



(51) International Patent Classification:

H04L 41/082 (2022.01) *G06F 8/65* (2018.01)

(21) International Application Number:

PCT/IN2024/051250

(22) International Filing Date:

15 July 2024 (15.07.2024)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

202321048114 17 July 2023 (17.07.2023) IN

(71) Applicant: **JIO PLATFORMS LIMITED** [IN/IN]; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN).

(72) Inventors: -, **Sundaresh**; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad -

380006, Gujarat, Ahmedabad 380006 (IN). **BHATNAGAR, Aayush**; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN). **BHATNAGAR, Pradeep Kumar**; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN). **POLSONI, Chaitanya Reddy**; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN). **KUMAR, Mukesh**; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN). **PATEL, Vivek**; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN). **TOMAR, Mahendra**; Office - 101, Saffron, Nr. Centre Point, Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN). **WADHWANI, Karan**; Office - 101, Saffron, Nr. Centre Point,

(54) Title: METHOD AND SYSTEM FOR AUTOMATIC UPGRADE OF SOFTWARE FOR NETWORK ELEMENTS

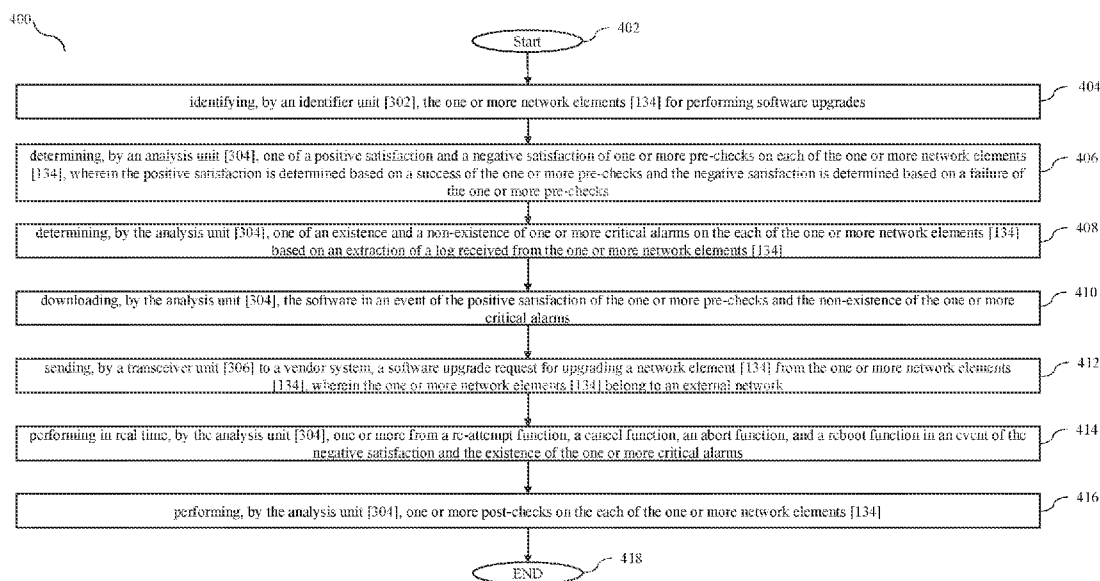


FIG. 4

(57) Abstract: The present disclosure relates to a method and a system for automatic upgrade of software for one or more network elements [134]. The present disclosure encompasses: identifying the one or more network elements [134] for performing software upgrades; determining positive or negative satisfaction of one or more pre-checks and existence or non-existence of one or more critical alarms; downloading the software if positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms; sending to a vendor system, a software upgrade request for upgrading a network element [134] from the one or more network elements [134]; performing in real time a re-attempt function, a cancel function, an abort function, and a reboot function if negative satisfaction and the existence of the one or more critical alarms; and performing one or more post-checks on the each of the one or more network elements [134].



Panchwati 5 Rasta, Ambawadi, Ahmedabad - 380006, Gujarat, Ahmedabad 380006 (IN).

(74) **Agent:** SAHNEY, Garima; SAIKRISHNA & ASSOCIATES ADVOCATES, B-140, Sector 51, Noida-201301, Uttar Pradesh, India, Noida 201301 (IN).

(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, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, 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, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW.

(84) **Designated States** (*unless otherwise indicated, for every kind of regional protection available*): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, 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, ME, 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 applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*
- *of inventorship (Rule 4.17(iv))*

Published:

- *with international search report (Art. 21(3))*
- *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

METHOD AND SYSTEM FOR AUTOMATIC UPGRADE OF SOFTWARE FOR NETWORK ELEMENTS

FIELD OF INVENTION

5

[0001] Embodiments of the present disclosure generally relate to software upgradation for one or more network elements. More particularly, embodiments of the present disclosure relate to a method and system for an automatic upgrade of a software for the one or more network elements.

10 BACKGROUND

[0002] The following description of the related art is intended to provide background information pertaining to the field of the disclosure. This section may include certain aspects of the art that may be related to various features of the present disclosure. However, it should be appreciated that
15 this section is used only to enhance the understanding of the reader with respect to the present disclosure, and not as admissions of the prior art.

[0003] Wireless communication technology has rapidly evolved over the past few decades, with each generation bringing significant improvements and advancements. The first generation of
20 wireless communication technology was based on analog technology and offered only voice services. However, with the advent of the second-generation (2G) technology, digital communication and data services became possible, and text messaging was introduced. The third-generation (3G) technology marked the introduction of high-speed internet access, mobile video calling, and location-based services. The fourth-generation (4G) technology revolutionized
25 wireless communication with faster data speeds, better network coverage, and improved security. Currently, the fifth-generation (5G) technology is being deployed, promising even faster data speeds, low latency, and the ability to connect multiple devices simultaneously. With each generation, wireless communication technology has become more advanced, sophisticated, and capable of delivering more services to its users.

30

[0004] Various solutions have been developed to upgrade software and firmware of wired and wireless communication network elements. The conventional and existing solution technologies include manual software upgrade and manual firmware upgrade of network elements which require multiple set of steps to execute on network element system(s). Such execution may be done
35 via manual login to a corresponding management system which handles minimal number of

network elements. This manual software or firmware upgrade of network elements could take months to upgrade all the nodes of network to the target software version.

[0005] Therefore, there is a need in the art to provide a system and method for an automatic upgrade of a software for one or more network elements and upgrade, at scale, the firmware in the network without human interference.

SUMMARY

[0006] This section is provided to introduce certain aspects of the present disclosure in a simplified form that are further described below in the detailed description. This summary is not intended to identify the key features or the scope of the claimed subject matter.

[0007] An aspect of the present disclosure relates to a method for an automatic upgrade of a software for one or more network elements. The method comprises identifying, by an identifier unit, one or more network elements for performing software upgrades. The method further comprises determining, by an analysis unit, one of a positive satisfaction and a negative satisfaction of one or more pre-checks on each of the one or more network elements, wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks. The method further comprises determining, by the analysis unit, one of an existence and a non-existence of one or more critical alarms on said each of the one or more network elements based on an extraction of a log received from the one or more network elements. The method further comprises downloading, by the analysis unit, the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms. The method further comprises sending, by a transceiver unit to a vendor system, a software upgrade request for upgrading a network element from the one or more network elements, wherein the one or more network elements belong to an external network. The method further comprises performing in real time, by the analysis unit, one or more from a re-attempt function, a cancel function, an abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms. The method further comprises performing, by the analysis unit, one or more post-checks on said each of the one or more network elements.

[0008] In an exemplary aspect of the present disclosure, the one or more pre-checks include a memory availability, a central processing unit (CPU) utilization, a health check, a system requirement check, and a backup configuration check.

5 [0009] In another exemplary aspect of the present disclosure, the one or more post-checks comprises one or more of a software version check, and a performance metrics check.

[0010] In another exemplary aspect of the present disclosure, the performance metrics check comprises measuring one or more key performance metrics comprising a throughput metric, a
10 latency metric, and a capacity metric.

[0011] In another exemplary aspect of the present disclosure, the upgrading of the network element from the one or more network elements is performed based on a stage associated with said
15 network element.

[0012] In another exemplary aspect of the present disclosure, in the event of the upgrading of the network element from the one or more network elements, a coverage area associated with said
network element is assigned to one or more other network elements.

20 [0013] Another aspect of the present disclosure relates to a system for an automatic upgrade of a software for one or more network elements. The system comprises an identifier unit, an analysis unit, and a transceiver unit connected to each other. The identifier unit is configured to identify one or more network elements for performing software upgrades. The analysis unit is configured to determine one of a positive satisfaction and a negative satisfaction of one or more pre-checks
25 on each of the one or more network elements, wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks. The analysis unit is further configured to determine one of an existence and a non-existence of one or more critical alarms on said each of the one or more network elements based on an extraction of a log received from the one or more network elements.
30 The analysis unit is further configured to download the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms. The transceiver unit is configured to send to a vendor system, a software upgrade request for upgrading a network element from the one or more network elements, wherein the one or more network elements belong to an external network. The analysis unit is further configured to perform
35 in real time, one or more from a re-attempt function, a cancel function, an abort function, and a

reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms. The analysis unit is further configured to perform one or more post-checks on said each of the one or more network elements.

5 **[0014]** Yet another aspect of the present disclosure relates to a non-transitory computer readable storage medium storing instructions for an automatic upgrade of a software for one or more network elements, the instructions include executable code which, when executed by one or more units of a system, causes: an identifier unit to identify one or more network elements for performing software upgrades; an analysis unit to determine one of a positive satisfaction and a
10 negative satisfaction of one or more pre-checks on each of the one or more network elements, wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks; the analysis unit to determine one of an existence and a non-existence of one or more critical alarms on the each of the one or more network elements based on an extraction of a log received from the
15 one or more network elements; the analysis unit to download the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms; a transceiver unit to send to a vendor system, a software upgrade request for upgrading a network element from the one or more network elements, wherein the one or more network elements belong to an external network; the analysis unit to perform in real time one or more from
20 a re-attempt function, a cancel function, an abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms; and the analysis unit to perform one or more post-checks on the each of the one or more network elements.

OBJECTS OF THE DISCLOSURE

25 **[0015]** Some of the objects of the present disclosure, which at least one embodiment disclosed herein satisfies are listed herein below.

30 **[0016]** It is an object of the present disclosure to provide a system and a method for an automatic upgrade of a software for one or more network elements.

[0017] It is an object of the present disclosure to provide a system and a method for automatic multi-generation multiple vendor global scale software upgrade.

[0018] It is another object of the present disclosure to monitor the progress in near real time on dynamic dashboard at multiple levels.

[0019] It is yet another object of the present disclosure to to apply configuration changes dynamically on timers for triggers, reboot, parameter changes, and alarms.

[0020] It is yet another object of the present disclosure to revert upgrade.

[0021] It is yet another object of the present disclosure to address network element level pre-checks and network element level post-checks.

DESCRIPTION OF THE DRAWINGS

[0022] The accompanying drawings, which are incorporated herein, and constitute a part of this disclosure, illustrate exemplary embodiments of the disclosed method and system in which like reference numerals refer to the same parts throughout the different drawings. Components in the drawings are not necessarily to scale, emphasis instead being placed upon clearly illustrating the principles of the present disclosure. Also, the embodiments shown in the figures are not to be construed as limiting the disclosure, but the possible variants of the method and system according to the disclosure are illustrated herein to highlight the advantages of the disclosure. It will be appreciated by those skilled in the art that disclosure of such drawings include disclosure of electrical components or circuitry commonly used to implement such components.

[0023] FIG. 1 illustrates an exemplary block diagram representation of a system architecture for performing software upgrades of the one or more network elements, in accordance with an exemplary implementation of the present disclosure.

[0024] FIG. 2 illustrates an exemplary block diagram of a computing device upon which the features of the present disclosure may be implemented in accordance with exemplary implementation of the present disclosure.

[0025] FIG. 3 illustrates an exemplary block diagram of a system for an automatic upgrade of a software for one or more network elements, in accordance with exemplary implementations of the present disclosure.

[0026] FIG. 4 illustrates a method flow diagram for an automatic upgrade of a software for one or more network elements in accordance with an exemplary implementation of the present disclosure.

[0027] FIG. 5 illustrates a method flow diagram for an automatic upgrade of a software for one or more network elements in accordance with another exemplary implementation of the present disclosure.

[0028] The foregoing shall be more apparent from the following more detailed description of the disclosure.

DETAILED DESCRIPTION

[0029] In the following description, for the purposes of explanation, various specific details are set forth in order to provide a thorough understanding of embodiments of the present disclosure.

It will be apparent, however, that embodiments of the present disclosure may be practiced without these specific details. Several features described hereafter may each be used independently of one another or with any combination of other features. An individual feature may not address any of the problems discussed above or might address only some of the problems discussed above.

[0030] The ensuing description provides exemplary embodiments only, and is not intended to limit the scope, applicability, or configuration of the disclosure. Rather, the ensuing description of the exemplary embodiments will provide those skilled in the art with an enabling description for implementing an exemplary embodiment. It should be understood that various changes may be made in the function and arrangement of elements without departing from the spirit and scope of the disclosure as set forth.

[0031] Specific details are given in the following description to provide a thorough understanding of the embodiments. However, it will be understood by one of ordinary skill in the art that the embodiments may be practiced without these specific details. For example, circuits, systems, processes, and other components may be shown as components in block diagram form in order not to obscure the embodiments in unnecessary detail.

[0032] Also, it is noted that individual embodiments may be described as a process which is depicted as a flowchart, a flow diagram, a data flow diagram, a structure diagram, or a block diagram. Although a flowchart may describe the operations as a sequential process, many of the

operations may be performed in parallel or concurrently. In addition, the order of the operations may be re-arranged. A process is terminated when its operations are completed but could have additional steps not included in a figure.

- 5 **[0033]** The word “exemplary” and/or “demonstrative” is used herein to mean serving as an example, instance, or illustration. For the avoidance of doubt, the subject matter disclosed herein is not limited by such examples. In addition, any aspect or design described herein as “exemplary” and/or “demonstrative” is not necessarily to be construed as preferred or advantageous over other aspects or designs, nor is it meant to preclude equivalent exemplary structures and techniques
- 10 known to those of ordinary skill in the art. Furthermore, to the extent that the terms “includes,” “has,” “contains,” and other similar words are used in either the detailed description or the claims, such terms are intended to be inclusive—in a manner similar to the term “comprising” as an open transition word—without precluding any additional or other elements.
- 15 **[0034]** As used herein, a “processing unit” or “processor” or “operating processor” includes one or more processors, wherein processor refers to any logic circuitry for processing instructions. A processor may be a general-purpose processor, a special purpose processor, a conventional processor, a digital signal processor, a plurality of microprocessors, one or more microprocessors in association with a Digital Signal Processing (DSP) core, a controller, a microcontroller,
- 20 Application Specific Integrated Circuits, Field Programmable Gate Array circuits, any other type of integrated circuits, etc. The processor may perform signal coding data processing, input/output processing, and/or any other functionality that enables the working of the system according to the present disclosure. More specifically, the processor or processing unit is a hardware processor.
- 25 **[0035]** As used herein, “a user equipment”, “a user device”, “a smart-user-device”, “a smart-device”, “an electronic device”, “a mobile device”, “a handheld device”, “a wireless communication device”, “a mobile communication device”, “a communication device” may be any electrical, electronic and/or computing device or equipment, capable of implementing the features of the present disclosure. The user equipment/device may include, but is not limited to, a
- 30 mobile phone, smart phone, laptop, a general-purpose computer, desktop, personal digital assistant, tablet computer, wearable device or any other computing device which is capable of implementing the features of the present disclosure. Also, the user device may contain at least one input means configured to receive an input from unit(s) which are required to implement the features of the present disclosure.

35

[0036] As used herein, “storage unit” or “memory unit” refers to a machine or computer-readable medium including any mechanism for storing information in a form readable by a computer or similar machine. For example, a computer-readable medium includes read-only memory (“ROM”), random access memory (“RAM”), magnetic disk storage media, optical storage media, flash memory devices or other types of machine-accessible storage media. The storage unit stores at least the data that may be required by one or more units of the system to perform their respective functions.

[0037] As used herein “interface” or “user interface refers to a shared boundary across which two or more separate components of a system exchange information or data. The interface may also be referred to a set of rules or protocols that define communication or interaction of one or more modules or one or more units with each other, which also includes the methods, functions, or procedures that may be called.

[0038] All modules, units, components used herein, unless explicitly excluded herein, may be software modules or hardware processors, the processors being a general-purpose processor, a special purpose processor, a conventional processor, a digital signal processor (DSP), a plurality of microprocessors, one or more microprocessors in association with a DSP core, a controller, a microcontroller, Application Specific Integrated Circuits (ASIC), Field Programmable Gate Array circuits (FPGA), any other type of integrated circuits, etc.

[0039] As used herein the transceiver unit include at least one receiver and at least one transmitter configured respectively for receiving and transmitting data, signals, information or a combination thereof between units/components within the system and/or connected with the system.

[0040] As discussed in the background section, the current known solutions have several shortcomings. The present disclosure aims to overcome the above-mentioned and other existing problems in this field of technology by providing the method and system for an automatic upgrade of a software for one or more network elements.

[0041] FIG. 1 illustrates an exemplary block diagram representation of a system architecture [100] for performing software upgrades of the one or more network elements [134]. The system architecture [100] as per the present disclosure comprises at least one of a load balancer layer [102], a network management platform layer [104] and an element management layer [106]. The load balancer layer [102] comprises one or more load balancers. The one or more load balancers

can be a device or service that enables distribution of network traffic dynamically across resources which may be to support an application. The one or more load balancers may also be shared by the network management platform layer [104].

5 **[0042]** The network management platform layer [104] comprises a web server [108], a gateway server [110], a firewall [112], an Application Programming Interface (API) [114], and a work order engine [132]. The web server [108] refers to a software and hardware that uses Hypertext Transfer Protocol (HTTP) and other protocols to respond to client requests made over the World Wide Web and telecommunication networks. The gateway server [110] is a device used for transforming data
10 streams among a client and another server. It acts as a node that routes traffic between networks, translating data formats or communication protocols to facilitate communication between different types of devices.

[0043] The work order engine [132] comprises a micro service [116], a dashboard [118], a service
15 scheduler [120], a service execution engine [122], a service monitoring and status update unit [124], a database [126], a hadoop distributed file system (HDFS) [128], and an elastic load balancer [130]. The micro service [116] refers to an architecture and organization approach for software development where software is composed of small independent services that communicate over well-defined APIs. The dashboard [118] is a way of displaying various types of
20 visual data in one place and can be a user interface as well. The Service Scheduler [120] is a component that automates the work that is necessary to sustain various enterprise applications and functionalities. The service execution engine [122] is a component capable of processing data to compute attributes of data. The service monitoring and status update unit [124] is a component that monitors the microservice [116] and updates the status of the microservices [116]. The
25 database [126] refers to an organized collection of digital data that is available electronically, and can be manipulated, updated, stored, backed up, secured, and managed. The HDFS [128] is a software platform that manages data processing and storage for big data applications and operates as a distributed file system that is designed to run on commodity hardware. The elastic load balancer [130] is a device or service that enables distribution of network traffic such as load,
30 request, and responses dynamically across the one or more network elements [134] such as servers, network slice instances, and microservices [116] so that no network element [134] becomes overloaded.

[0044] The element management layer [106] comprises one or more network elements [134]. The
35 element management layer [106] can also comprise an element management system which

manages the one or more network elements [134]. The element management system can be managed or owned by a vendor which manages or owns the one or more network elements [134]. The one or more network elements [134] are one or more network nodes or facilities/equipment that are used within the telecommunication network. In an example, the one or more network elements [134] may be such as an outdoor small cell, an indoor small cell, a macro base station, etc. used in the telecommunication network. Further, in another example, the one or more network elements may also be a server, a network slice instance, a microservice, etc.

[0045] FIG. 2 illustrates an exemplary block diagram of a computing device [200] upon which the features of the present disclosure may be implemented in accordance with exemplary implementation of the present disclosure. In an implementation, the computing device [200] may also implement a method for an automatic upgrade of a software for one or more network elements [134] utilising the system [300]. In another implementation, the computing device [200] itself implements the method for the automatic upgrade of the software for the one or more network elements [134] using one or more units configured within the computing device [200], wherein said one or more units are capable of implementing the features as disclosed in the present disclosure.

[0046] The computing device [200] may include a bus [202] or other communication mechanism for communicating information, and a hardware processor [204] coupled with bus [202] for processing information. The hardware processor [204] may be, for example, a general-purpose microprocessor. The computing device [200] may also include a main memory [206], such as a random-access memory (RAM), or other dynamic storage device, coupled to the bus [202] for storing information and instructions to be executed by the processor [204]. The main memory [206] also may be used for storing temporary variables or other intermediate information during execution of the instructions to be executed by the processor [204]. Such instructions, when stored in non-transitory storage media accessible to the processor [204], render the computing device [200] into a special-purpose machine that is customized to perform the operations specified in the instructions. The computing device [200] further includes a read only memory (ROM) [208] or other static storage device coupled to the bus [202] for storing static information and instructions for the processor [204].

[0047] A storage device [210], such as a magnetic disk, optical disk, or solid-state drive is provided and coupled to the bus [202] for storing information and instructions. The computing device [200] may be coupled via the bus [202] to a display [212], such as a cathode ray tube (CRT), Liquid

crystal Display (LCD), Light Emitting Diode (LED) display, Organic LED (OLED) display, etc. for displaying information to a computer user. An input device [214], including alphanumeric and other keys, touch screen input means, etc. may be coupled to the bus [202] for communicating information and command selections to the processor [204]. Another type of user input device may be a cursor controller [216], such as a mouse, a trackball, or cursor direction keys, for communicating direction information and command selections to the processor [204], and for controlling cursor movement on the display [212]. This input device typically has two degrees of freedom in two axes, a first axis (e.g., x) and a second axis (e.g., y), that allow the device to specify positions in a plane.

[0048] The computing device [200] may implement the techniques described herein using customized hard-wired logic, one or more ASICs or FPGAs, firmware and/or program logic which in combination with the computing device [200] causes or programs the computing device [200] to be a special-purpose machine. According to one implementation, the techniques herein are performed by the computing device [200] in response to the processor [204] executing one or more sequences of one or more instructions contained in the main memory [206]. Such instructions may be read into the main memory [206] from another storage medium, such as the storage device [210]. Execution of the sequences of instructions contained in the main memory [206] causes the processor [204] to perform the process steps described herein. In alternative implementations of the present disclosure, hard-wired circuitry may be used in place of or in combination with software instructions.

[0049] The computing device [200] also may include a communication interface [218] coupled to the bus [202]. The communication interface [218] provides a two-way data communication coupling to a network link [220] that is connected to a local network [222]. For example, the communication interface [218] may be an integrated services digital network (ISDN) card, cable modem, satellite modem, or a modem to provide a data communication connection to a corresponding type of telephone line. As another example, the communication interface [218] may be a local area network (LAN) card to provide a data communication connection to a compatible LAN. Wireless links may also be implemented. In any such implementation, the communication interface [218] sends and receives electrical, electromagnetic or optical signals that carry digital data streams representing various types of information.

[0050] The computing device [200] can send messages and receive data, including program code, through the network(s), the network link [220] and the communication interface [218]. In the

Internet example, a server [230] might transmit a requested code for an application program through the Internet [228], the ISP [226], the local network [222], the host [224] and the communication interface [218]. The received code may be executed by the processor [204] as it is received, and/or stored in the storage device [210], or other non-volatile storage for later execution.

5

[0051] Referring to **FIG. 3**, an exemplary block diagram of a system [300] for an automatic upgrade of a software for one or more network elements [134], is shown, in accordance with the exemplary implementations of the present disclosure. The system [300] may comprise at least one identifier unit [302], at least one analysis unit [304], and at least one transceiver unit [306]. Also, all the components/ units of the system [300] are assumed to be connected to each other unless otherwise indicated below. As shown in the figures all units shown within the system [300] should also be assumed to be connected to each other. Also, in FIG. 3 only a few units are shown, however, the system [300] may comprise multiple such units or the system [300] may comprise any such numbers of said units, as required to implement the features of the present disclosure. Further, in an implementation, the system [300] may be present in a user device/ user equipment to implement the features of the present disclosure. The system [300] may be a part of the user device / or may be independent of but in communication with the user device (may also referred herein as a UE). In another implementation, the system [300] may reside in a server or a network entity. In yet another implementation, the system [300] may reside partly in the server/ network entity and partly in the user device.

10

15

20

[0052] The system [300] is configured for the automatic upgrade of the software for the one or more network elements [134], with the help of the interconnection between the components/units of the system [300].

25

[0053] The present disclosure provides the automatic upgrade of the software for the one or more network elements [134]. The automatic upgrade refers to the upgradation of software elements that are used by the one or more network elements [134] for enhancement of the software. The software for one or more network elements [134] are the set of instructions, data or programs that are used to operate and execute specific tasks in the one or more network elements [134]. The one or more network elements [134] are the one or more network nodes or facilities/equipment that are used within the telecommunication network. In an example, the one or more network elements [134] may be such as an outdoor small cell, an indoor small cell, a macro base station, etc. used in the telecommunication network. Further, in another example, the one or more network elements may also be a server, a network slice instance, a microservice, etc.

30

35

[0054] Initially, the identifier unit [302] is configured to identify the one or more network elements [134] for performing software upgrades. Performing software upgrades, as used herein, refers to installing a software which provides enhancement of the software used by the one or more network elements [134]. For performing software upgrades, the present disclosure provides identifying the one or more network elements [134] that requires upgradation of software.

[0055] The present disclosure further discloses that, for identifying the one or more network elements [134], the identifier unit [302] may also configured to select a list of vendors, then select a list of one or more network elements [134], thereafter, mapping the relevant or latest software version and generating work order for upgrade. Then, selecting network element details and enabling bulk operation for the software upgrade.

[0056] In one example, the identifier unit [302] may select the list of the vendors. The vendor devices can exist in multiple generations (4G, 5G, 6G ... nG) of cellular networks. Then, for selecting the list of one or more network elements [134] in multiple generation of cellular networks having multiple network element types, the identifier unit [302] may be configured to identify one or more network elements [134] within each generation of network element type. Further, the end user can select a subset of select network element(s) for upgradation based on different logical choices (such as network area coverage, location criticality, etc.).

[0057] For mapping the relevant or latest software version that the one or more network elements [134] are required to be upgraded. Each Vendor Network Element Type will have a unique Software Version that is required to be upgraded or installed at the network element[134]. Each unique software version may have a one-to-one association with a unique recipe.

[0058] For generating work orders related to performing software upgrades, Work Order having differing number of Network Elements of different Network Element Types are opted for upgrading to different version along with their recipes.

[0059] For selecting network element details and enabling bulk operation, the system will automatically pick the selected network element details and enable the bulk operation. The above listed fields can be tweaked or configured based on Operations and Management (O and M) admin choices. Target nodes are grouped together based on different regions and/or admin options (EMS) such as District-wise/State-wise/Country-wise, EMS across region and Individual EMS. The target

node may refer to the nodes or the one or more network elements [134] that are identified or selected for performing software upgrade. For example, the target nodes are selected based on a geographical area and the network element for a state are required to be updated. Now, the entire one or more network elements [134] within such geographical area may be selected as the target nodes, and then may be used for performing software upgrades.

[0060] On identification of the one or more network elements [134], the analysis unit [304] is configured to determine one of a positive satisfaction and a negative satisfaction of one or more pre-checks on each of the one or more network elements [134], wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks. The one or more pre-checks refer to checking of one or more parameters before performing software upgrades. The positive satisfaction is a positive response from the one or more pre-checks based on the success of the pre-checks. The negative satisfaction is a negative response from the one or more pre-checks based on the failure of the pre-checks. The success and the failure of the pre-checks are the outputs of the pre-checks after the pre-checks have been performed and indicates the outputs of the pre-checks.

[0061] The one or more pre-checks comprises a memory availability, a central processing unit (CPU) utilization, a health check, a system requirement check, and a backup configuration check. The memory availability refers to a check related to availability of storage/ space for storing additional data. For example, for memory availability check, the positive response would be existence of additional storage space in the memory unit of the one or more network nodes, and the negative response would be non-availability of additional storage for performing software upgrades. The central processing unit (CPU) utilization is the amount of load on an individual processor cores for running various programs on the one or more network nodes. For example, if the amount of load/ CPU utilization is high, then it may be that the CPU cannot be used for performing software upgrade and this will be the negative response, however, if the CPU utilization is lower, then it can be used for performing software upgrade, which will result in a positive response. The health check refers to checking of a health status of the one or more network nodes which checks if the one or more network nodes are performing effectively and efficiently which can be done by evaluation of the Quality of Service provided by the one or more network nodes. For example, a positive response for the health check would be if the one or more network nodes are healthy, and if the one or more network nodes are not healthy, then it would be considered as the negative response. The system requirements check is a test which verifies the hardware and the software installed in the one or more network nodes and identifies if the one or more network

nodes are capable of performing software upgrades without any error based on the predefined system requirements. The predefined system requirement can be dynamically defined based on the software upgrade. For example, a positive response for system requirement check is if all the hardware and software are verified based on one or more predefined system requirements, and in case the hardware and software is not verified then it would provide a negative response. The backup configuration check comprises checking whether the one or more network nodes has a backup configuration for handling any errors in the software upgrades. For example, a positive response would be if the backup configuration exists, otherwise it would be a negative response.

[0062] Also, after identification of the one or more network elements [134], the analysis unit [304] is further configured to determine an existence or a non-existence of one or more critical alarms. The one or more critical alarms may be on the one or more network elements [134]. The determination of existence or non-existence of the one or more critical alarms are based on an extraction of a log received from the one or more network elements [134]. The one or more critical alarms refer to the indication that there may be one or more issues in the one or more network elements [134]. The existence of the one or more critical alarms is the presence of one or more issues in the one or more network elements [134] based on the one or more critical alarms. The non-existence of the one or more critical alarms is the presence of the one or more issues in the one or more network elements [134] based on the one or more critical alarms. For checking the one or more critical alarms the one or more network elements [134] are sent a command to extract the alarms along with the corresponding details of the one or more network elements [134]. The one or more network elements [134] send the log that is a detailed log file which contains a record of events that occurred in the one or more network elements [134], the events may also contain information associated with functioning of the one or more network elements [134]. The determination of the existence or non-existence of the one or more critical alarms can be determined based on extraction and evaluation of the log, and then analysing the record of events in the detailed log file. Furthermore, a trap event is caused due to occurrence of an interrupt during functioning of the one or more network elements [134]. The network node may, while transmission of the log, convert the log into a compressed or encrypted format, and then while the analysis unit [304] receives the compressed or encrypted format from the one or more network elements [134] for obtaining the log or extraction of the log, the analysis unit may decompress or decrypt that format and extract the log for obtaining such information associated with existence or non-existence of one or more critical alarms.

[0063] After determination of the one or more pre-checks and the one or more critical alarms, the analysis unit [304] is further configured to download the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms. Based on the determination of non-existence of the one or more critical alarms by the analysis unit [304], and determination of positive satisfaction of the one or more pre-checks, it is determined whether the one or more network elements [134] are ready for performing the software upgrade. After the one or more network elements [134] are ready, the files containing software may also be downloaded on the one or more network elements [134].

[0064] In case of positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms, the transceiver unit [306] is configured to send to a vendor system, a software upgrade request for upgrading a network element from the one or more network elements [134] in cases where the one or more network elements [134] belong to an external network. The software upgrade request is a request for performing software upgrade of the one or more network elements [134]. A network element is a network nodes or facility/equipment that is used within the telecommunication network. The vendor system can be a vendor management system or an element management system which manages the one or more network elements [134]. In case the one or more network elements [134] are owned or managed by a vendor or external party, the software upgrade request is sent to the vendor system. The external network refers to the part of network which is owned or managed by a separate entity.

[0065] In case of the negative satisfaction of the one or more pre-checks and the existence of the one or more critical alarms, the analysis unit [304] is further configured to perform in real time, one or more from a re-attempt function, a cancel function, an abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms. The re-attempt function is an attempt for performing software upgrade of the one or more network elements [134] again. The cancel function is cancelling the attempt for performing software upgrade. The abort function is aborting the attempt for performing the software upgrade. The reboot function is rebooting the software of the one or more network elements [134]. The event of the negative satisfaction and the existence of the one or more critical alarms is the case when the one or more pre-checks results in negative satisfaction or negative response, and there is an existence of the one or more critical alarms. This is a scenario where if there are any issues which may lead to inability of performing the software upgrade, the one or more network elements [134] are required to perform the above-mentioned functions for enabling the one or more network elements [134] to perform the software upgrade. The above-mentioned functions are performed in

real-time to solve the problems in the one or more network nodes and make them capable for performing the software upgrade without requiring any manual intervention.

[0066] Once the software is downloaded, the analysis unit [304] is further configured to perform one or more post-checks on the each of the one or more network elements [134]. The one or more post-checks are the test performed on each of the one or more network elements [134] after said performing software upgrade of the one or more network elements [134].

[0067] The one or more post-checks comprises one or more of a software version check, and a performance metrics check. The software version check is a test which checks for the information associated with the software version and may also contain the information associated with last upgrade of the one or more network elements [134]. The performance metrics check comprises measuring one or more key performance metrics comprising a throughput metric, a latency metric, and a capacity metric. The throughput metric refers to the amount of data which is transferred across the telecommunication network from a source to a destination, over a period of time. The latency metric is the delay or time taken in the transfer of the data from the source to the destination. The capacity metric is the capability or amount of traffic that can be handled by the one or more network elements [134].

[0068] The present disclosure further discloses that the upgrading of the network element from the one or more network elements [134] is performed based on a stage associated with said network element. The present disclosure also provides that performing software upgrades can also be done in stages, where a predefined count of network elements [134] from the one or more network elements [134] can be selected for performing software upgrades, and after performing software upgrades for the predefined count of network elements [134], the one or more network elements [134] that are next to the predefined count of network elements [134] from the one or more network elements [134] are then considered for performing software upgrades. Since, the one or more network elements [134] cannot be used for providing any service during the performance of the software upgrades, the performance of software upgrades stage-wise allows for continuous usage of the telecommunication network without interrupting the service of the telecommunication network, and the software upgrade can be automatically installed. There may be various stages for performing software upgrade. For example, a predefined count of the one or more network elements [134] can be upgraded in a first stage, thereafter, at next stage another predefined count of the one or more network elements [134] can be upgraded. There can be various stages based on

the total availability and requirements of one or more network elements [134]. Finally, at last stage, the remaining count of the one or more network elements [134] may be upgraded.

[0069] The present disclosure further discloses that the event of the upgrading of the network element from the one or more network elements [134], a coverage area associated with said network element is assigned to one or more other network elements [134]. The coverage refers to the geographical region which is covered by the one or more network elements [134] for providing the service for telecommunication networks. In order to avoid interrupting the service provided by the one or more network elements [134], the identification of the one or more network elements [134] for performing software upgrade is such that the one or more network elements [134] that can provide the services of telecommunication network for the same coverage area are not identified for performing software upgrades. Such one or more network elements [134] would be selected in another stage in case of stage-wise software upgrades. Such one or more network elements [134] can remain active for providing services in the coverage area while the software for the identified one or more network elements [134] are being upgraded.

[0070] Referring to **FIG. 4**, an exemplary method flow diagram [400] for an automatic upgrade of a software for one or more network elements [134], in accordance with exemplary implementations of the present disclosure is shown. In an implementation the method [400] is performed by the system [300]. Further, in an implementation, the system [300] may be present in a server device to implement the features of the present disclosure. Also, as shown in FIG. 4, the method [400] starts at step [402].

[0071] The present disclosure provides the method [400] for automatic upgrade of a software for one or more network elements [134]. The automatic upgrade refers to the upgradation of software elements that are used by the one or more network elements [134] for enhancement of the software. The software for one or more network elements [134] are the set of instructions, data or programs that are used to operate and execute specific tasks in the one or more network elements [134]. The one or more network elements [134] are the one or more network nodes or facilities/equipment that are used within the telecommunication network. In an example, the one or more network elements [134] may be such as an outdoor small cell, an indoor small cell, a macro base station, etc. used in the telecommunication network.

[0072] Initially, at step [404], the method [400] comprises identifying, by an identifier unit [302], the one or more network elements [134] for performing software upgrades. The performing

software upgrades, as used herein, refers to installing a software which provides enhancement of the software used by the one or more network elements [134]. For performing software upgrades, the present disclosure provides identifying the one or more network elements [134] that requires upgradation of software.

5

[0073] On identification of the one or more network elements [134], at step [406], the method [400] comprises determining, by an analysis unit [304], one of a positive satisfaction and a negative satisfaction of one or more pre-checks on each of the one or more network elements [134], wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks. The one or more pre-checks refers to checking of one or more parameters before performing software upgrades. The positive satisfaction is a positive response from the one or more pre-checks based on the success of the pre-checks. The negative satisfaction is a negative response from the one or more pre-checks based on the failure of the pre-checks. The success and the failure of the pre-checks are the outputs of the pre-checks after the pre-checks have been performed and indicates the output of the pre-checks.

[0074] The one or more pre-checks comprises a memory availability, a central processing unit (CPU) utilization, a health check, a system requirement check, and a backup configuration check. The memory availability refers to a check related to availability of storage/ space for storing additional data. For example, for memory availability check, the positive response would be existence of additional storage space in the memory unit of the one or more network nodes, and the negative response would be non-availability of additional storage for performing software upgrades. The central processing unit (CPU) utilization is the amount of load on an individual processor core for running various programs on the one or more network nodes. For example, if the amount of load/ CPU utilization is high, then it may be that the CPU cannot be used for performing software upgrade and this will be the negative response, however, if the CPU utilization is lower, then it can be used for performing software upgrade, which will result in a positive response. The health check refers to checking of a health status of the one or more network nodes which checks if the one or more network nodes are performing effectively and efficiently which can be done by evaluation of the Quality of Service provided by the one or more network nodes. For example, a positive response for the health check would be if the one or more network nodes are healthy, and if the one or more network nodes are not healthy, then it would be considered as the negative response. The system requirement check is a test which verifies the hardware and the software installed in the one or more network nodes, and identifies if the one or more network

nodes are capable for performing software upgrades without any error based on the predefined system requirements, the predefined system requirement can be dynamically defined based on the software upgrade. For example, a positive response for system requirement check is if all the hardware and software are verified based on one or more predefined system requirements, and in case the hardware and software is not verified then it would provide a negative response. The backup configuration check comprises checking whether the one or more network nodes has a backup configuration for handling any errors in the software upgrades. For example, a positive response would be if the backup configuration exists, otherwise it would be a negative response.

[0075] Also, after identification of the one or more network elements [134], then at step [408], the method [400] further comprise determining, by the analysis unit [304], an existence or a non-existence of one or more critical alarms. The one or more critical alarm may be on the one or more network elements [134]. The determination of the existence or the non-existence of the one or more critical alarms are based on an extraction of a log received from the one or more network elements [134]. The one or more critical alarms refers to the indication that there may be one or more issues in the one or more network elements [134]. The existence is the presence of one or more issues in the one or more network elements [134] based on the one or more critical alarms. The non-existence is the presence of the one or more issues in the one or more network elements [134] based on the one or more critical alarms. For checking the one or more critical alarms the one or more network elements [134] are sent a command to extract the alarms along with the corresponding details of the one or more network elements [134]. The one or more network elements [134] send the log that is a detailed log file which contains a record of events that occurred in the one or more network elements [134], the events may also contain information associated with functioning of the one or more network elements [134]. The determination of the existence or non-existence of the one or more critical alarms can be determined based on extraction and evaluation of the log, and then analysing the record of events in the detailed log file, and a trap event caused due to occurrence of an interrupt during functioning of the one or more network elements [134]. The network node may while transmission of the log converts the log into a compressed or encrypted format, and then while the analysis unit [304] receives the compressed or encrypted format from the one or more network elements [134], for obtaining the log or extraction of the log, the analysis unit may decompress or decrypt that format and extract the log for obtaining such information associated with existence or non-existence of one or more critical alarms.

[0076] After determination of the one or more pre-checks and the one or more critical alarms, then at step [410], the method [400] comprises downloading, by the analysis unit [304], the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms. Based on the determination of non-existence of the one or more critical
5 alarms by the analysis unit [304], and determination of positive satisfaction of the one or more pre-checks, it is determined whether the one or more network elements [134] are ready for performing the software upgrade. After the one or more network elements [134] are ready, the files containing software may also be downloaded on the one or more network elements [134].

10 [0077] In case of positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms, then at step [412], the method [400] further comprises sending, by a transceiver unit [306] to a vendor system, a software upgrade request for upgrading a network element from the one or more network elements [134], wherein the one or more network elements [134] belong to an external network. The software upgrade request is a request for performing
15 software upgrade of the one or more network elements [134]. A network element is a network node or facility/equipment that is used within the telecommunication network. The vendor system can be a vendor management system or an element management system which manages the one or more network elements [134]. In case the one or more network elements [134] are owned or managed by a vendor or external party, the software upgrade request is sent to the vendor system.
20 The external network refers to the part of network which is owned or managed by a separate entity.

[0078] In case of the negative satisfaction of the one or more pre-checks and the existence of the one or more critical alarms, then at step [414], the method [400] further comprises performing in real time, by the analysis unit [304], one or more from a re-attempt function, a cancel function, an
25 abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms. The re-attempt function is an attempt for performing software upgrade of the one or more network elements [134] again. The cancel function is cancelling the attempt for performing software upgrade. The abort function is aborting the attempt for performing the software upgrade. The reboot function is rebooting the software of the one or more network
30 elements [134]. The event of the negative satisfaction and the existence of the one or more critical alarms is the case when the one or more pre-checks results in negative satisfaction or negative response, and there is the existence of the one or more critical alarms. This is a scenario where if there are any issues which may lead to inability for performing the software upgrade, the one or more network elements [134] are required to perform the above-mentioned functions for enabling
35 the one or more network elements [134] to perform the software upgrade. The above-mentioned

functions are performed in real-time to solve the problems in the one or more network nodes and make them capable for performing the software upgrade without requiring any manual intervention.

5 **[0079]** Once the software is downloaded, then at step [416], the method [400] further comprises performing, by the analysis unit [304], one or more post-checks on the each of the one or more network elements [134]. The one or more post-checks are the test performed on each of the one or more network elements [134] after said performing software upgrade of the one or more network elements [134].

10

[0080] The one or more post-checks comprises one or more of a software version check, and a performance metrics check. The software version check is a test which checks for the information associated with the software version and may also contain the information associated with last upgrade of the one or more network elements [134]. The performance metrics check comprises
15 measuring one or more key performance metrics comprising a throughput metric, a latency metric, and a capacity metric. The throughput metric refers to the amount of data which is transferred across the telecommunication network from a source to a destination, over a period of time. The latency metric is the delay or time taken in the transfer of the data from the source to the destination. The capacity metric is the capability or amount of traffic that can be handled by the
20 one or more network elements [134].

[0081] The upgrading of the network element from the one or more network elements [134] is performed based on a stage associated with said network element. The present disclosure also provides performing software upgrades in stages, where a predefined count of network elements
25 [134] from the one or more network elements [134] can be selected for performing software upgrades, and after performing software upgrades for the predefined count of network elements [134], the one or more network elements [134] that are next to the predefined count of network elements [134] from the one or more network elements [134] are then considered for performing software upgrades. Since, the one or more network elements [134] cannot be used for providing
30 any service during the performance of the software upgrades, this stagewise software upgrade performance allows for continuous usage of the telecommunication network without interrupting the service of the telecommunication network, and the software upgrade can be automatically installed. There may be various stages for performing software upgrade. For example, a predefined count of the one or more network elements [134] can be upgraded in a first stage, thereafter, at
35 next stage another predefined count of the one or more network elements [134] can be upgraded.

There can be various stages based on the total availability and requirements of one or more network elements [134]. Finally, at last stage, the remaining count of the one or more network elements [134] may be upgraded.

5 **[0082]** The present disclosure further discloses that the event of the upgrading of the network element from the one or more network elements [134], a coverage area associated with said network element is assigned to one or more other network elements [134]. The coverage refers to the geographical region which is covered by the one or more network elements [134] for providing the service for telecommunication networks. In order to avoid interrupting the service provided by
10 the one or more network elements [134], the identification of the one or more network elements [134] for performing software upgrade is such that the one or more network elements [134] that can provide the services for telecommunication network for the same coverage area are not identified for performing software upgrades. Such one or more network elements [134] would be selected in another stage in case of stage-wise software upgrades. Such one or more network
15 elements [134] can remain active for providing services in the coverage area while the software for the identified one or more network elements [134] are being upgraded.

[0083] Thereafter, at step [418], the method [400] is terminated.

20 **[0084]** Referring to **FIG. 5**, an exemplary method flow diagram [500] for an automatic upgrade of a software for one or more network elements [134], in accordance with exemplary implementations of the present disclosure is shown. In an implementation the method [500] is performed by the system [300]. Further, in an implementation, the system [300] may be present in a server device to implement the features of the present disclosure. Also, as shown in FIG. 5, the
25 method [500] starts at step [502].

[0085] Initially, at step [504], the method [500] comprises creating a workorder for performing software upgrade for one or more network elements [134]. Then at step [506], the method [500] involves submitting the workorder for performing software upgrade for the one or more network
30 elements [134].

[0086] Then as a part of the present implementation at step [508], the method [500] comprises performing one or more pre-checks of each of the one or more network elements [134] and determining a positive satisfaction at step [510], in case of a success of the one or more pre-checks,
35 and determining a negative satisfaction at step [512], in case of a failure of the one or more pre-

checks. Thereafter, at step [514], the method [500] further comprises performing one or more critical alarms check for determining, at step [516], an existence of one or more critical alarms on each of the one or more network elements [134], and determining, at step [518], a non-existence of the one or more critical alarms on each of the one or more network elements [134]. Thereafter, in case of non-existence of the one or more critical alarms, and the positive satisfaction of the one or more pre-checks, the method [500] at step [520] comprises downloading the software of the one or more network elements [134] for performing software upgrades. Then at step [522], determining a failure of downloading of the software of the one or more network elements [134] have not gone through/not completed, and determining, at step [524], a success of downloading of the software of the one or more network elements [134] based on completion of the download of software.

[0087] The method [500] then moves to step [526] for performing a software upgrade of the one or more network elements [134], and also comprises sending a software upgrade request for upgrading a network element from the one or more network elements [134] in case the one or more network elements [134] belong to the external network. Further, at step [528], the method [500] comprises determining a failure of performing software upgrade, and at step [530], the method [500] comprises determining a success of performing software upgrade. This indicates that the software upgrade for each of the one or more network elements [134] is successful or not. After, performing software upgrades, at step [532], the method [500] involves performing one or more post-checks on each of the one or more network elements [134]. Then determining, at step [534], a failure in case the one or more post-checks resulted in an undesirable outcome. Further, determining, at step [536], a success of the one or more post-checks in case of a desirable outcome. The desirable and undesirable outcome refers to the outcome during checking of one or more parameters of the one or more network elements [134] and the performance of the one or more network elements [134] is determined to be efficient and effective or not.

[0088] In case, the one or more post-checks are successful, then the method [500] is terminated.

[0089] In any case of the failure of the one or more pre-checks, existence of the one or more critical alarms, the failure of performing software upgrades, and failure of one or more post checks, the method [500] may also comprises performing one or more from a re-attempt function, a cancel function, an abort function, and a reboot function. This would enable solving of any issues that is a hinderance for performing software upgrades.

[0090] The present disclosure further discloses a non-transitory computer readable storage medium storing instructions for an automatic upgrade of a software for one or more network elements [134], the instructions include executable code which, when executed by one or more units of a system [300], causes: an identifier unit [302] to identify one or more network elements [134] for performing software upgrades; an analysis unit [304] to determine one of a positive satisfaction and a negative satisfaction of one or more pre-checks on each of the one or more network elements [134], wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks; the analysis unit [304] to determine one of an existence and a non-existence of one or more critical alarms on the each of the one or more network elements [134] based on an extraction of a log received from the one or more network elements [134]; the analysis unit [304] to download the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms; a transceiver unit [306] to send to a vendor system, a software upgrade request for upgrading a network element from the one or more network elements [134], wherein the one or more network elements [134] belong to an external network; the analysis unit [304] to perform in real time one or more from a re-attempt function, a cancel function, an abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms; and the analysis unit [304] to perform one or more post-checks on the each of the one or more network elements [134].

[0091] As is evident from the above, the present disclosure provides a technically advanced solution for an automatic upgrade of a software for one or more network elements. The present disclosure provides an automated mechanism to upgrade node software/firmware with very less human intervention along with multiple vendor selection options. Further, the present disclosure ensures a smooth and successful transition to newer software versions while minimizing disruptions and maintaining system functionality and performance. This leads to decrease in manual steps, time taken to upgrade, and the amount of effort required of user(s). This results in increased performance of the network elements, decrease in manual errors and provides a superior user experience.

[0092] While considerable emphasis has been placed herein on the disclosed implementations, it will be appreciated that many implementations can be made and that many changes can be made to the implementations without departing from the principles of the present disclosure. These and other changes in the implementations of the present disclosure will be apparent to those skilled in

the art, whereby it is to be understood that the foregoing descriptive matter to be implemented is illustrative and non-limiting.

[0093] Further, in accordance with the present disclosure, it is to be acknowledged that the functionality described for the various components/units can be implemented interchangeably. While specific embodiments may disclose a particular functionality of these units for clarity, it is recognized that various configurations and combinations thereof are within the scope of the disclosure. The functionality of specific units as disclosed in the disclosure should not be construed as limiting the scope of the present disclosure. Consequently, alternative arrangements and substitutions of units, provided they achieve the intended functionality described herein, are considered to be encompassed within the scope of the present disclosure.

We Claim:

1. A method for an automatic upgrade of a software for one or more network elements [134], the method comprising:
 - 5 - identifying, by an identifier unit [302], the one or more network elements [134] for performing software upgrades;
 - determining, by an analysis unit [304], one of a positive satisfaction and a negative satisfaction of one or more pre-checks on each of the one or more network elements [134], wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks;
 - 10 - determining, by the analysis unit [304], one of an existence and a non-existence of one or more critical alarms on the each of the one or more network elements [134] based on an extraction of a log received from the one or more network elements [134];
 - 15 - downloading, by the analysis unit [304], the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms;
 - sending, by a transceiver unit [306] to a vendor system, a software upgrade request for upgrading a network element [134] from the one or more network elements [134], wherein the one or more network elements [134] belong to an external network;
 - 20 - performing in real time, by the analysis unit [304], one or more from a re-attempt function, a cancel function, an abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms; and
 - performing, by the analysis unit [304], one or more post-checks on the each of the one or more network elements [134].
 - 25
2. The method as claimed in claim 1, wherein the one or more pre-checks include a memory availability, a central processing unit (CPU) utilization, a health check, a system requirement check, and a backup configuration check.
- 30
3. The method as claimed in claim 1, wherein the one or more post-checks comprises one or more of a software version check, and a performance metrics check.

4. The method as claimed in claim 3, wherein the performance metrics check comprises measuring one or more key performance metrics comprising a throughput metric, a latency metric, and a capacity metric.

5. The method as claimed in claim 1, wherein the upgrading of the network element from the one or more network elements [134] is performed based on a stage associated with said network element.

6. The method as claimed in claim 1, wherein in the event of the upgrading of the network element from the one or more network elements [134], a coverage area associated with said network element is assigned to one or more other network elements [134].

7. A system [300] for an automatic upgrade of a software for one or more network elements [134], the system [300] comprising:

- an identifier unit [302], configured to identify the one or more network elements [134] for performing software upgrades;
- an analysis unit [304] connected at least to the identifier unit [302], the analysis unit [304] is configured to:

determine one of a positive satisfaction and a negative satisfaction of one or more pre-checks on each of the one or more network elements [134], wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks, determine one of an existence and a non-existence of one or more critical alarms on the each of the one or more network elements [134] based on an extraction of a log received from the one or more network elements [134],

download the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms; and

- a transceiver unit [306] connected at least to the analysis unit [304], the transceiver unit [306] is configured to send to a vendor system, a software upgrade request for upgrading a network element [134] from the one or more network elements [134], wherein the one or more network elements [134] belong to an external network;

- the analysis unit [304] is further configured to:
 - perform in real time, one or more from a re-attempt function, a cancel function, an abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms, and

perform one or more post-checks on the each of the one or more network elements [134].

8. The system [300] as claimed in claim 7, wherein the one or more pre-checks include a memory availability, a central processing unit (CPU) utilization, a health check, a system requirement check, and a backup configuration check.

9. The system [300] as claimed in claim 7, wherein the one or more post-checks comprises one or more of a software version check, and a performance metrics check.

10. The system [300] as claimed in claim 9, wherein the performance metrics check comprises measuring one or more key performance metrics comprising a throughput metric, a latency metric, and a capacity metric.

11. The system [300] as claimed in claim 7, wherein the upgrading of the network element [134] from the one or more network elements [134] is performed based on a stage associated with said network element.

12. The system [300] as claimed in claim 7, wherein in the event of the upgrading of the network element [134] from the one or more network elements [134], a coverage area associated with said network element [134] is assigned to one or more other network elements [134].

13. A non-transitory computer-readable storage medium storing instructions for an automatic upgrade of a software for one or more network elements [134], the instructions comprising executable code which, when executed by one or more units of a system [300], causes:

- an identifier unit [302] of the system [300] to identify the one or more network elements [134] for performing software upgrades;
- an analysis unit [304] of the system [300] to determine one of a positive satisfaction and a negative satisfaction of one or more pre-checks on each of the one or more network elements [134], wherein the positive satisfaction is determined based on a success of the one or more pre-checks and the negative satisfaction is determined based on a failure of the one or more pre-checks;
- the analysis unit [304] of the system [300] to determine one of an existence and a non-existence of one or more critical alarms on the each of the one or more network

elements [134] based on an extraction of a log received from the one or more network elements [134];

- the analysis unit [304] of the system [300] to download the software in an event of the positive satisfaction of the one or more pre-checks and the non-existence of the one or more critical alarms;
- a transceiver unit [306] of the system [300] to send, to a vendor system, a software upgrade request for upgrading a network element [134] from the one or more network elements [134], wherein the one or more network elements [134] belong to an external network;
- the analysis unit [304] of the system [300] to perform in real time one or more from a re-attempt function, a cancel function, an abort function, and a reboot function in an event of the negative satisfaction and the existence of the one or more critical alarms; and
- the analysis unit [304] of the system [300] to perform one or more post-checks on the each of the one or more network elements [134].

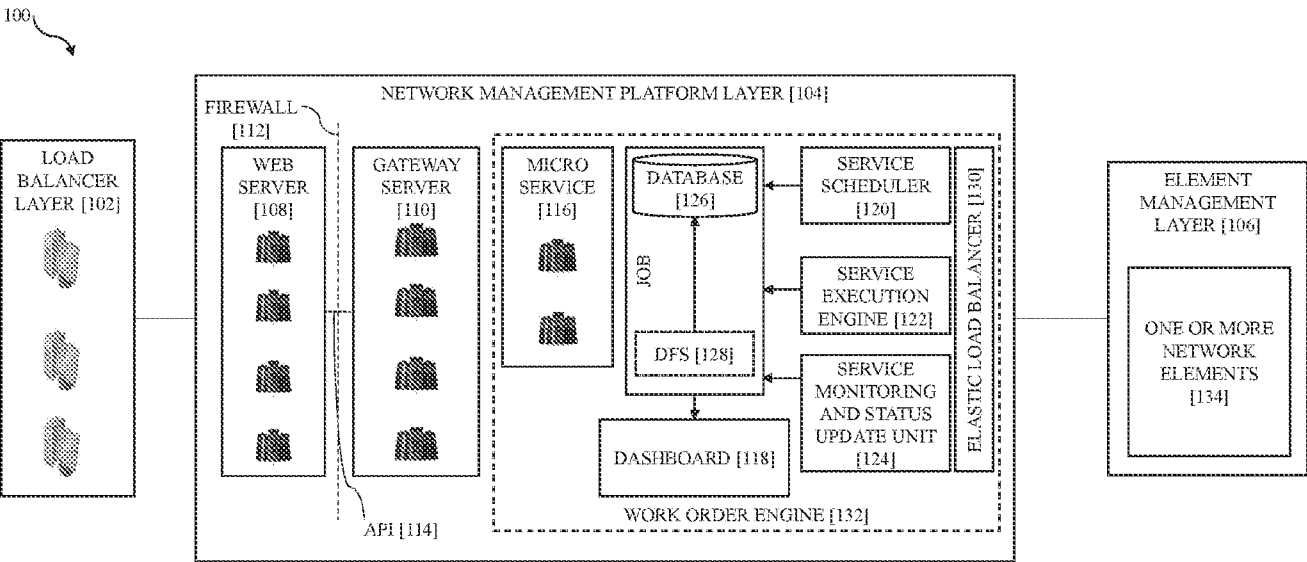


FIG. 1

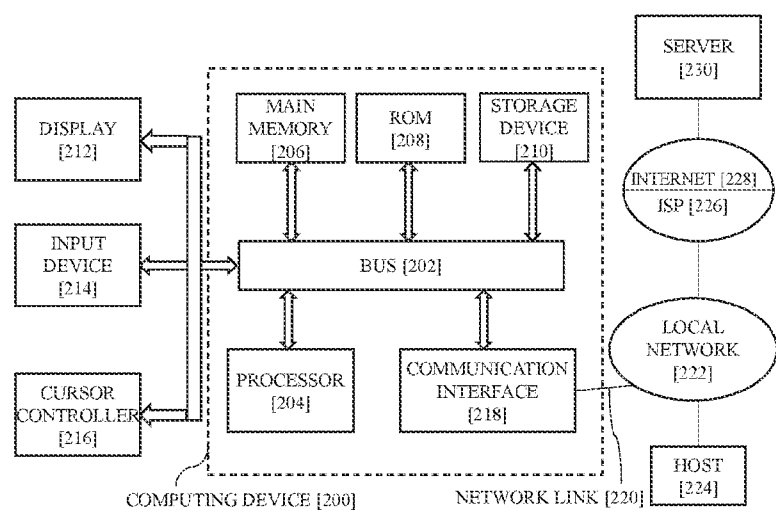


FIG. 2

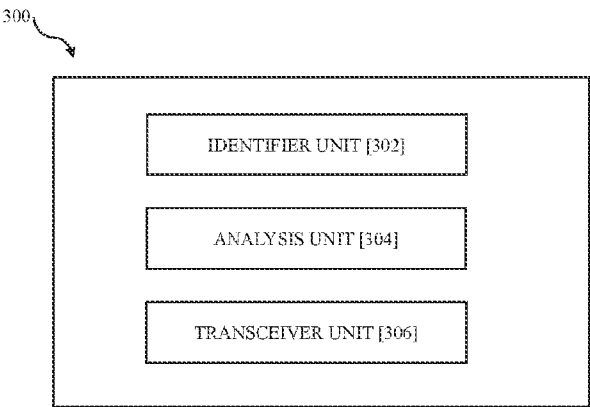


FIG. 3

4/5

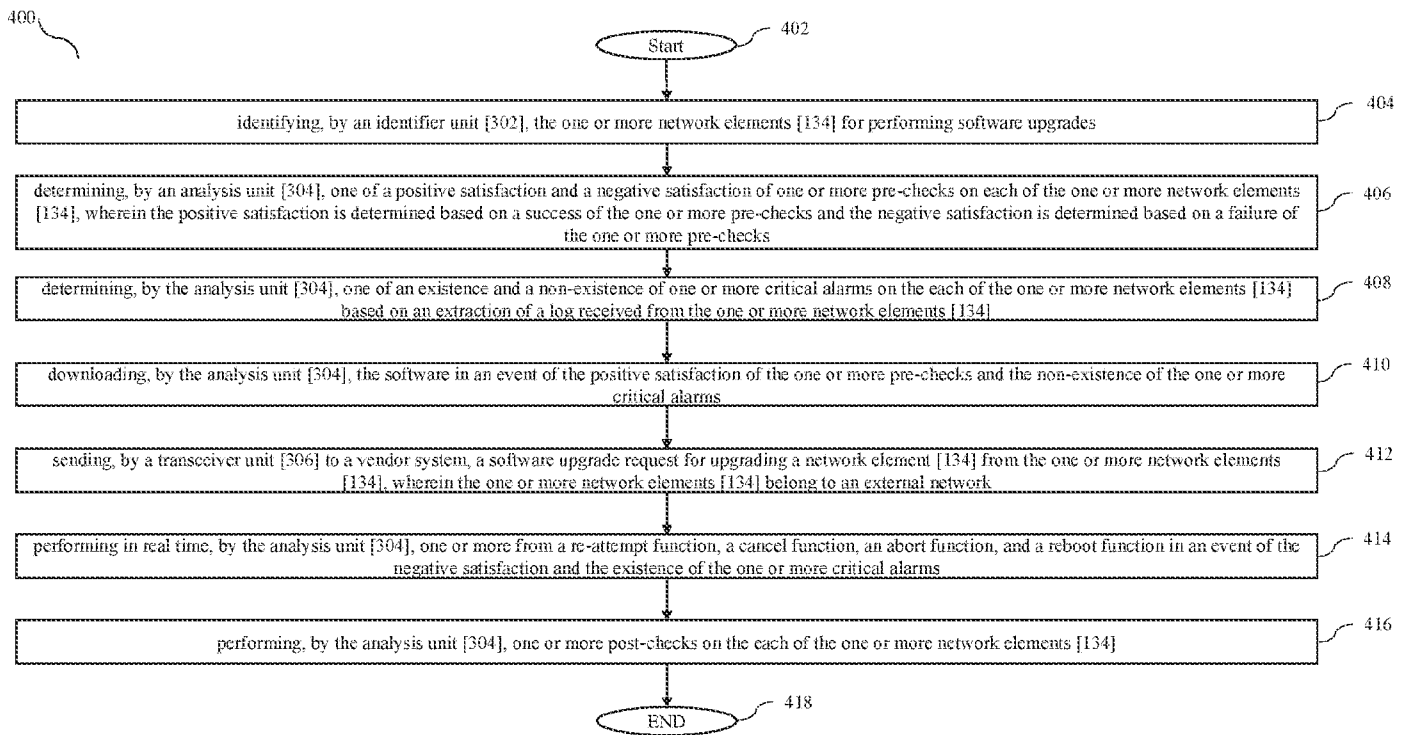


FIG. 4

5/5

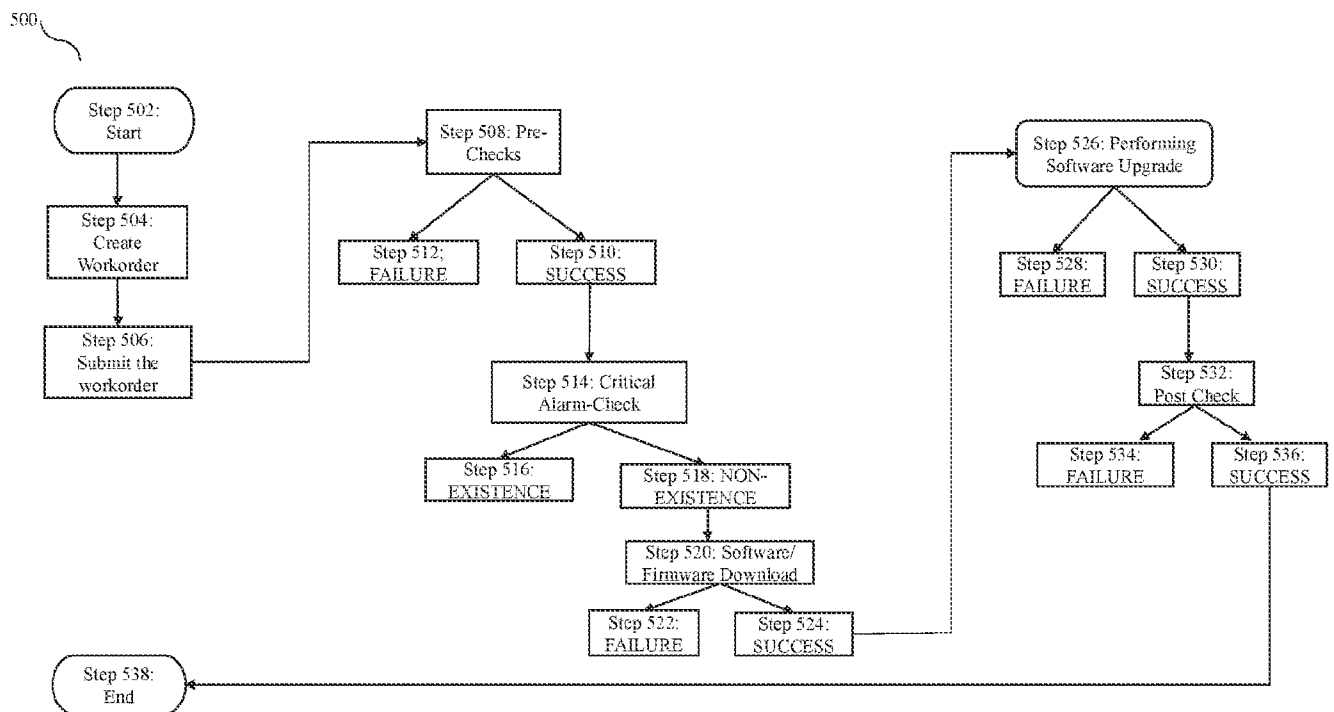


FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/IN2024/051250

A. CLASSIFICATION OF SUBJECT MATTER
H04L41/082, G06F8/65 Version=2024.01

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04L, H04W, G06F

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

Electronic database consulted during the international search (name of database and, where practicable, search terms used)

PatSeer, IPO Internal Database; PRE-CHECK, POST-CHECK, SOFTWARE UPGRADE, ALARM, NETWORK ELEMENT, RE-ATTEMPT, REBOOT, CANCEL, ABORT

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US20210359905A1 (ERICSSON AB) (18-Nov-2021) (18-11-2021) paragraphs [0044]-[0057], [0059]-[0073], figures 1-2, 14-15, and claims 1-14	1-13
Y	CN114928523A (GUANGZHOU AIPU ROAD NETWORK TECHNOLOGY CO LTD) (English translation from Google Patents) (19-Aug-2022) (19-08-2022) claims 1-9	1-13



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

"D" document cited by the applicant in the international application

"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

"I" 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

20-11-2024

Date of mailing of the international search report

20-11-2024

Name and mailing address of the ISA/

Indian Patent Office
Plot No.32, Sector 14, Dwarka, New Delhi-110075
Facsimile No.

Authorized officer

Vivek Vishwakarma

Telephone No. +91-1125300200

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/IN2024/051250

Citation	Pub.Date	Family	Pub.Date
US 20210359905 A1	18-11-2021	CN 113196706 A	30-07-2021
		EP 3871371 A1	01-09-2021
		WO 2020084497 A1	30-04-2020
CN 114928523 A	19-08-2022	CN 114928523 B	05-09-2023