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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
2 February 2017 (02.02.2017)

WIPO | PCT

(10) International Publication Number
WO 2017/019006 Al

(51) International Patent Classification:
G06F 9/54 (2006.01)

(21) International Application Number:
PCT/US20 15/042 188

(22) International Filing Date:
27 July 2015 (27.07.2015)

(25) Filing Language: English

(26) Publication Language: English

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

(72) Inventor: **GATES, Matthew;** 11445 Compaq Center Drive West, Houston, Texas 77070 (US).

(74) Agents: **SORENSEN, C. Blake** 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 4.1 7(i))*

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.1 7(H))*

Published:

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

WO 2017/019006 A1

(54) Title: COMPONENT COMPATIBILITY VERIFICATION

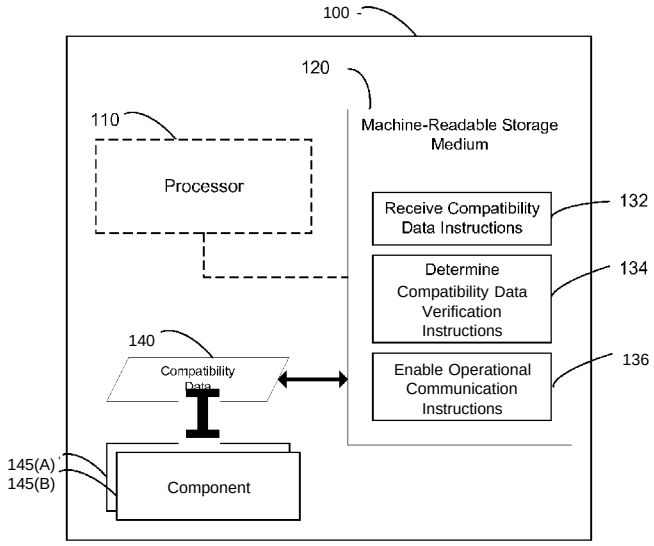

FIG. 1

(57) Abstract: Examples disclosed herein relate to compatibility verification instructions to receive a first element of compatibility data associated with a first component, determine whether the first element of compatibility data is verified according to a second component, determine whether a second element of compatibility data associated with a second component is verified according to the first component, and in response to determining that the first element of compatibility data is verified according to the second component and that the second element of compatibility data is verified according to the first component, enable operational communication between the first component and the second component.

COM PON ENT COM PATIBILITY VERIFICATION

BACKGROUND

[0001] Software components may provide different features and/or functionality to a complex operation. By using separate components, each may focus on a specialization and be maintained and updated separately rather than using a single, overly complex controller component to handle all aspects of an operation.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] In the accompanying drawings, like numerals refer to like components or blocks. The following detailed description references the drawings, wherein:

[0003] FIG. 1 is a block diagram of an example compatibility verification device;

[0004] FIG. 2 is a flowchart of an example of a method for providing compatibility verifications; and

[0005] FIG. 3 is a block diagram of an example system for providing compatibility verifications.

DETAILED DESCRIPTION

[0006] As described above, different software components may provide different functionalities to a complex operation. For example, a server may operate a controller, comprising a software stack of individual components. For a data storage controller, such a software stack may comprise, for example, a Redundant Array of Independent Disks (RAID) component, a cache management component, an encryption component, a third party plugin component, a replication component, a logging component, etc.

[0007] In multi-controller environments, the components of the software stack of one controller often communicate with the components of another controller to

coordinate operations. For example, a storage server with two controllers may replicate data to be stored to disk(s) managed by each of the controllers. If the components of each controller are incompatible, however, the operation may not be performed successfully. Compatibility, such as the ability to share a communication protocol between components, may be determined by sharing elements of compatibility data among respective components of each software stack to verify that the versions of each component can understand each other. The elements of compatibility data may also be shared across the components of the software stack in case a component of one type (e.g., a RAID component) needs to communicate with a component of another type (e.g., an encryption component). Controllers with incompatible components may be disabled and/or removed from an operation's communications to help avoid errors.

[0008] Referring now to the drawings, FIG. 1 is a block diagram of an example compatibility verification device 100 consistent with disclosed implementations. Compatibility verification device 100 may comprise a processor 110 and a non-transitory machine-readable storage medium 120. Compatibility verification device 100 may comprise a computing device such as a server computer, a desktop computer, a laptop computer, a handheld computing device, a smart phone, a tablet computing device, a mobile phone, a network device (e.g., a switch and/or router), or the like.

[0009] Processor 110 may comprise a central processing unit (CPU), a semiconductor-based microprocessor, a programmable component such as a complex programmable logic device (CPLD) and/or field-programmable gate array (FPGA), or any other hardware device suitable for retrieval and execution of instructions stored in machine-readable storage medium 120. In particular, processor 110 may fetch, decode, and execute a plurality of provide compatibility data instructions 130, receive compatibility data instructions 132, determine compatibility data verification instructions 134, and enable operational communication instructions 136 to implement the functionality described in detail below.

[001 0] Executable instructions may comprise logic stored in any portion and/or component of machine-readable storage medium 120 and executable by processor 110. The machine-readable storage medium 120 may comprise both volatile and/or nonvolatile memory and data storage components. Volatile components are those that do not retain data values upon loss of power. Nonvolatile components are those that retain data upon a loss of power.

[001 1] The machine-readable storage medium 120 may comprise, for example, random access memory (RAM), read-only memory (ROM), hard disk drives, solid-state drives, USB flash drives, memory cards accessed via a memory card reader, floppy disks accessed via an associated floppy disk drive, optical discs accessed via an optical disc drive, magnetic tapes accessed via an appropriate tape drive, and/or other memory components, and/or a combination of any two and/or more of these memory components. In addition, the RAM may comprise, for example, static random access memory (SRAM), dynamic random access memory (DRAM), and/or magnetic random access memory (MRAM) and other such devices. The ROM may comprise, for example, a programmable read-only memory (PROM), an erasable programmable read-only memory (EPROM), an electrically erasable programmable read-only memory (EEPROM), and/or other like memory device.

[001 2] Receive compatibility data instructions 132 may receive a first element of compatibility data 140 associated with a first component 145(A). For example, compatibility data 140 may comprise a software version, a list of supported data formats and/or supported protocols, localization settings, tolerance requirements, available resources such usable as communication ports and/or storage space, etc.

[0013] Determine compatibility data verification instructions 134 may determine whether the first element of compatibility data 140 is verified according to a second component 145(B) and may determine whether a second element of compatibility data (not shown) associated with the second component 145(B) is verified according to the first component 145(A). For example, components 145(A)-(B) may comprise corresponding RAID controllers in a pair of storage controllers'

software stacks. The compatibility data for each RAID controller may be examined by the corresponding controller in the other stack to determine if the software version of the other component is compatible. In some implementations, for example, compatibility data 140 may comprise a version of component 145(A) and the instructions 134 to determine whether the first element of compatibility data 140 is verified according to the second component 145(B) may comprise determining whether the version associated with the first component 145(A) comprises a later version than a version associated with the second component 145(B). Such a determination may rely, for example, on a version number, a date stamp of the component, a text string, and/or other data elements.

[0014] Compatibility may be restricted by a communication protocol format, the presence of required resources such as sufficient memory, and/or security requirements such as access to a particular encryption/decryption key and/or supporting a certain level of encryption on exchanged data. Receive compatibility data instructions 132 may provide for the exchange of compatibility data 140 directly between the components 145(A)-(B) and/or compatibility data may be collected from a plurality of components associated with each controller and provided in bulk to the other controller(s) to distribute among their respective components.

[0015] In some implementations, a determination of compatibility may be deferred to the later version among the components 145(A)-(B). For example, if second component 145(B) comprises a version number of 2.1 and first component 145(A) comprises a version number of 2.0, the compatibility determination of second component 145(B) may overrule a verification of compatibility with first component 145(A).

[001 6] In some implementations, determine compatibility data verification instructions 134 may determine whether a new version of the second component has been installed. In response to determining that the new version of the second component has been installed, the element of compatibility data associated with the second component may be provided to the first component to determine

whether the new version of the second component may be verified as compatible with the first component.

[001 7] Enable operational communication instructions 136 may, in response to determining that the first element of compatibility data 140 is verified according to the second component 145(B) and that the second element of compatibility data is verified according to the first component 145(A), enable operational communication between the first component 145(A) and the second component 145(B). In response to determining that the first element of compatibility data 140 is not verified according to the second component 145(B), enable operational communication instructions 136 may disable operational communication between the first component 145(A) and the second component 145(B). When operational communication between the components 145(A)-(B) is enabled, the two controllers associated with components 145(A)-(B) may communicate and cooperate to complete operations. For example, two storage controllers may perform cooperative data replication and/or de-duplication operations. When operational communication between the components 145(A)-(B) is disabled, the two controllers associated with components 145(A)-(B) may be prevented from communication in the performance of such operations, and the operation may be performed by a single controller and/or only those controllers whose components have been verified as compatible.

[0018] FIG. 2 is a flowchart of an example method 200 for compatibility verification consistent with disclosed implementations. Although execution of method 200 is described below with reference to the components of compatibility verification device 100, other suitable components for execution of method 200 may be used.

[0019] Method 200 may begin in block 205 and proceed to block 210 where device 100 may exchange a plurality of compatibility data between a first component and a second component. In some implementations, the first component and the second component may comprise respective software components of a first software stack and a second software stack, such as corresponding RAID components of two storage controller's stacks.

[0020] For example, receive compatibility data instructions 132 may receive a first element of compatibility data 140 associated with a first component 145(A) and/or a second element of compatibility data associated with a second component 145(B). For example, compatibility data 140 may comprise a software version, a list of supported data formats and/or supported protocols, localization settings, tolerance requirements, available resources such usable as communication ports and/or storage space, etc.

[0021] Method 200 may then advance to block 215 where device 100 may determine whether the first component and the second component are compatible according to the exchanged plurality of compatibility data. For example, determine compatibility data verification instructions 134 may determine whether the first element of compatibility data 140 is verified according to a second component 145(B) and may determine whether a second element of compatibility data (not shown) associated with the second component 145(B) is verified according to the first component 145(A). For example, components 145(A)-(B) may comprise corresponding RAID controllers in a pair of storage controllers' software stacks. The compatibility data for each RAID controller may be examined by the corresponding controller in the other stack to determine if the software version of the other component is compatible. In some implementations, for example, compatibility data 140 may comprise a version of component 145(A) and the instructions 134 to determine whether the first element of compatibility data 140 is verified according to the second component 145(B) may comprise determining whether the version associated with the first component 145(A) comprises a later version than a version associated with the second component 145(B). Such a determination may rely, for example, on a version number, a date stamp of the component, a text string, and/or other data elements.

[0022] Compatibility may be restricted by a communication protocol format, the presence of required resources such as sufficient memory, and/or security requirements such as access to a particular encryption/decryption key and/or supporting a certain level of encryption on exchanged data. Receive compatibility

data instructions 132 may provide for the exchange of compatibility data 140 directly between the components 145(A)-(B) and/or compatibility data may be collected from a plurality of components associated with each controller and provided in bulk to the other controller(s) to distribute among their respective components.

[0023] In some implementations, a determination of compatibility may be deferred to the later version among the components 145(A)-(B). For example, if second component 145(B) comprises a version number of 2.1 and first component 145(A) comprises a version number of 2.0, the compatibility determination of second component 145(B) may overrule a verification of compatibility with first component 145(A).

[0024] In some implementations, determine compatibility data verification instructions 134 may determine whether a new version of the second component has been installed. In response to determining that the new version of the second component has been installed, the element of compatibility data associated with the second component may be provided to the first component to determine whether the new version of the second component may be verified as compatible with the first component.

[0025] In some implementations, a further determination may be made by device 100 at block 215 as to whether at least one third component associated with the first software stack is compatible with the second component according to the exchanged plurality of compatibility data. For example, component 245(A) may comprise a RAID component and may further determine whether compatibility data from a corresponding component of another controller is also compatible with a cache component of the controller for component 245(A).

[0026] In some implementations, determining whether the first component and the second component are compatible may comprise determining whether the first component and the second component comprise a compatible communication protocol. For example, software components may define a specific message format to exchange data with each other. New versions of the format may include

additional fields and/or properties that may not be understood by older versions of the component. In some implementations, newer versions of the component may modify the communication protocol between the first component and the second component to be compatible with the older version, such as by reverting to an older version of the messaging format and/or stripping out unknown fields.

[0027] In response to determining that the first component and the second component are not compatible, method 200 may advance to block 220 where device 100 may disable communication with the second component, wherein the second component comprises an older version of the first component. For example, in response to determining that the first element of compatibility data 140 is not verified according to the second component 145(B), enable operational communication instructions 136 may disable operational communication between the first component 145(A) and the second component 145(B). When operational communication between the components 145(A)-(B) is disabled, the two controllers associated with components 145(A)-(B) may be prevented from communication in the performance of such operations, and the operation may be performed by a single controller and/or only those controllers whose components have been verified as compatible. In some implementations, in response to determining that the first component and the second component are not compatible, device 100 may upgrade the second component to a compatible version with the first component. For example, the version of the component from a first controller may be copied to and installed on the second controller. For another example, an update server may be contacted to download a new version of the second component.

[0028] In response to determining that the first component and the second component are compatible, method 200 may advance to block 226 where device 100 may enable communication between the first component and the second component. For example, enable operational communication instructions 136 may, in response to determining that the first element of compatibility data 140 is verified according to the second component 145(B) and that the second element of

compatibility data is verified according to the first component 145(A), enable operational communication between the first component 145(A) and the second component 145(B). When operational communication between the components 145(A)-(B) is enabled, the two controllers associated with components 145(A)-(B) may communicate and cooperate to complete operations. For example, two storage controllers may perform cooperative data replication and/or de-duplication operations.

[0029] Method 200 may then end at block 250.

[0030] FIG. 3 is a block diagram of an example system 300 for providing compatibility verification. System 300 may comprise a computing device 310 comprising a first controller engine 315 that may comprise a software stack of a first plurality of components 320(A)-(C) and a second controller engine 325 that may comprise a software stack of a second plurality of components 330(A)-(C). Device 310 may further comprise a compatibility engine 340. Although only one device 310 is pictured in system 300, other devices and/or other controller engines may be present and/or in communication with device 310 and controller engines 315, 325.

[0031] Computing device 310 may comprise, for example, a general and/or special purpose computer, server, mainframe, desktop, laptop, tablet, smart phone, game console, and/or any other system capable of providing computing capability consistent with providing the implementations described herein. Computing device 410 may comprise any combination of hardware and programming to implement the functionalities of the respective components. In examples described herein, such combinations of hardware and programming may be implemented in a number of different ways. For example, programming may comprise processor executable instructions stored on a non-transitory machine-readable storage medium and the hardware for the engines may include a processing resource to execute those instructions.

[0032] Controller engine 315 may provide a first plurality of compatibility data associated with a first set of software components 320(A)-(C) to a second controller engine and receive a second plurality of compatibility data associated

with a second set of software components 330(A)-(C) from the second controller engine. The compatibility data may be exchanged directly between respective components, aggregated and exchanged between controller engines 315, 325 and/or via compatibility engine 340.

[0033] Compatibility engine 340 may provide the received second plurality of compatibility data to each of the first set of software components. For example, controller engine 325 may provide the compatibility data associated with components 330(A)-(C) to compatibility engine 340. Compatibility engine 340 may then distribute this compatibility data to controller engine 315, to a respective component of components 320(A)-(C), and/or to each of components 320(A)-(C). Thus, in some implementations, compatibility data for a RAID component of controller engine 325 (e.g., component 330(A)) may be provided to the RAID component of controller engine 315 (e.g., component 320(A)).

[0034] Compatibility engine 340 may determine whether each of the first set of software components 320(A)-(C) comprises a compatible communication protocol with the second set of software components 330(A)-(C). In response to determining that at least one of the first set of software components does not comprise a compatible communication protocol with the second set of software components, compatibility engine 340 may determine whether a modified communication protocol is available, and in response to determining that a modified communication protocol is not available, disable communication between the first controller engine 315 and the second controller engine 325.

[0035] In some implementations, in order to disable communication between the first controller engine and the second controller engine, compatibility engine 340 may identify which of the first controller engine 315 and the second controller engine 325 comprises an older version of the at least one of the first set of software components 320(A)-(C) that does not comprise the compatible communication protocol with the second set of software components 325(A)-(C). The controller engine comprising the older version may be the one disabled from communication by compatibility engine 340.

[0036] The disclosed examples may include systems, devices, computer-readable storage media, and methods for compatibility verification. For purposes of explanation, certain examples are described with reference to the components illustrated in the Figures. The functionality of the illustrated components may overlap, however, and may be present in a fewer or greater number of elements and components. Further, all or part of the functionality of illustrated elements may co-exist or be distributed among several geographically dispersed locations. Moreover, the disclosed examples may be implemented in various environments and are not limited to the illustrated examples.

[0037] Moreover, as used in the specification and the appended claims, the singular forms "a," "an," and "the" are intended to include the plural forms as well, unless the context indicates otherwise. Additionally, although the terms first, second, etc. may be used herein to describe various elements, these elements should not be limited by these terms. Instead, these terms are only used to distinguish one element from another.

[0038] Further, the sequence of operations described in connection with the Figures are examples and are not intended to be limiting. Additional or fewer operations or combinations of operations may be used or may vary without departing from the scope of the disclosed examples. Thus, the present disclosure merely sets forth possible examples of implementations, and many variations and modifications may be made to the described examples. All such modifications and variations are intended to be included within the scope of this disclosure and protected by the following claims.

CLAIMS

We claim:

1. A non-transitory machine-readable storage medium comprising instructions to:

receive a first element of compatibility data associated with a first component;

determine whether the first element of compatibility data is verified according to a second component;

determine whether a second element of compatibility data associated with the second component is verified according to the first component; and

in response to determining that the first element of compatibility data is verified according to the second component and that the second element of compatibility data is verified according to the first component, enable operational communication between the first component and the second component.

2. The non-transitory machine-readable medium of claim 1, wherein the first element of compatibility data comprises a first component version.

3. The non-transitory machine-readable medium of claim 2, wherein the instructions to determine whether the first element of compatibility data is verified according to the second component comprises determining whether the first component version comprises a later version than a second component version.

4. The non-transitory machine-readable medium of claim 3, wherein the instructions, in response to determining that the first component version comprises a later version than a second component version, further comprise instructions to defer to the determination of whether the second element of compatibility data is verified according to the first component.

5. The non-transitory machine-readable medium of claim 1, wherein the instructions, in response to determining that the first element of compatibility data is not verified according to the second component, further comprise instructions to disable operational communication between the first component and the second component.

6. The non-transitory machine-readable medium of claim 5, wherein the instructions to receive the second element of compatibility data associated with the second component further comprise instructions to:

determine whether a new version of the second component has been installed; and

in response to determining that the new version of the second component has been installed, provide the second element of compatibility data associated with the new version of the second component to the first component.

7. The non-transitory machine-readable medium of claim 1, further comprising instructions to:

determine whether the second element of compatibility data associated with the second component is verified according to a third component in communication with the first component.

8. A computer-implemented method, comprising:

exchanging a plurality of compatibility data between a first component and a second component;

determining whether the first component and the second component are compatible according to the exchanged plurality of compatibility data;

in response to determining that the first component and the second component are not compatible, disabling communication with the second component, wherein the second component comprises an older version of the first component;

in response to determining that the first component and the second component are compatible, enabling communication between the first component and the second component.

9. The computer-implemented method of claim 8, wherein the first component and the second component comprise respective software components of a first software stack and a second software stack.

10. The computer-implemented method of claim 9, further comprising determining whether at least one third component associated with the first software stack is compatible with the second component according to the exchanged plurality of compatibility data.

11. The computer-implemented method of claim 8, wherein determining whether the first component and the second component are compatible comprises determining whether the first component and the second component comprise a compatible communication protocol.

12. The computer-implemented method of claim 11, wherein in response to determining that the first component and the second component are not compatible, modifying a communication protocol between the first component and the second component.

13. The computer-implemented method of claim 8, wherein in response to determining that the first component and the second component are not compatible, upgrading the second component to a compatible version with the first component.

14. A system, comprising:

a first controller engine to:

provide a first plurality of compatibility data associated with a first set of software components to a second controller engine, and

receive a second plurality of compatibility data associated with a second set of software components from the second controller engine; and

a compatibility engine to:

provide the received second plurality of compatibility data to each of the first set of software components,

determine whether each of the first set of software components comprises a compatible communication protocol with the second set of software components,

in response to determining that at least one of the first set of software components does not comprise a compatible communication protocol with the second set of software components, determine whether a modified communication protocol is available, and

in response to determining that a modified communication protocol is not available, disable communication between the first controller engine and the second controller engine.

15. The system of claim 14, wherein to disable communication between the first controller engine and the second controller engine comprises the compatibility engine to identify which of the first controller engine and the second controller engine comprises an older version of the at least one of the first set of software components that does not comprise the compatible communication protocol with the second set of software components.

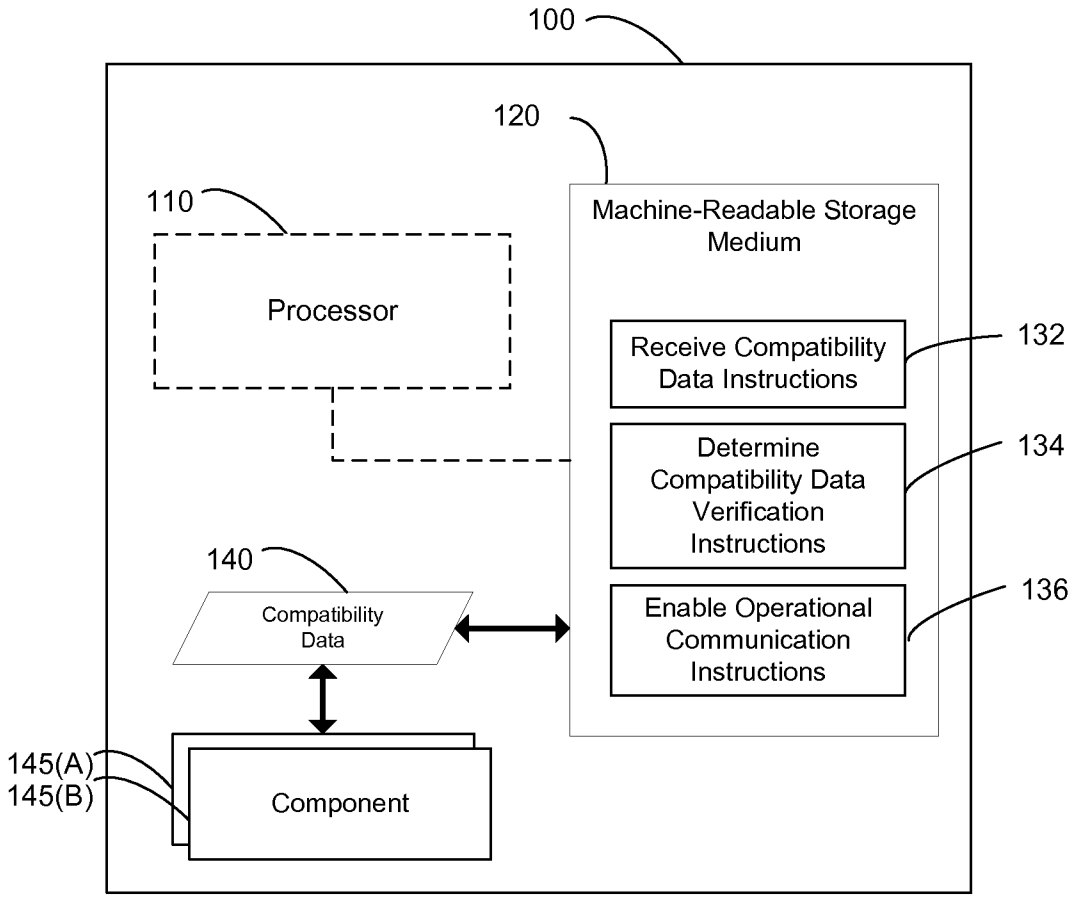
100
120
110
Machine-Readable Storage Medium
Processor
Receive Compatibility Data Instructions
132
Determine Compatibility Data Verification Instructions
134
140
Compatibility Data
Enable Operational Communication Instructions
136
145(A)
145(B)
Component

FIG. 1

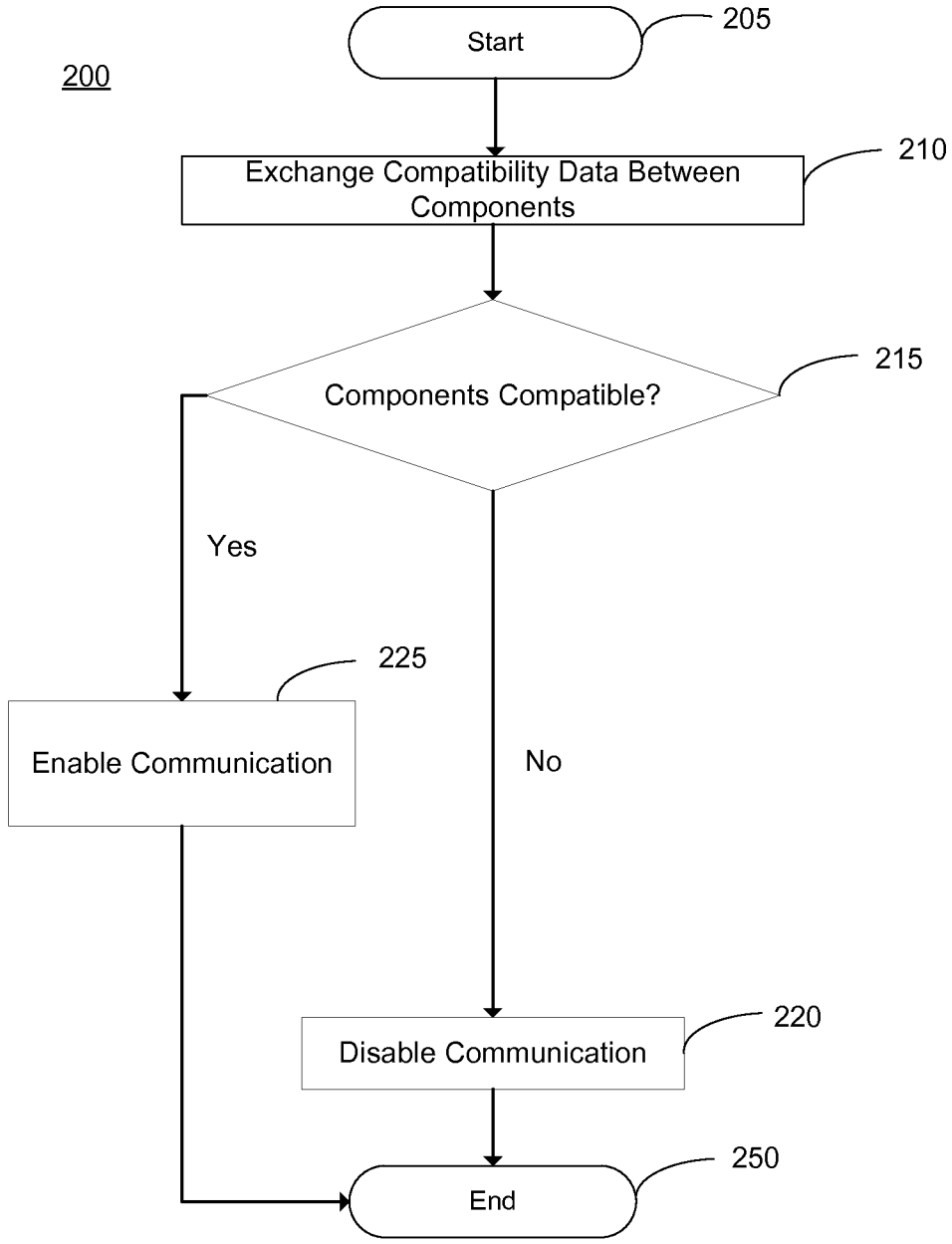
200
Start
205
Exchange Compatibility Data Between Components
210
Components Compatible?
215
Yes
No
Enable Communication
225
Disable Communication
220
End
250

FIG. 2

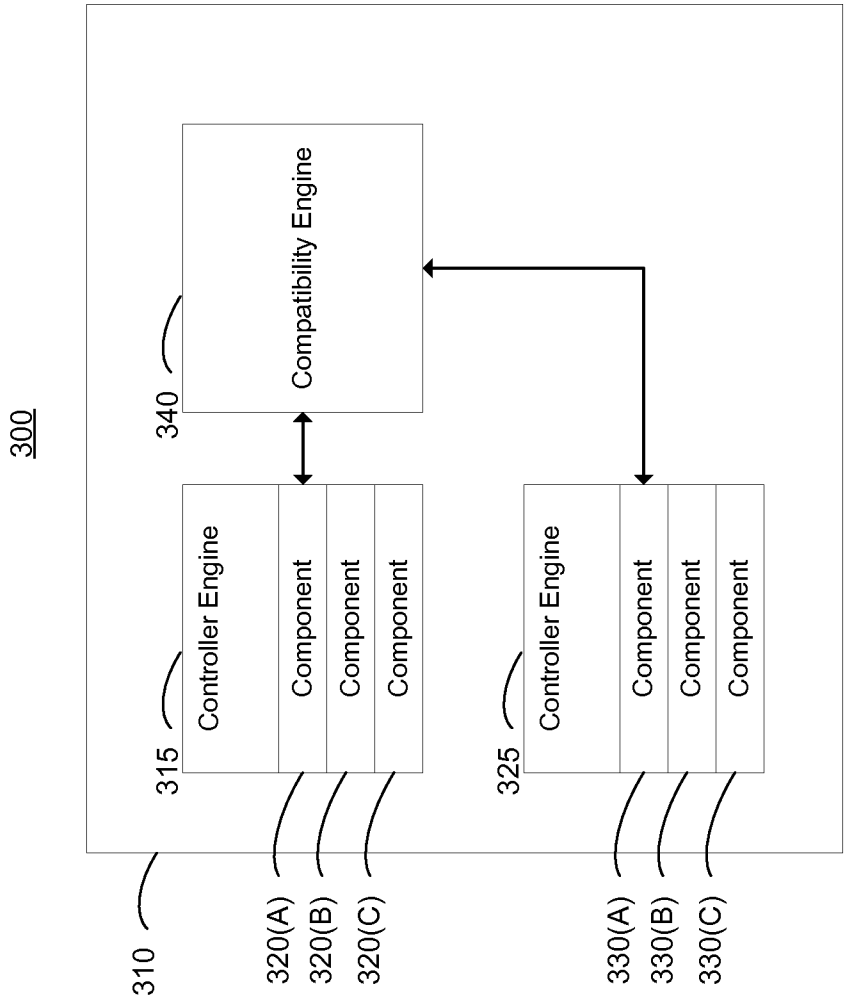
300
Compatibility Engine
340
Controller Engine
315
Component
Component
Component
Controller Engine
325
Component
Component
Component
310
320(A)
320(B)
320(C)
330(A)
330(B)
330(C)

FIG. 3

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2015/042188

A. CLASSIFICATION OF SUBJECT MATTER

G06F 9/54(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 9/54; G06F 9/445; G06F 9/44; G06F 9/455; G06F 13/00; G06F 9/45

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: compatibility, verification, first/second component, enable/disable, exchanging, older version, new version, software, protocol, upgrading

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 5579509 A (DAVID A. FURTNEY et al.) 26 November 1996 See column 1, lines 23-29; column 2, lines 32-48; column 3, line 52 - column 4, line 3; column 6, lines 1-6; claims 1, 4-8; and figure 2.	1-15
Y	US 2006-0265696 A1 (KEVIN B. MAYFIELD et al.) 23 November 2006 See abstract; paragraphs [0016], [0026]; claims 13, 15-16; and figure 1.	1-15
A	US 2011-0321029 A1 (JONATHAN FRED KERN et al.) 29 December 2011 See paragraphs [0040]-[0042]; claims 1-5, 9; and figures 5-6.	1-15
A	US 2004-0181790 A1 (JOSEPH W. HERRICK) 16 September 2004 See paragraphs [0055]-[0057]; claims 2, 14; and figures 8-9.	1-15
A	US 2013-0060558 A1 (JASON SCHULTZ et al.) 07 March 2013 See paragraphs [0020]-[0025]; and figure 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
22 April 2016 (22.04.2016)	**22 April 2016 (22.04.2016)**

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. +82-42-481-8578

Authorized officer
CHIN, Sang Bum
Telephone No. +82-42-481-8398

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/US2015/042188

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 5579509 A	26/11/1996	EP 0498130 A2 JP 05-088859 A	12/08/1992 09/04/1993
US 2006-0265696 Al	23/11/2006	US 7861239 B2	28/12/2010
US 2011-0321029 Al	29/12/2011	US 8689208 B2	01/04/2014
US 2004-0181790 Al	16/09/2004	WO 2004-081758 A2 WO 2004-081758 A3	23/09/2004 24/03/2005
US 2013-0060558 Al	07/03/2013	None	