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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 September 2016 (01.09.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/137502 A1

(51) International Patent Classification:
G06F 15/76 (2006.01)

(21) International Application Number:
PCT/US2015/018053

(22) International Filing Date:
27 February 2015 (27.02.2015)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: **HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP** [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).

(72) Inventors: **WILKINSON, Lyle**; 1501 Page Mill Rd., Palo Alto, CA 94304 (US). **EASTERLING, Tyler**; 3404 E Harmony Rd., Fort Collins, CO 80528 (US).

(74) Agents: **BELL, Scott** et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road, Mail Stop 79 , Fort Collins, CO 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, 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))*

WO 2016/137502 A1

(54) Title: ANALYZING HARDWARE SYSTEMS

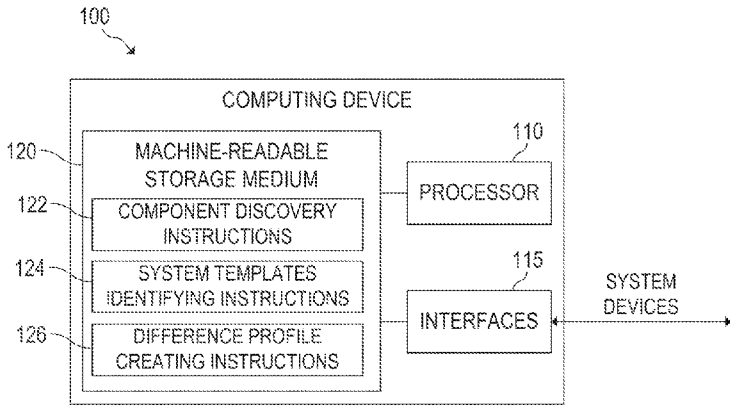


FIG. 1

(57) Abstract: Examples relate to analyzing hardware systems. In some examples, an interface is used to discover components and connectivity attributes of the components. System templates are then identified based on the components and the connectivity attributes. For each system template, a difference profile describing differences between the system template and the components is created.

ANALYZING HARDWARE SYSTEMS

BACKGROUND

[0001] Site integration of complex hardware systems are typically performed via manual system integration of physical components. The integration processes can be accomplished using detailed "cookbooks" (i.e., documented instructions). Detailed verification of the system requirements may be used, which can introduce errors in the instructions and/or the manual verification process.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The following detailed description references the drawings, wherein:

[0003] FIG. 1 is a block diagram of an example computing device for analyzing hardware systems;

[0004] FIG. 2 is a block diagram of an example system computing device including a computing device for analyzing components in a hardware system;

[0005] FIG. 3 is a flowchart of an example method for execution by a computing device for analyzing hardware systems;

[0006] FIG. 4 is an example system template for use in analyzing hardware systems; and

[0007] FIG. 5 is an example difference map resulting from the analysis of a hardware system.

DETAILED DESCRIPTION

[0008] As described above, complex hardware systems can be manually configured according to cookbooks. Such processes are time consuming and error prone. Examples herein use system templates that describe system types along with their component containment, hierarchy, and other dependencies as well as methods to configure the components. The system templates can be collected into a template library, which allows a collection of real system

components to be examined and compared to the template library to determine compatibility of the components with the templates.

[0009] The template library enables several use cases: (1) suggesting to a system administrator alternative system types, each derived from a template that can be created from those components; (2) determining if a set of components satisfies the requirements of a particular template; (3) indicating changes (e.g., hardware addition/substitution, component connectivity changes, power configuration changes, etc.) that would allow a set of components to satisfy the requirements of a particular template; and (4) fully configuring a set of components to support the application of a particular template (e.g. applying applicable profiles to servers, applying BIOS settings, setting network configurations, allocating storage partitions and setting RAID levels on storage arrays, etc.).

[0010] In some examples, an interface is used to discover components and connectivity attributes of the components. System templates are then identified based on the components and the connectivity attributes. For each system template, a difference profile describing differences between the system template and the components is created.

[0011] Referring now to the drawings, FIG. 1 is a block diagram of an example computing device 100 for analyzing hardware systems. The example computing device 100 may be a desktop computer, server, notebook computer, tablet, or other device suitable for analyzing hardware systems as described below. In the example of FIG. 1, networking device 100 includes processor 110, interfaces 115, and machine-readable storage medium 120.

[0012] Processor 110 may be one or more central processing units (CPUs), microprocessors, and/or other hardware devices suitable for retrieval and execution of instructions stored in machine-readable storage medium 120. Processor 110 may fetch, decode, and execute instructions 122, 124, 126 to enable analyzing hardware systems, as described below. As an alternative or in addition to retrieving and executing instructions, processor 110 may include one or more electronic circuits comprising a number of electronic components for performing the functionality of one or more of instructions 122, 124, 126.

[0013] Interfaces 115 may include a number of electronic components for communicating with computing devices. For example, interfaces 115 may be wireless interfaces such as wireless local area network (WLAN) interfaces and/or physical interfaces such as Ethernet interfaces, Universal Serial Bus (USB) interfaces, external Serial Advanced Technology Attachment (eSATA) interfaces, or any other physical connection interface suitable for communication with end devices. In operation, as detailed below, interfaces 115 may be used to send and receive data to and from other computing devices.

[0014] Machine-readable storage medium 120 may be any electronic, magnetic, optical, or other physical storage device that stores executable instructions. Thus, machine-readable storage medium 120 may be, for example, Random Access Memory (RAM), Content Addressable Memory (CAM), Ternary Content Addressable Memory (TCAM), an Electrically-Erasable Programmable Read-Only Memory (EEPROM), flash memory, a storage drive, an optical disc, and the like. As described in detail below, machine-readable storage medium 120 may be encoded with executable instructions for enabling analyzing hardware systems.

[0015] Component discovery instructions 122 use interfaces 115 to scan a network in order to identify components and connections between them. Examples of components include access points, switches, racks, server computing devices, storage arrays, etc. Component discovery instructions 122 can use various techniques for scanning the network such as, but not limited to, domain name system (DNS) service discovery, dynamic host configuration protocol (DHCP) packet inspection, DNS packet inspection, network basic input/output system (NetBIOS) inspection, etc. The discovered components can be arranged in a system map that shows the relationships between the components in the network.

[0016] System templates identifying instructions 124 identify system templates that match the components and relationships (i.e., system map) discovered above. The system templates can be preconfigured by a system administrator to create a template library of potential system configurations. Each template of the

list of matching system templates should include at least some of the components from the system map.

[0017] Difference profile creating instructions 126 creates difference profiles for each template in the list of matching templates. Each difference profile specifies how closely a corresponding template matches the system map (e.g., the percentage of components from system map that are in the template, etc.). Difference profiles can be used to generate difference maps, which visually show the difference between the system map and each corresponding matching template. The difference maps can then be used by an administrator to identify a target template that can be used to modify the current hardware system.

[0018] FIG. 2 is a block diagram of an example system 200 including computing device 202 and components 210A, 210N for analyzing hardware systems. The components of computing device 202 may be similar to the corresponding components of computing device 100 described with respect to FIG. 1.

[0019] As illustrated, computing device 202 is connected to components 210A, 210N via interface(s) 204. For example, computing device 202 may be connected to a LAN or WAN that includes the components 210A, 210N. In another example, computing device 202 may be connected to a rack that includes components 210A, 210N by a USB cable. Because computing device 202 is connected to components 210A, 210N, computing device 202 is able to discover the components 210A, 210N and how the components 210A, 210N are connected to each other.

[0020] Computing device 202 also includes a configuration application 206. Configuration application 206 can provide an interface for managing system templates (e.g., creating, modifying, searching system templates, etc.). For example, a user interface can be provided that allows a user to drag-and-drop components into a system template map. In another example, auto-discovery scripts can be used to build system templates based on existing hardware systems.

[0021] Configuration application 206 can also use the discovered data about components 210A, 210N to analyze the hardware system that includes the components 210A, 210N. For example, the discovered data can be used to generate system maps, which are compared to preconfigured system templates. In this example, configuration application 206 may allow a user to review matching templates to determine how to reconfigure the hardware system. In some cases, the differences between the components 210A, 210N and the matching templates can be visually displayed in a difference map such as the one described below with respect to FIG. 5.

[0022] When a template is selected from the matching templates, the selected template can then be used by configuration application 206 to reconfigure the components 210A, 210N. Specifically, configuration application 206 can execute configuration scripts that modify the configuration of the components 210A, 210N to match the selected system template. In this case, the configuration scripts can modify various aspects of the components 210A, 210N such as firmware configurations, application configurations, application installations, network configurations, etc. For example, the configuration scripts can reconfigure an access point to modify connections between components 210A, 210N while also installing an application on a blade of an enclosure.

[0023] Configuration application 206 can also make recommendations based on the system maps and matching system templates. For example, configuration application 206 may accept input from the user regarding the desired attributes of the hardware system (e.g., data storage, rapid response times, media streaming, etc.). In this example, configuration application 206 may use the planned uses and identified components 210A, 210N to identify the matching templates, which minimize the implementation cost.

[0024] FIG. 3 is a flowchart of an example method 300 for execution by a computing device 100 for analyzing hardware systems. Although execution of method 300 is described below with reference to computing device 100 of FIG. 1, other suitable devices for execution of method 300 may be used such as computing device 202 of FIG. 2. Method 300 may be implemented in the form of

executable instructions stored on a machine-readable storage medium, such as computer readable medium 120 of FIG. 1, and/or in the form of electronic circuitry.

[0025] Method 300 may start in block 305 and continue to block 310, where computing device 100 uses interface(s) to scan a network in order to discover components and connections between them. The discovered components and connections can be used to generate a system map that describes the hardware system. In block 315, computing device 100 identifies system templates that match the system map.

[0026] In block 320, computing device 100 creates difference profiles for each template in the list of matching templates. Each difference profile specifies how closely a corresponding template matches the system map. In block 325, in response to selection of one of the matching templates, configurations scripts are executed to modify the hardware system to match the selected system template. Method 300 may then continue block 330, where method 300 may stop.

[0027] FIG. 4 is an example system template 400 for use in analyzing hardware systems. The system template 400 shows components arrange in a hardware system. The root of system template 400 is rack 402, which is connected to enclosure 404, switch 410, and storage array 420. Enclosure 404 is further connected to blade A 406A of type 1 and blade B 406B of type 2. In this example, system template 400 is shown as a hierarchical tree, where the connection between components is shown as node relationships. Similar hierarchical trees could be used to depict system maps of actual hardware systems that have been discovered as described above.

[0028] FIG. 5 is an example difference map 500 resulting from the analysis of a hardware system. The difference map 500 shows components in a hardware system as compared to the system template 400 of FIG. 4. Difference map 500 includes the same components (rack 502, enclosure A 504A, blade A 506A, blade B 506B, switch 510, storage array 520) as system template 400, which are highlighted to show where the hardware system and the system template 400 overlap. The hardware system also includes additional components (enclosure B 504B, blade M 506M, blade N 506N), which are not highlighted. Difference map

500 allows a user to quickly review an analyzed hardware system to determine how closely it matches a system template. Based on the differences, the user can adapt the hardware system to match the system template.

[0029] The foregoing disclosure describes a number of examples for analyzing hardware systems. The examples disclosed herein facilitate analyzing hardware systems by using auto-discovery to identify characteristics of hardware systems, which are used to search for matching system templates.

CLAIMS

We claim:

1. A computing device for analyzing hardware systems, comprising:

an interface to connect to a plurality of components;

a processor to:

use the interface to discover the plurality of components and connectivity attributes of the plurality of components;

identify a plurality of system templates based on the plurality of components and the connectivity attributes; and

for each system template of the plurality of system templates, create a difference profile describing differences between the system template and the plurality of components.

2. The computing device of claim 1, wherein the processor is further to in response to a user selecting a system template of the plurality of system templates, execute a configuration script to resolve the differences in the difference profile for the system template.

3. The computing device of claim 2, wherein the processor is further to initially filter the plurality of system templates based on desired attributes specified by the user.

4. The computing device of claim 1, wherein the processor is further to generate a difference map for each of the plurality of system templates, wherein the difference map reflects the difference profile of a corresponding system template.

5. The computing device of claim 1, wherein each of the plurality of components are discovered using domain name system service discovery, dynamic host configuration protocol packet inspection, DNS packet inspection, or network basic input/output system inspection.

6. The computing device of claim 1, wherein each of the plurality of system templates are preconfigured by a user based on an existing hardware system.

7. A method for analyzing hardware systems, comprising:

using an interface to discover a plurality of components and connectivity attributes of the plurality of components;

identifying a plurality of system templates based on the plurality of components and the connectivity attributes;

for each system template of the plurality of system templates, creating a difference profile describing differences between the system template and the plurality of components; and

in response to a user selecting a system template of the plurality of system templates, executing a configuration script to resolve the differences in the difference profile for the system template.

8. The method of claim 7, further comprising initially filtering the plurality of system templates based on desired attributes specified by the user.

9. The method of claim 7, further comprising generating a difference map for each of the plurality of system templates, wherein the difference map reflects the difference profile of a corresponding system template.

10. The method of claim 7, wherein each of the plurality of components are discovered using domain name system service discovery, dynamic host configuration protocol packet inspection, DNS packet inspection, or network basic input/output system inspection.

11. The method of claim 7, wherein each of the plurality of system templates are preconfigured by the user based on an existing hardware system.

12. A non-transitory machine-readable storage medium encoded with instructions executable by a processor for analyzing hardware systems, the machine-readable storage medium comprising instructions to:

use an interface to discover a plurality of components and connectivity attributes of the plurality of components;

identify a plurality of system templates based on the plurality of components and the connectivity attributes, wherein each of the plurality of system templates are preconfigured by a user based on an existing hardware system;

for each system template of the plurality of system templates, create a difference profile describing differences between the system template and the plurality of components; and

in response to the user selecting a system template of the plurality of system templates, execute a configuration script to resolve the differences in the difference profile for the system template.

13. The non-transitory machine-readable storage medium of claim 12, wherein the instructions are further to initially filter the plurality of system templates based on desired attributes specified by the user.

14. The non-transitory machine-readable storage medium of claim 12, wherein the instructions are further to generate a difference map for each of the plurality of system templates, wherein the difference map reflects the difference profile of a corresponding system template.

15. The non-transitory machine-readable storage medium of claim 12, wherein each of the plurality of components are discovered using domain name system service discovery, dynamic host configuration protocol packet inspection, DNS packet inspection, or network basic input/output system inspection.

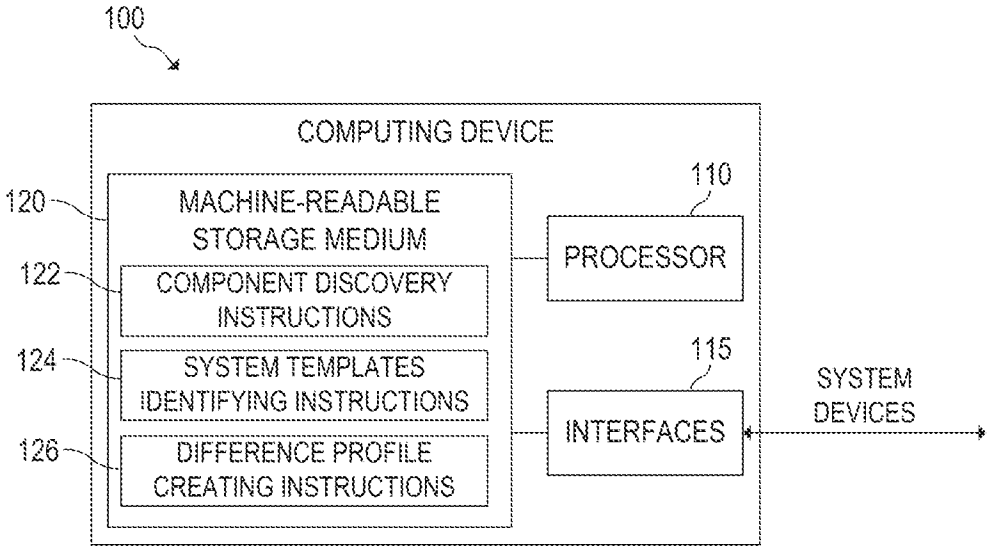

100
COMPUTING DEVICE
120
MACHINE-READABLE STORAGE MEDIUM
122
COMPONENT DISCOVERY INSTRUCTIONS
124
SYSTEM TEMPLATES IDENTIFYING INSTRUCTIONS
126
DIFFERENCE PROFILE CREATING INSTRUCTIONS
110
PROCESSOR
115
INTERFACES
SYSTEM DEVICES

FIG. 1

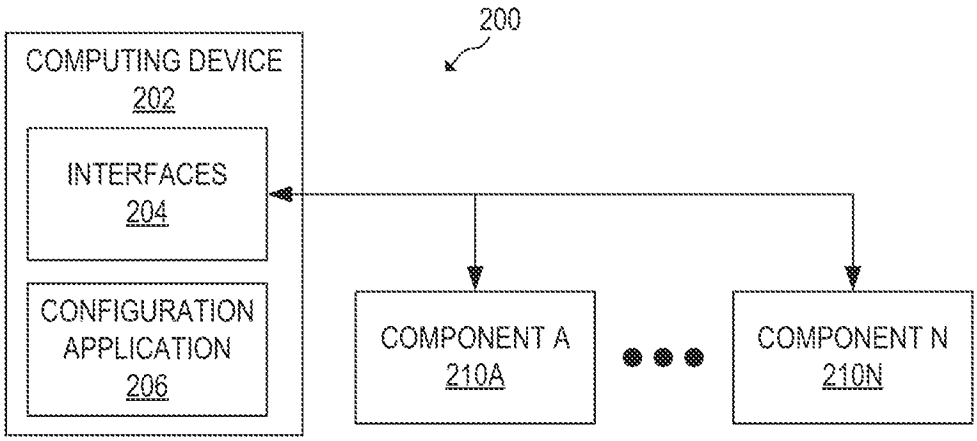

200
COMPUTING DEVICE
202
INTERFACES
204
CONFIGURATION APPLICATION
206
COMPONENT A
210A
COMPONENT N
210N

FIG. 2

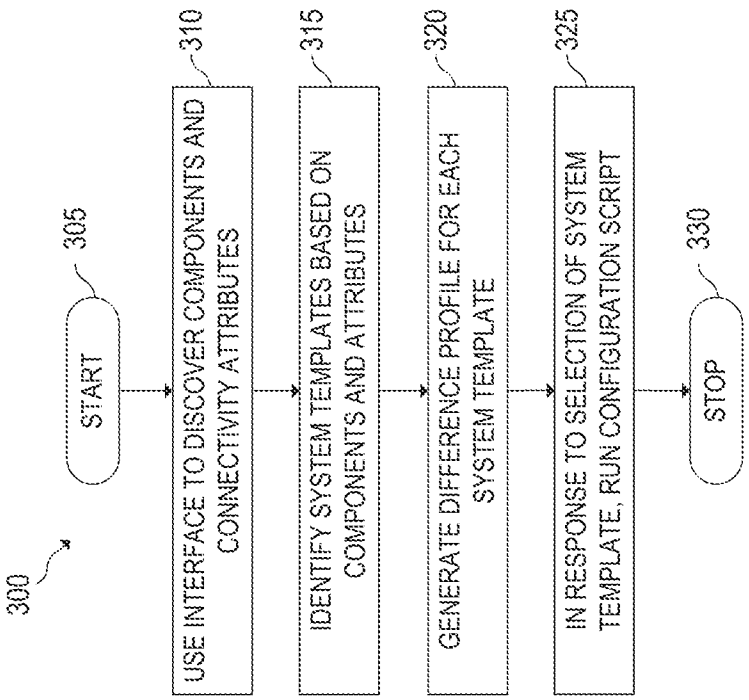

300
START
305
USE INTERFACE TO DISCOVER COMPONENTS AND CONNECTIVITY ATTRIBUTES
310
IDENTIFY SYSTEM TEMPLATES BASED ON COMPONENTS AND ATTRIBUTES
315
GENERATE DIFFERENCE PROFILE FOR EACH SYSTEM TEMPLATE
320
IN RESPONSE TO SELECTION OF SYSTEM TEMPLATE, RUN CONFIGURATION SCRIPT
325
STOP
330

FIG. 3

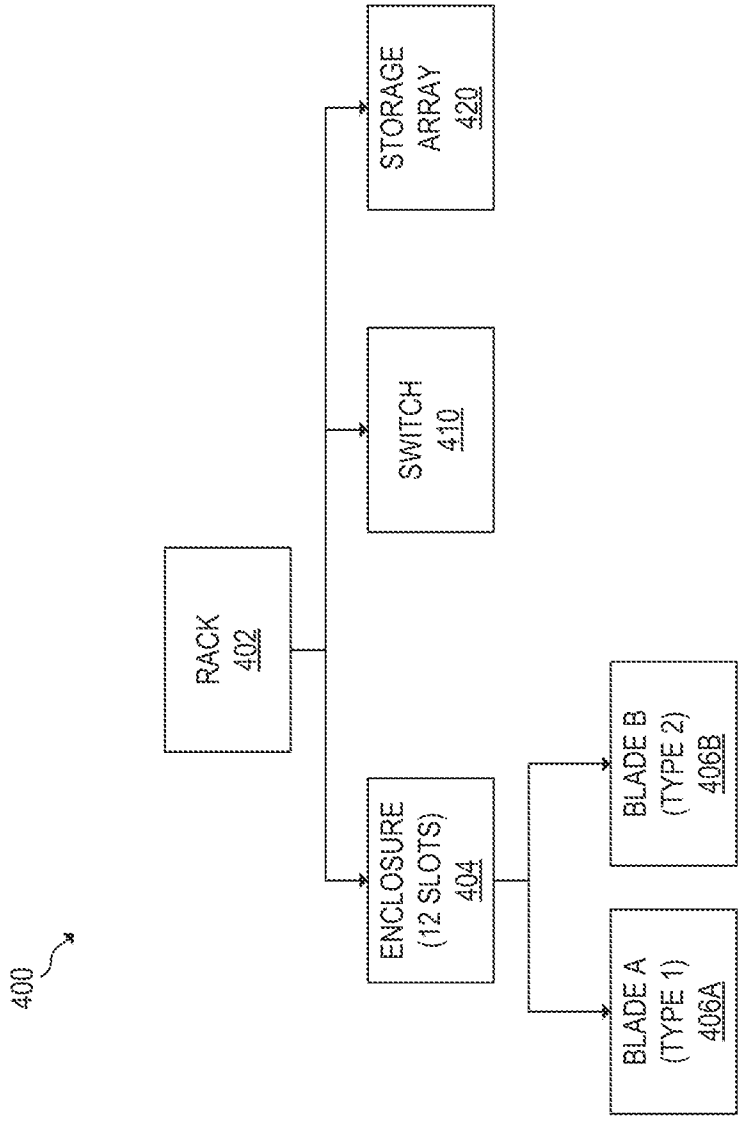

400
RACK
402
STORAGE
ARRAY
420
SWITCH
410
ENCLOSURE
(12 SLOTS)
404
BLADE B
(TYPE 2)
406B
BLADE A
(TYPE 1)
406A

FIG. 4

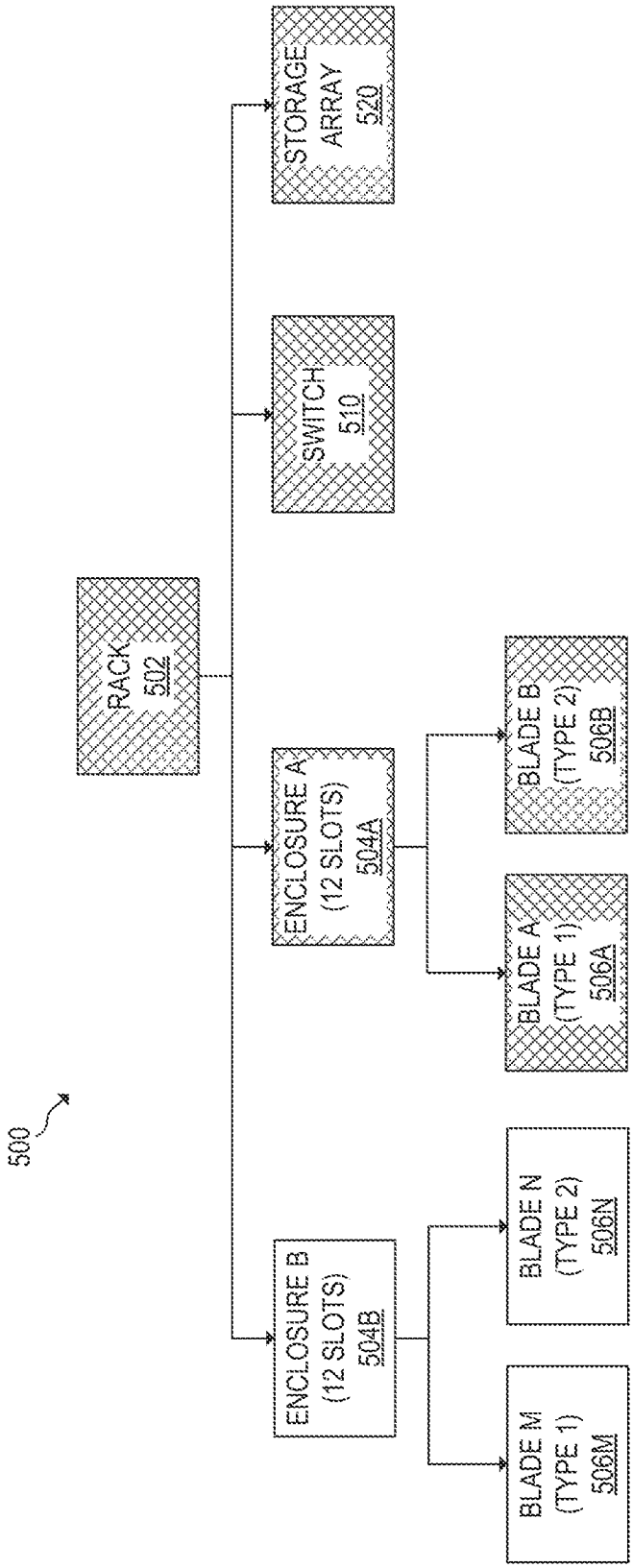

500
RACK 502
STORAGE ARRAY 520
SWITCH 510
ENCLOSURE A (12 SLOTS) 504A
BLADE B (TYPE 2) 506B
BLADE A (TYPE 1) 506A
ENCLOSURE B (12 SLOTS) 504B
BLADE N (TYPE 2) 506N
BLADE M (TYPE 1) 506M

FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2015/018053

A. CLASSIFICATION OF SUBJECT MATTER

G06F 15/76(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 15/76; G06F 15/16; G06F 19/00; G06F 9/44; G06F 7/00; H04L 12/28

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: scan, network, system, map, configuration, template.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2006-0092861 A1 (CHRISTOPHER CORDAY et al.) 04 May 2006 See abstract, paragraphs [0014]-[0053], claims 6,10,19,21-24 and figures 1-3.	1-15
Y	WO 2013-180703 A1 (HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P. et al.) 05 December 2013 See abstract, paragraphs [0014]-[0026], claim 1 and figures 1,2.	1-15
A	US 2007-0043525 A1 (AARON B. BROWN et al.) 22 February 2007 See abstract, paragraphs [0029]-[0053], claim 1 and figures 1,2.	1-15
A	US 2005-0038827 A1 (DAVID EUGENE HOOKS) 17 February 2005 See abstract, paragraphs [0036]-[0061], claim 1 and figures 1-3.	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
10 November 2015 (10.11.2015)	**11 November 2015 (11.11.2015)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea
Facsimile No. +82-42-472-7140

Authorized officer
Park Sungcheol
Telephone No. +82-42-481-5696

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/US2015/018053

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2006-0092861 A1	04/05/2006	EP 1782246 A2	09/05/2007
		EP 1782246 A4	20/08/2014
		US 2008-0228908 A1	18/09/2008
		US 9077611 B2	07/07/2015
		WO 2006-014504 A2	09/02/2006
		WO 2006-014504 A3	15/02/2007
WO 2013-180703 A1	05/12/2013	CN 104303147 A	21/01/2015
		EP 2856305 A1	08/04/2015
		JP 2015-522858 A	06/08/2015
		US 2015-0207682 A1	23/07/2015
US 2007-0043525 A1	22/02/2007	CN 101243322 A	13/08/2008
		CN 101243322 C	13/08/2008
		JP 2009-505274 A	05/02/2009
		US 2007-0043524 A1	22/02/2007
		US 7177774 B1	13/02/2007
		US 7472037 B2	30/12/2008
		WO 2007-021313 A1	22/02/2007
US 2005-0038827 A1	17/02/2005	CN 101882102 A	10/11/2010
		CN 1856781 A	01/11/2006
		CN 1856781 C	01/11/2006
		CN 1860476 A	08/11/2006
		CN 1860476 B	23/06/2010
		CN 1860476 C	08/11/2006
		EP 1661025 A2	31/05/2006
		EP 1661025 A4	26/05/2010
		EP 1661047 A2	31/05/2006
		JP 04886512 B2	29/02/2012
		JP 2007-502467 A	08/02/2007
		JP 2007-516495 A	21/06/2007
		US 2005-0038818 A1	17/02/2005
		US 2010-0005339 A1	07/01/2010
		US 2011-0145640 A1	16/06/2011
		US 2012-0102359 A1	26/04/2012
		US 2014-0365825 A1	11/12/2014
		US 7593936 B2	22/09/2009
		US 7908271 B2	15/03/2011
		US 8103664 B2	24/01/2012
		US 8819005 B2	26/08/2014
		WO 2005-017690 A2	24/02/2005
		WO 2005-017690 A3	23/03/2006
		WO 2005-020001 A2	03/03/2005
		WO 2005-020001 A3	18/08/2005