the arbitration policies 216 may provide that the number of nodes within a given node group be the basis on which the arbitration between the node groups is to be carried out. In such a case, depending on the number of nodes within the different node groups, the arbitration engine 106 may determine which node group is to be retained and which is to be gracefully shut down. In another example, the arbitration policies 216 may prescribe analysis of the node message(s) 214 based on the criticality, or characteristics of the workloads. In such a case, any node group handling critical workloads would be retained, whereas other node groups handling less or non-critical workloads would be allowed to gracefully shut down and would be removed from the cluster. In a similar manner, the arbitration policies 216 may also be based on other parameters, previous, i.e., historical performance and operational information associated with the node, computational resources, number of applications already executing on the node group, state of the nodes within node group, and state of the application.

[0050] Based on the analysis, the decision module 110 may determine which node group is to be removed from the cluster, and which of the node groups are to be retained. For example, the decision module 110 within the arbitration engine 106, may determine the node group 112-1 to be handling critical workloads. The criticality or characteristics of the workloads may be determined based on the functional and operational characteristics of workload present as part of node message(s) 214. Once the criticality or the characteristics of the workloads being handled by the node group 112-1 is determined, the decision module 110 may compare the level of criticality of the workloads with a threshold level prescribed by the arbitration policies 216. If it is determined that the level of criticality exceeds the threshold level, the node group 112-1 may be retained and the node group 112-2 may be gracefully shut down. In such a manner, the arbitration engine 106 may determine from amongst the node groups 112-1, 2, which node groups are to be retained, and which are to be shut down.

[0051] Accordingly, the determinations by the arbitration engine 106 may be

utilized for affecting the shutting down of the determined candidate node groups. In one example, load balancing systems may transfer existing workloads of the candidate node groups to the node groups which are determined to be retained.

[0052] In one example, to manage the operation of the node groups 112-1, 2, the arbitration engine 106 may further generate a control indication for each of the node groups 112-1, 2. The control indication may be stored as control indication 220 within the cluster arbitration system 102. The control indication 220 may subsequently be communicated to the node groups 112-1, 2. In one example, the control indication 220 may indicate whether any node group is to be retained or is to be removed from the cluster and gracefully shut down. The operation of the node groups 112-1, 2 may then be subsequently carried out in accordance with the control indication 220. For example, if the control indication 220 for node group 112-2 indicates that it is to be removed from the cluster, the graceful shut down of the node group 112-2 may be initiated. Similarly, node groups which are to be retained may continue their operations.

[0053] Therefore, the arbitration between the node groups can be controlled and managed based on the arbitration policies, i.e., arbitration policies 216. The arbitration policies 216 may also be configured depending on the need and the nature of the applications being hosted on the nodes. In this way, the arbitration between the various groups is more objective and efficient, thereby ensuring that the applications operate seamlessly on the clusters.

[0054] FIGS. 3A and 3B provide various example implementations for arbitrating between node groups within a cluster. FIGS. 3A-3B provide an illustration depicting different types of cluster-related event, affecting node groups having an unequal number of nodes. In FIG. 3A, a scenario is depicted in which node group 112-1 has two nodes and node group 112-2 has three nodes. In the present example, it may be assumed that the node group 112-1 is handling critical workloads and node group 112-2 is handling non-critical workloads. As depicted, the environment 100 may experience a partitioned network thereby severing the communication between the node group 112-1 and the node group 112-2. On occurrence of the partitioning of the network, the

arbitration engine 106 may determine the criticality or characteristics of the workloads being handled by both the node group 112-1 and the node group 112-2. Since the node group 112-1 is handling critical workloads the arbitration engine 106 may determine to retain the node group 112-1 despite the fact that the number of nodes within node group 112-2 is greater than the number of nodes in the node group 112-1. Accordingly, the node group 112-2 may be allowed to be shut down.

[0055] FIG. 3B depicts an example in which the node group 112-1 has three nodes and node group 112-2 has two nodes. The node group 112-1 is handling plurality of critical workloads. For the present example, the workload to be handled requires that at least three nodes be present within the node group 112-1. It may be the case that the entire node group 112-1 may experience an outage. In such a case, the arbitration engine 106 based on the criticality of or characteristics of workloads may transfer the workloads to the node group 112-2 despite the fact that the nodes within the node group 112-2 may not fulfil the optimum requirement of computational resources. In such a case, the need for waiting to reinstate the node group 112-1 is eliminated. The workloads may be handled by the node group 112-2, with the available resources. As a result, the application availability is ensured till the time the node group 112-1 is brought back online.

[0056] FIGS. 4 and 5 illustrate example methods 400 and 500, respectively, for arbitrating between node groups within a cluster, according to an implementation of the present subject matter. The order in which the methods are described is not intended to be construed as a limitation, and any number of the described method blocks may be combined in any order to implement the aforementioned methods, or an alternative method. Furthermore, methods 400 and 500 may be implemented by processing resource or computing device(s) through any suitable hardware, non-transitory machine readable instructions, or combination thereof.

[0057] It may also be understood that methods 400 and 500 may be performed by programmed computing devices, such as the cluster arbitration system 102 as depicted in FIGS. 1-3. Furthermore, the methods 400 and 500 may be executed based on instructions stored in a non-transitory computer readable medium. The

non-transitory computer readable medium may include, for example, digital memories, magnetic storage media, such as one or more magnetic disks and magnetic tapes, hard drives, or optically readable digital data storage media. Although, the methods 400 and 500 are described below with reference to the cluster arbitration system 102 as described above, other suitable systems for the execution of these methods can also be utilized. Additionally, implementation of these methods is not limited to such examples.

[0058] At block 402, in response to an occurrence of a cluster-related event, one node from each of a plurality of node groups in a cluster, is determined. For example, within one of the nodes within node group 112-1, such as coordinator node 114-2, may be selected as a coordinator node. Similarly, within the node group 112-2, node 116-2 is selected as a coordinator node (referred to as the coordinator node 116-2). In the present example, the coordinator nodes for each of the node groups 112-1, 2 are determined on occurrence of a cluster-related event. In another example, the coordinator node 114-2 and the coordinator node 116-2 are selected by each of the nodes within the respective node groups 112-1, 2.

[0059] At block 404, a control indication is received by coordinator node for each of the node groups. For example, each of the node groups 112-1, 2 may receive a control indication from the arbitration engine 106 within the cluster arbitration system 102. In the present example, the control indication, received as control indication(s) 220, may be generated by the decision module 110 of the arbitration engine 106. The decision module 110 may generate control indication(s) 220 for each of the node groups 112-1, 2 based on the node message(s) 220 received from the respective node groups and the arbitration policies 216.

[0060] At block 406, a set of node groups within the cluster which are to be retained within the cluster, are determined based on the control indication. Accordingly, one or more node groups which are to be gracefully shut down are also determined. To this end, the arbitration engine 106 may select, based on the node-related information received as part of node message(s) 214, which of the node groups within the cluster are to be retained and for which the graceful

shutdown is to be initiated. As mentioned previously, node groups, such as node groups 112-1, 2, which are to be retained and which of the nodes are to be shut down, as determined by the arbitration engine 106 based on the arbitration policies 216.

[0061] FIG. 5 provides another example method 500 for arbitrating between node groups within a cluster. At block 502, occurrence of a cluster-related event is determined. For example, the arbitration engine 106 may determine the occurrence of a cluster-related event. In one example, the cluster-related event may be detected and an appropriate message may be communicated to the cluster arbitration system 102. The arbitration engine 106 of the cluster arbitration system 102 may subsequently determine whether a cluster-related event has occurred. In one example, the cluster-related event results in the communication channel between the node group 112-1 and the node group 112-2 being lost.

[0062] At block 504, one node from each of the affected node groups is selected as a coordinator node. For example, within the node group 112-1, a node 114-2 may be elected as a coordinator node (i.e., coordinator node 114-2). Similarly, for the node group 112-2, node 116-2 may be elected as the coordinator node (i.e., coordinator node 116-2). In the present example, each of the nodes 114, 116 may determine which node is to be selected as a coordinator node.

[0063] At block 506, each of the coordinator nodes obtain and provide node-related information as a message. For example, the coordinator node 114-2 obtains the node-related information from each of the nodes 114 within the node group 112-1. Similarly, the coordinator node 116-2 obtains node-related information from each of the nodes 116 within the node group 112-2. Once the node-related information for each of the nodes 114, 116 is received, the coordinator node 114-2 and the coordinator node 116-2 transmit the same as a message to the cluster arbitration system 102.

[0064] At block 508, the node-related information within the received message is analyzed based on configurable policies. For example, the cluster arbitration system 102 may save the node-related information as node message(s) 214. The node message(s) 214 may be received by the messaging

module 108. The node message(s) 214 may be subsequently analyzed by the decision module 110 of the arbitration engine 106 based on the policy definitions stored in arbitration policies 216. The arbitration policies 216 may prescribe the criteria based on which the arbitration of the node groups is to be carried. In one example, the arbitration policies 216 may provide criteria based on which the arbitration engine 106 may determine which of the node groups 112-1, 2 are to be retained and which are to be gracefully shut down.

[0065] At block 510, one or more control indications are generated based on the analysis of the node-related information, and communicated to the respective node groups. For example, the decision module 110 based on the analyzing the node-related information included in the node message(s) 214, may generate one or more control indications for controlling the operations of the node groups 112-1, 2. In the present example, the arbitration engine 106 may determine that the node group 112-1 is to be retained and that node group 112-2 is to be shut down. Accordingly, the decision module 110 may generate control indications for further controlling how the node group 112-1 is to manage its existing workloads. Similarly, the decision module 110 may also generate control indications, indicating that a graceful shut down is to be initiated, say for the node group 112-2. In one example, the control indication is stored as control indication 220.

[0066] At block 512, the operations of the nodes within the node groups are controlled based on the control indications. For example, each of the node groups 112-1, 2 may either continue their operations or may initiate shut down process depending on their control indication. Continuing with the previous example, for node group 112-1 (which is to be retained), other computational entities, such as a load balancing server, may determine the manner in which the existing workloads of node group 112-1 are handled. For node group 112-2 (which is to be shut down) graceful shutdown may be initiated.

[0067] FIG. 6 illustrates a cluster arbitration system 602 for arbitrating between node groups within a cluster, according to an example of the present disclosure. The cluster arbitration system 602 may be part of a public networking environment or a private networking environment, or a combination

thereof. In one implementation, the cluster arbitration system 602 includes a processing resource 604 communicatively coupled to a machine-readable storage medium 606 (referred to as the storage medium 606). In the example of FIG. 6, cluster arbitration system 602, the machine-readable storage medium 606 comprising (e.g., encoded with) instructions 608-610 executable by processing resource 604. In some examples, storage medium 606 may include additional instructions. In other examples, the functionalities described herein in relation to instructions 608-610, and any additional instructions described herein in relation to storage medium 606, may be implemented as engines or modules comprising any combination of hardware and programming to implement the functionalities of the engines, as described below. Furthermore, the functionalities described herein may be implemented as a result of the execution of the instructions 608-610.

[0068] As used herein, the cluster arbitration system 602 may be a desktop computer, laptop (or notebook) computer, workstation, tablet computer, mobile phone, smart device, server, blade enclosure, or any other processing device or equipment. In examples described herein, a processing resource may include, for example, one processor or multiple processors included in a single computing device or distributed across multiple computing devices.

[0069] In operation, on occurrence of a cluster-related event, instructions 608 may analyze one or more received messages associated with a plurality of node groups. The cluster-related event may occur within a cluster formed of a plurality of node groups, such as node groups 112-1, 2 (as depicted in FIGS. 1-3). Each of the node groups 112-1, 2 may further include one or more nodes. During the occurrence of a cluster-related event, at least one node is no longer in communication with any other node within the cluster. In one example, this may happen as a result of a partitioned network in which the communication channel between two or more node groups is disrupted. It may also happen in case any node group itself goes down or becomes unavailable.

[0070] The messages associated with a node group may be received by a coordinator node. The coordinator node may be selected from nodes present within the respective node groups 112-1, 2. In one example, nodes 114-2, 116-2

are selected as coordinator node 114-2 and coordinator node 116-2, respectively.

[0071] In another example, the instruction 608 may further enable the coordinator nodes, namely coordinator node 114-2 and coordinator node 116-2 to obtain node-related information about each node within the cluster. For example, as a result of the instructions 608, the coordinator node 114-2 obtains the node-related information from each of the nodes 114 within the node group 112-1. Similarly, the coordinator node 116-2 obtains node-related information from each of the nodes 116 within the node group 112-2. Once the node-related information for each of the nodes 114, 116 is received, the coordinator node 114-2 and the coordinator node 116-2 transmit the same as a message to the cluster arbitration system 602.

[0072] The message received from the various coordinator nodes, i.e., the coordinator node 114-2 and the coordinator node 116-2 is analyzed based on the instructions 608. In one example, the node-related information within the messages may be subsequently analyzed by the instruction 608 based on the policy definitions, for example, policies stored in arbitration policies 216. In one example, the arbitration policies 216 may provide criteria based on which it may be determined which of the node groups 112-1, 2 are to be retained, and which are to be gracefully shut down.

[0073] Based on the analysis, the instructions 608 may determine which of the node groups 112-1, 2 is to be retained and which is to be gracefully shut down. On determining the same, the instructions 610 may generate one or more control indications based on the analysis of the node-related information. The generated control indications are communicated to the respective node groups. In the present example, the instructions 610 based on the analyzing the node-related information, may generate one or more control indications for controlling the operations of the node groups 112-1, 2. If it is determined that the node group 112-1 is to be retained and that node group 112-2 is to be shut down, the arbitration engine 614 may generate the control indications accordingly.

[0074] Once the control indications are generated, the same may be communicated to the respective node groups 112-1, 2 by the instructions 610.

In one example, for node group 112-2 (which is to be shut down), the graceful shutdown may be initiated based on the control indication. In another example, accordingly, the workloads of the node group 112-2 are transferred to the node group 112-1 (which was determined to be retained). In one example, the control indications may be stored as control indication 220.

[0075] Although examples for the present disclosure have been described in language specific to structural features and/or methods, it is to be understood that the appended claims are not necessarily limited to the specific features or methods described. Rather, the specific features and methods are disclosed and explained as examples of the present disclosure.

I/We claim:

1. A cluster arbitration system for a cluster, the cluster including a plurality of node groups, with each of the plurality of node groups comprising at least one node, the system comprising:

a processor;

an arbitration engine coupled to the processor, the arbitration engine further comprises:

a messaging module, the messaging module is to,

in response to occurrence of a cluster-related event, receive a message from a coordinator node associated with each of the plurality of node groups, the message includes information associated with operation of each of the node groups; and

a decision module, the decision module is to,

based on information included within the message, determine whether to gracefully shut down a node group from amongst the plurality of node groups within the cluster.

2. The system as claimed in claim 1, wherein the message includes one of:
an indication of a level of criticality of application workloads running on each node within each of the plurality of node groups;

information depicting state of each of the node; and

analytical information depicting historical operational data for each node within each of the plurality of node groups.

3. The system as claimed in claim 1, wherein the cluster-related event comprises one of:

a network partition, wherein on occurrence of the network partition connectivity between at least two node groups, from amongst the plurality of node groups, is lost; and

a node failure.

4. The system as claimed in claim 1, wherein the decision module is to further:

determine for removal from the cluster, at least one node group from amongst the plurality of node groups, which has lost connectivity with any of the other node groups.

5. The system as claimed in claim 1, wherein the decision module is to, based on information included within the message, determine whether to retain another node group within the cluster.

6. The system as claimed in claim 4, wherein the decision module is to determine the at least one node group for removal based on configurable policies.

7. A method for arbitrating over cluster comprising a plurality of node groups, the method comprising:

determining one node from each node group as a coordinator node, on occurrence of cluster-related event affecting the plurality of node groups within the cluster;

receiving, by the coordinator node, a control indication from an arbitration engine, wherein the control indication is generated based on analyzing node-related information corresponding to each node within each of the node groups using configurable policies; and

based on the control indication, determining a set of node groups which are to be retained within the cluster, from amongst the plurality of node groups.

8. The method as claimed in claim 7, wherein the node-related information comprises computational resources, state of executing applications, criticality of workloads, complexity of workloads executing on each of the one or more nodes within each of the plurality of node groups, and historical operational data for each of the one or more nodes within each of the plurality of node groups.

9. The method as claimed in claim 7, wherein the method further comprises:

obtaining node-related information for each node within each of the node groups, by the coordinator node; and

communicating, by the coordinator node, the node-related information to the arbitration engine.

10. The method as claimed in claim 7, wherein the control indication is generated based on analyzing the node-related information.

11. The method as claimed in claim 7, based on the control indication the method further comprises:

further determining node groups, other than the determined set of node groups, as candidate node groups for removal from the cluster; and

initiating a graceful shutdown of the candidate node groups.

12. The method as claimed in claim 11, wherein for the candidate node groups, the method further comprises:

determining application load on the candidate node group;

shifting the application load to another node group from the candidate node group, prior to removing the candidate node group from the cluster.

13. The method as claimed in claim 7, wherein number of nodes within the node groups is unequal.

14. A non-transitory machine-readable storage medium encoded with instructions executable by a processing resource to:

analyze a received message associated with a node group from among a plurality of node groups based on at least one configurable policy, wherein the message includes information associated with operation of the node group; and

cause generation, based on the analyzing of the received message, of a control indication for use in controlling operations of one or more nodes within the plurality of node group, in accordance with the control indication.

15. The machine-readable storage medium as claimed in claim 14 further comprising instructions executable by a processing resource to shut down a candidate node group from among the plurality of node groups based on the control indication.

1/7

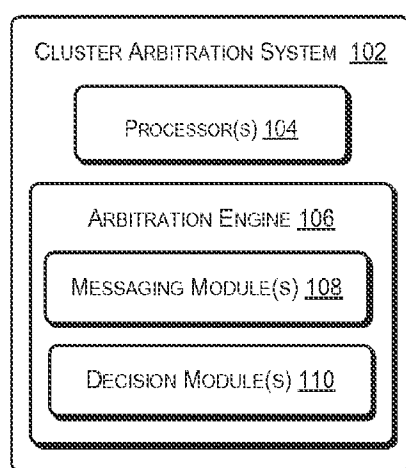


FIG. 1A

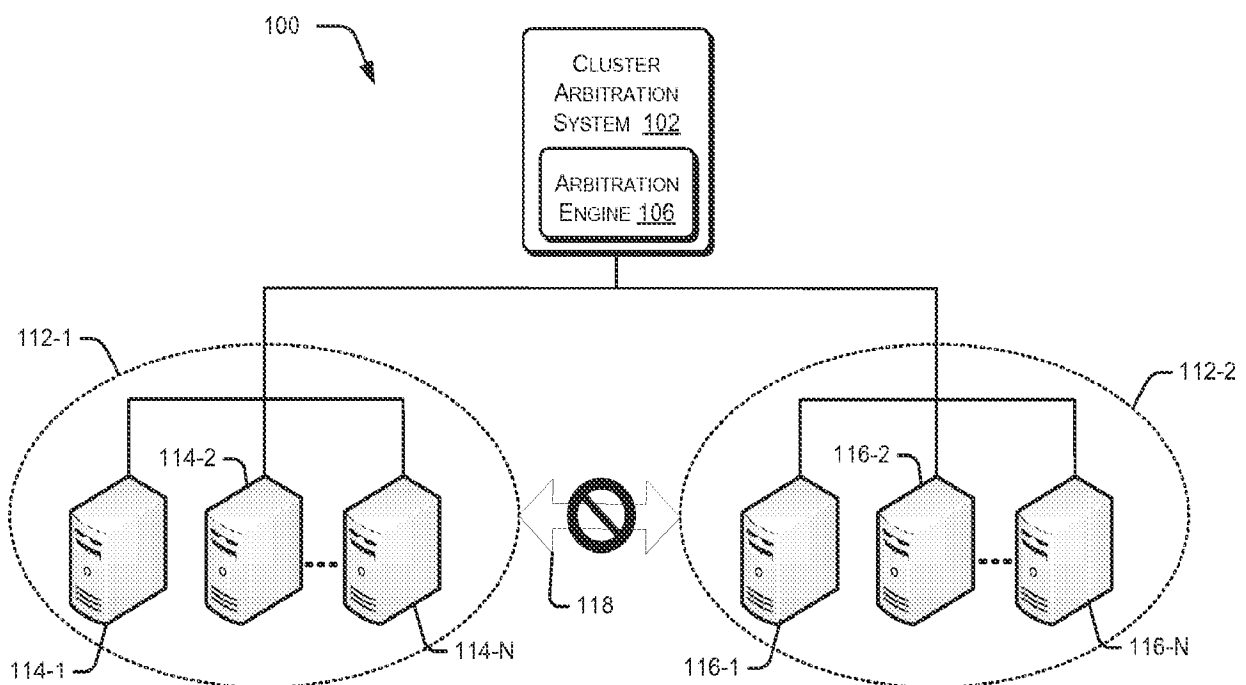
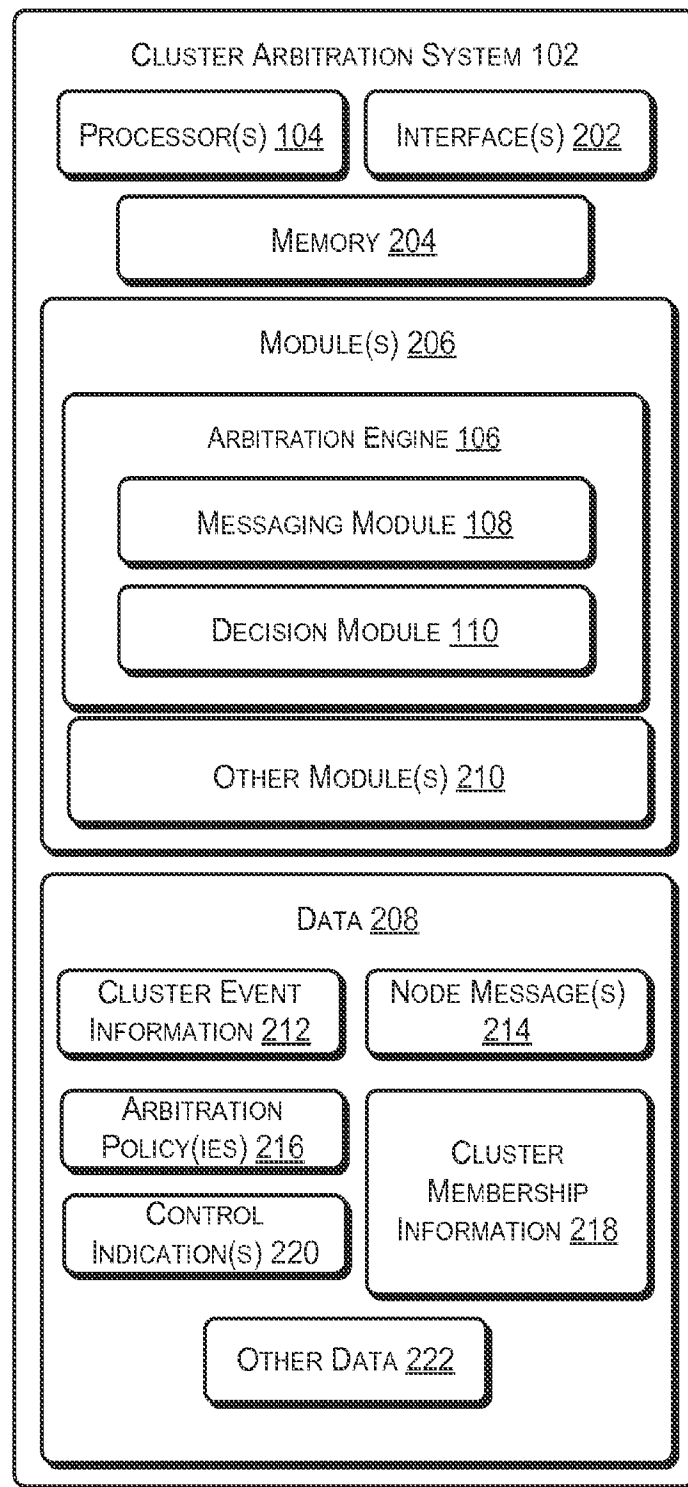
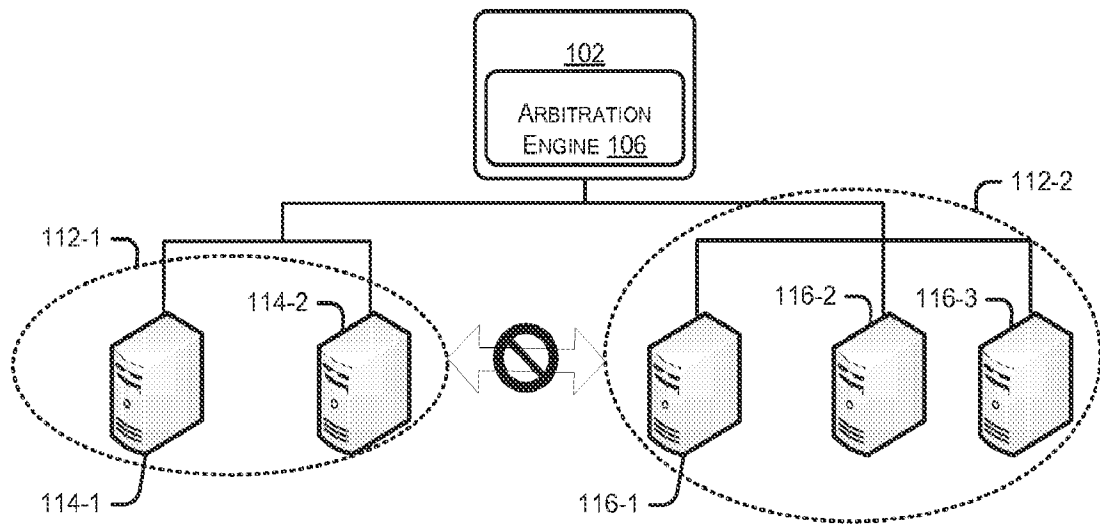


FIG. 1B

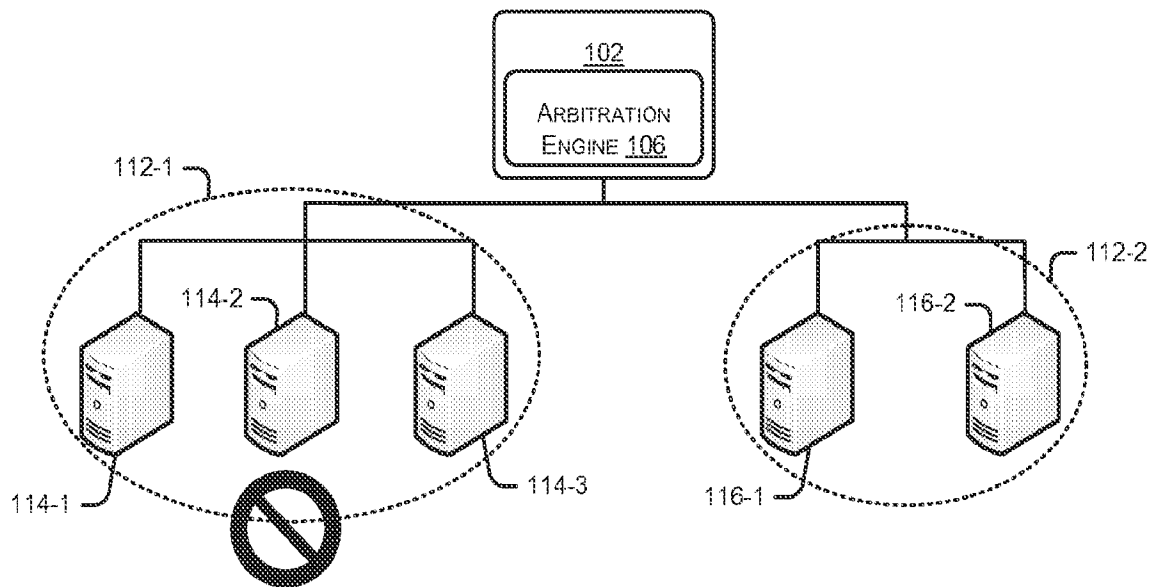
3/7

**FIG. 2**

4/7



[A]



[B]

FIG. 3

5/7

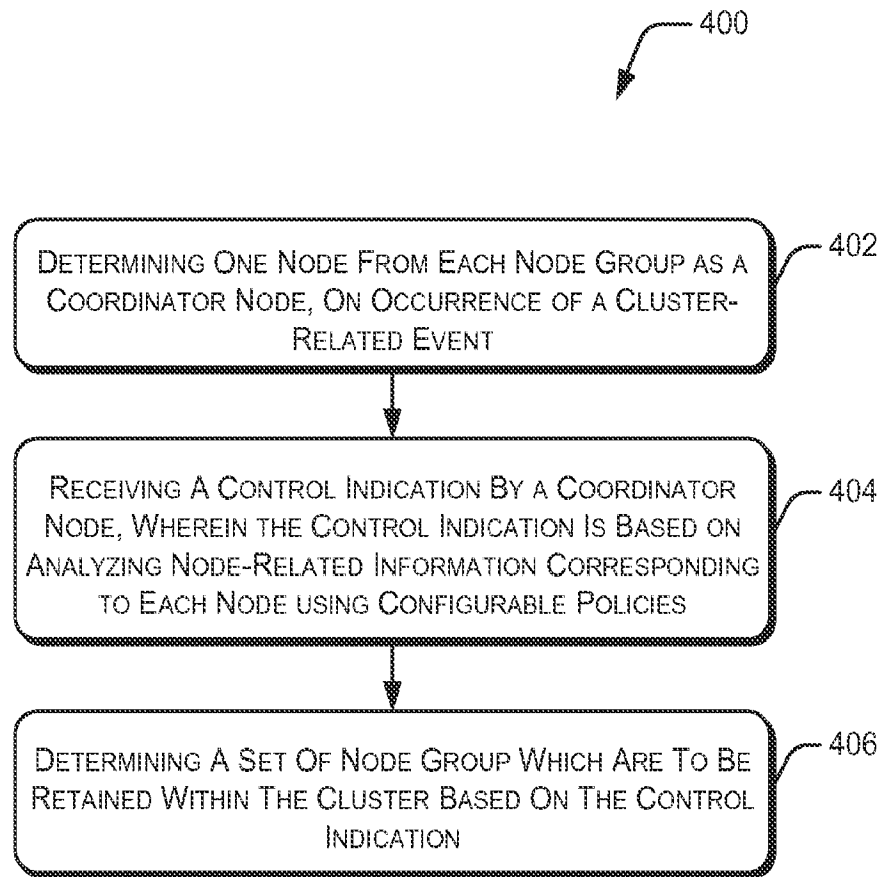
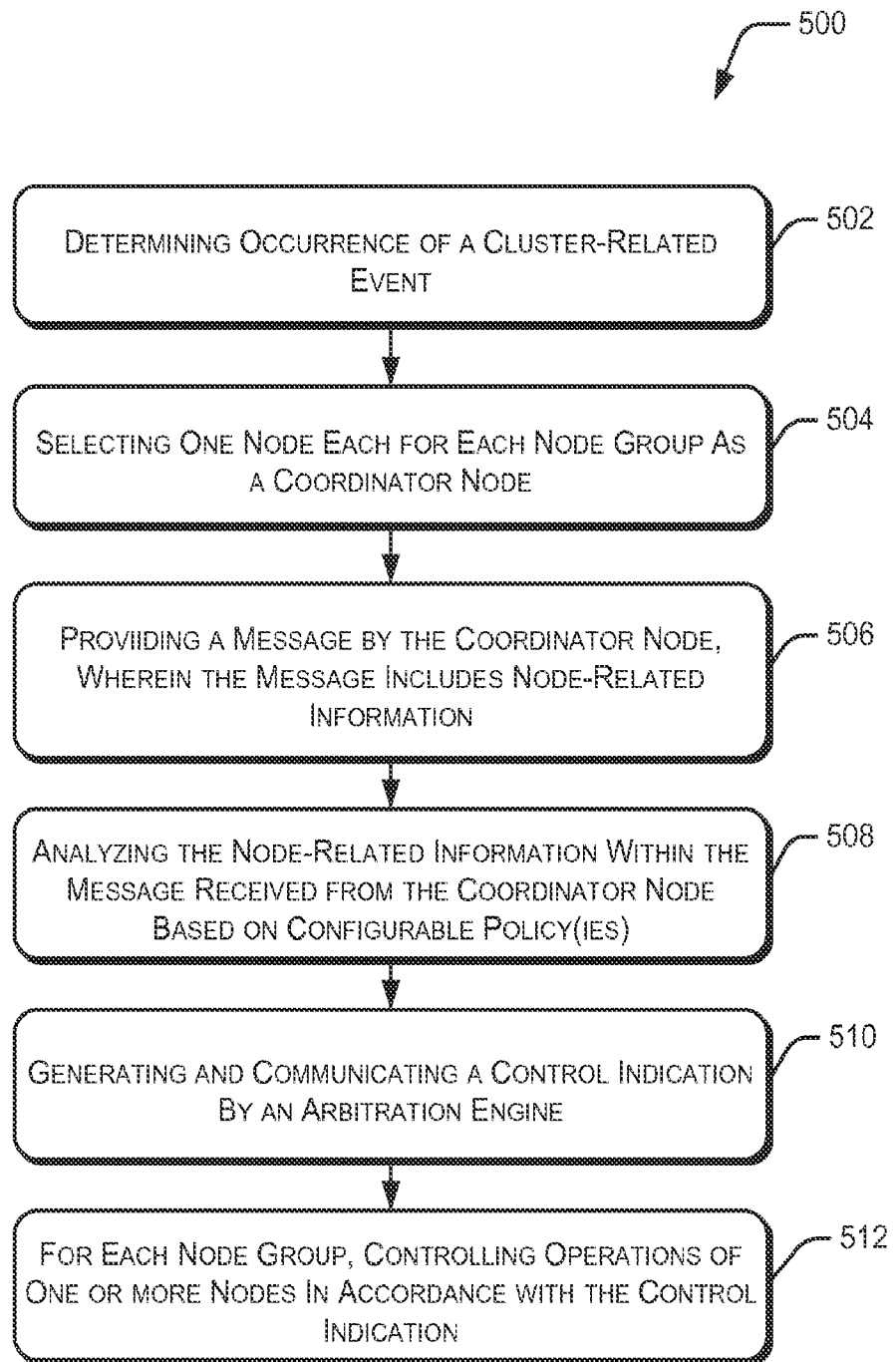


FIG. 4

6/7

**FIG. 5**

7/7

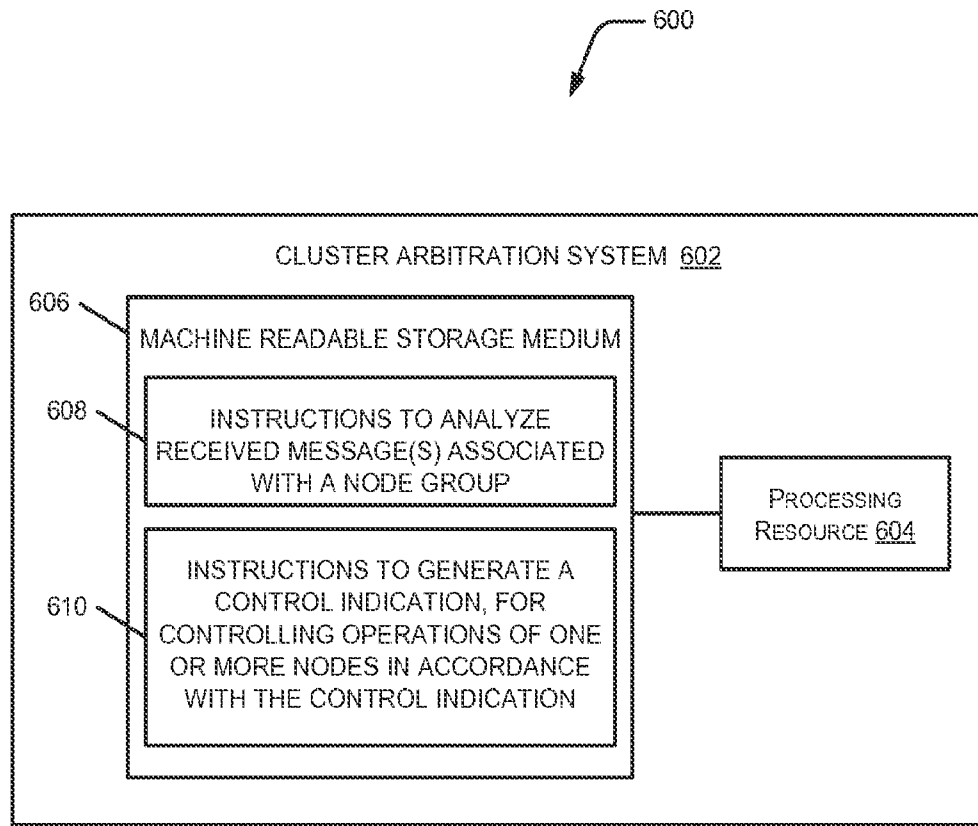


FIG. 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2015/026064**A. CLASSIFICATION OF SUBJECT MATTER****H04L 29/08(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)

H04L 29/08; G06F 15/16; G06F 11/07; G06F 15/173; G06F 15/177; G06F 11/00

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: arbitration, node, shutdown, analyze, information.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2006-0036896 A1 (ROD GAMACHE et al.) 16 February 2006 See paragraphs [0035], [0043]-[0054]; and figure 2.	1,3-7,9-11,13-15
A		2,8,12
A	US 8650281 B1 (ABHIJIT TOLEY et al.) 11 February 2014 See column 4, line 52 - column 7, line 40; column 9, line 7 - column 11, line 16; and figures 1-3.	1-15
A	US 2008-0184061 A1 (SHAIENDRA TRIPATHI et al.) 31 July 2008 See paragraphs [0016]-[0047]; and figures 3-6.	1-15
A	US 8024432 B1 (SUMIT SHARMA et al.) 20 September 2011 See column 2, line 6 - column 6, line 44; and figure 1.	1-15
A	US 2012-0239814 A1 (MARKUS MUELLER et al.) 20 September 2012 See paragraphs [0052]-[0060]; and figures 4-5.	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

16 September 2015 (16.09.2015)

Date of mailing of the international search report

25 September 2015 (25.09.2015)

Name and mailing address of the ISA/KR

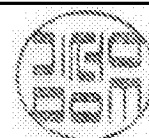
International Application Division
Korean Intellectual Property Office
189 Cheongsu-ro, Seo-gu, Daejeon Metropolitan City, 302-701,
Republic of Korea

Facsimile No. +82-42-472-7140

Authorized officer

KIM, Seong Woo

Telephone No. +82-42-481-3348



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2015/026064

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2006-0036896 A1	16/02/2006	AT 541259 T AU 2000-37295 A1 AU 3729500 A EP 1222540 A1 EP 1222540 B1 US 2002-0161889 A1 US 2006-0090095 A1 US 2010-0250750 A1 US 6401120 B1 US 6938084 B2 US 7774469 B2 US 7984155 B2 WO 00-58824 A1	15/01/2012 16/10/2000 16/10/2000 17/07/2002 11/01/2012 31/10/2002 27/04/2006 30/09/2010 04/06/2002 30/08/2005 10/08/2010 19/07/2011 05/10/2000
US 8650281 B1	11/02/2014	None	
US 2008-0184061 A1	31/07/2008	JP 2008-192139 A JP 4505763 B2 US 7840833 B2	21/08/2008 21/07/2010 23/11/2010
US 8024432 B1	20/09/2011	None	
US 2012-0239814 A1	20/09/2012	US 2012-239810 A1 US 8949430 B2	20/09/2012 03/02/2015