

Description

Provisioning of software applications on edge devices in an Internet-of-Things environment

5

The present invention generally relates to the field of Internet-of-Things (IoT), and more particularly relates to a method and system for provisioning software applications on edge devices in an IoT environment.

10

In an IoT environment, such as Industrial IoT (IIoT) environment, a plurality of edge devices is connected to an IoT cloud platform and sensors and actuators via the Internet. The edge devices may be light weight, low cost devices which collect data from various sensors and actuators deployed in a plant, store and buffer the collected data, conduct analysis of the collected data, and perform an action (e.g., issuing a control command) based on the outcome of analysis. The edge devices may also be configured to aggregate, filter, selectively report, compress, encrypt, and/or otherwise pre-process the plant data, resulting in less data and/or value added data being communicated to the IoT cloud platform. This may lead to consumption of less network communication and backend storage and processing resources than may have been required or consumed absent such pre-processing. Typically, each of the edge devices may perform the above functions using one or more software applications deployed therein. The IoT cloud platform is responsible for managing the edge devices, managing the data (raw or pre-processed) received from the edge devices, and provisioning software applications on the edge devices.

Typically, the edge devices in the IoT environment are resource constrained devices. Further, each of the resource constrained devices may vary in terms of hardware specification, and may have different real-time resource capabilities for executing new software applications provisioned by the IoT cloud platform. For example, the edge

35

device which runs on Raspberry PiTM may be capable of
executing different software applications than the edge
device based on typical industrial PC. Also, the real-time
resource capabilities may vary based on number of software
5 applications deployed on the edges device and resources
utilized by the edge device in executing the software
applications during run-time. A new software application
deployed on an edge device having incompatible hardware and
resource constraint may encounter performance issues (e.g.,
10 device uptime) during execution of the software application
in real-time. Also, this may lead to erroneous analysis of
the plant data.

US2016/0232366A1 discloses a verification system capable of
15 simulating performance of a real-time device in a given
scenario. The verification system receives information on
specific scenarios which loaded into a virtual simulator for
simulating capabilities of device of interest, runs a suite
of simulation events by the virtual simulator, and determines
20 whether the performance of the device meets the target
performance. The verification system deploys a device
configuration profile to the real-time device if the
performance of the device meets the target performance.
However, the verification system does not solve the problems
25 described above.

In light of the above, there exists a need for an improved
method and system for provisioning software applications on
edge devices in an IoT environment.

30

Therefore, it is an object of the present invention to
provide a method and system for provisioning software
applications on edge devices in an IoT environment.

35 The object of the present invention is achieved by a method
of provisioning a software application on edge devices in an
IoT environment. For example, the software application is an
application for performing data analytics on the edge device.

Alternatively, the software application can be any other application for performing a function on the edge device. The software application can be new application or upgrade patch. The method comprises generating a plurality of simulation
5 instances capable of simulating behavior of the software application on one or more edge devices in the IoT environment. Each simulation instance is configured with a unique resource configuration. The method comprises processing the software application on said each simulation
10 instance using data from an industrial plant, and computing an optimum resource configuration associated with the software application based on processing of the software application on the simulation instances. The optimal resource configuration associated with the software application may be
15 computed by determining at least one simulation instance from the plurality of simulation instances on which the behavior of the software application is satisfactory, and determining the unique resource configuration associated with the determined simulation instance, wherein the unique re-
20 source configuration (404B) is the optimum resource configuration. The optimum resource configuration indicates least resources required for executing the software application on the one or more edge devices. Accordingly, the software application is provisioned on at least one edge device based
25 optimum resource configuration.

In a preferred embodiment, the method may comprise storing the optimum resource configuration in a metafile associated with the software application in an application repository.
30

In another preferred embodiment, the method may comprise evaluating real-time resource capabilities of least one edge device in the IoT environment to execute the software application based on the optimum resource configuration
35 associated with the software application. For example, the real-time resource capabilities indicate real-time resources available on the edge device for executing the software application. The real-time resource may include hardware as

well as software resources. The real-time resources may include hardware capabilities and unutilized resources on the edge device. The method may comprise provisioning the software application on said edge device based on evaluation
5 of the real-time resource capabilities of said edge device.

In yet another preferred embodiment, the method may comprise selecting a set of unique resource configurations from a plurality of unique resource configurations, wherein the set
10 of unique resource configurations correspond to real-time resource capabilities of the edge devices in the IoT environment. The method may comprise generating the plurality of the simulation instances based on the selected set of unique resource configurations. Each unique resource
15 configuration comprises resource constraints influencing the behavior of the software application.

In still another preferred embodiment, the method may comprise simulating the software application on said each
20 simulation instance using the data from the industrial plant based on the unique resource configuration of said each simulation instance. The method may comprise capturing parameter values indicative of the behavior of the software application during simulation of the software application on
25 said each simulation instance. Advantageously, performance of the software application on the edge devices with different resource configurations can be determined prior to provisioning the software application on the edge devices.

30 In further another preferred embodiment, the method may comprise determining at least one simulation instance from the plurality of simulation instances on which the optimum behavior of the software application based on the captured parameters values. The method may also comprise determining
35 the unique resource configuration associated with the determined simulation instance, where the unique resource configuration is the optimum resource configuration.

The object of the present invention is also achieved by a system for provisioning software applications on edge devices in an IoT environment. In an exemplary embodiment, the system may be an IoT cloud platform in a cloud infrastructure
5 communicatively connected to edge devices in a plant. In another exemplary implementation, the system may be a remote server(s) communicatively coupled to the edge devices in the plant. The system comprises one or more processors, a memory, coupled to the processors, which comprises an application
10 repository capable of storing a plurality of software applications. The system also comprises a simulation module configured for generating a plurality of simulation instances capable of simulating behavior of the software application on one or more edge devices in the IoT environment. Each
15 simulation instance is configured with a unique resource configuration. The simulation module is configured for processing the software application on said each simulation instance using data from an industrial plant, and computing an optimum resource configuration associated with the
20 software application on the one or more edge devices based on processing of the software application on the simulation instances. The simulation module is configured to compute the optimum resource configuration associated with the software application by determining at least one simulation instance
25 from the plurality of simulation instances on which the behavior of the software application is satisfactory, and determining the unique resource configuration associated with the determined simulation instance, wherein the unique resource configuration (404B) is the optimum resource configuration.
30 The optimum resource configuration indicates least resources required for executing the software application on the one or more edge devices. Accordingly, the software application is provisioned on at least one edge device based on the optimum resource configuration.

35

In a preferred embodiment, the system may comprise an edge device management module configured for evaluating real-time resource capabilities of least one edge device in the IoT

environment to execute the software application based on the optimum resource configuration associated with the software application. In the preferred embodiment, the system may a provisioning module configured for provisioning the software application on said edge device based on evaluation of the real-time resource capabilities of said edge device.

In another preferred embodiment, the simulation module may be configured for storing the optimum resource configuration in a metafile associated with the software application in the application repository.

In yet another preferred embodiment, the simulation module may be configured for selecting a set of unique resource configurations from a plurality of unique resource configurations. The set of unique resource configurations correspond to real-time resource capabilities of the edge devices in the IoT environment. Each unique resource configuration comprises resource constraints influencing the behavior of the software application. The simulation module may be configured for generating the plurality of the simulation instances based on the selected set of unique resource configurations.

In still another preferred embodiment, the simulation module may be configured for simulating the software application on said each simulation instance using the data from the industrial plant based on the unique resource configuration of said each simulation instance. The simulation module may be also configured for capturing parameter values indicative of the behavior of the software application during simulation of the software application on said each simulation instance.

In further another preferred embodiment, the simulation module may be configured for determining at least one simulation instance from the plurality of simulation instances on which the behavior of the software application is found as satisfactory based on the captured parameters

values. Furthermore, the simulation module may be configured for determining the unique resource configuration associated with the determined simulation instance, where the unique resource configuration is the optimum resource configuration.

5

Also, the object of the present invention is achieved by a computer-program product, having machine-readable instructions stored therein, that when executed by a processor, cause the processor to perform method steps described above. The processor can be a processor of an IoT cloud platform.

The above-mentioned and other features of the invention will now be addressed with reference to the accompanying drawings of the present invention. The illustrated embodiments are intended to illustrate, but not limit the invention.

The present invention is further described hereinafter with reference to illustrated embodiments shown in the accompanying drawings, in which:

- FIG 1 is a schematic representation of an Internet-of-Things (IoT) environment, according to an embodiment of the present invention.
- 25 FIG 2 is a block diagram of an IoT cloud platform as shown in FIG 1, according to the embodiment of the present invention.
- FIG 3 is a flow diagram illustrating an exemplary method of provisioning a software application on an edge device by the IoT cloud platform, according to the embodiment of the present invention.
- 30 FIG 4 illustrates a schematic representation of a simulation environment for computing optimum resource configuration for deploying the software application on the edge device, according to the embodiment of the present invention.
- 35

Various embodiments are described with reference to the drawings, wherein like reference numerals are used to refer to the drawings, wherein like reference numerals are used to refer to like elements throughout. In the following
5 description, for the purpose of explanation, numerous specific details are set forth in order to provide thorough understanding of one or more embodiments. It may be evident that such embodiments may be practiced without these specific details.

10

FIG 1 is a schematic representation of an Internet-of-Things (IoT) environment 100, according to an embodiment of the present invention. The IoT environment 100 comprises an IoT cloud platform 102, a plant 103 comprising edge devices 104A-
15 N and groups of sensors/actuators 116A-N, and user devices 114A-C. Each group of sensors/actuators 116A-N is connected to respective edge devices 104A-N via wired network or wireless network. Each of the edge devices 104A-N is connected to the IoT cloud platform 102 via a network
20 connection such as wide area network. Also, the user devices 114A-C are connected to the IoT cloud platform 102 via the wide area network.

Although not shown, some or all of the edge devices 104A-N
25 can be connected to the IoT cloud platform 102 via an edge gateway. Also, the IoT cloud platform 102 can be connected to edge devices in plants 103 located in different geographical locations.

30 In the plant 103, the groups of sensors/actuators 116A-N are deployed for monitoring and controlling equipments and processes. An example of a sensor may be an infra-red camera deployed in the plant 103. The edge devices 104A-N may be battery-powered devices and equipped with an edge operating
35 system. For example, the edge devices 104A-N are configured for collecting plant data from the respective groups of sensors 116A-N, and analyzing the collected plant data using a set of software applications deployed on the edge devices

104A-N. The edge devices 104A-N may send commands to the respective actuators based on outcome of analysis of the collected plant data. The edge devices 104A-N may also push the raw/processed plant data and/or results of analysis of
5 the plant data to the IoT cloud platform 102.

The IoT cloud platform 102 can be a cloud infrastructure capable of providing cloud-based services such as provisioning of software applications on the edge devices
10 104A-N. The IoT cloud platform 102 can be part of public cloud and/or private cloud. The IoT cloud platform 102 may enable data scientists/software vendors to provide a software application as a service, thereby eliminating need for software maintenance, upgrading, and backup by the users. The
15 software application can be complete application, or software patch. In some embodiments, the software application may be an analytical application for performing data analytics on the edge devices 104A-N. For example, the software application may include an application for down-sampling of
20 time series data, filtering time series data based on thresholds or moving averages, performing Fast-Fourier transform on vibration data and filtering frequencies which indicate anomaly, performing linear regression and trend prediction, local classification using support vector machine
25 classifiers, neural network or deep learning classifiers, performing stream analytics, etc.

The IoT cloud platform 102 is further illustrated in greater detail in FIG 2. Referring to FIG 2, the IoT cloud platform
30 102 comprises a simulation module 106, a provisioning module 108, an edge device management module 110, a processor(s) 202, a memory unit 204, and a communication interface 206. The simulation module 106, the provisioning module 108, and the edge device management module 110 may take a form of
35 hardware such as a processor with embedded software, or a computer program in the form machine-readable instructions stored on a computer program product (e.g., the memory unit 204) and executable by the processor(s) 202. The processor(s)

102 may be one or more processing units (e.g., servers). The processor(s) 102 is also capable of executing machine-readable instructions stored on a computer-readable storage medium such as the memory unit 204 for performing various
5 functions such as processing plant data, analyzing plant data, providing visualization of the analyzed plant data, issuing control commands to the actuators, managing the edge devices 104A-N and so on. The memory unit 204 can be volatile or non-volatile memory. In the preferred embodiment, the
10 memory unit 204 is configured to store an application repository 112. The application repository 112 is capable of storing software applications for provisioning on the edge devices 104A-N. The software applications are stored in the application repository 112 as application binaries along with
15 associated optimum resource configuration in a metafile. The communication interface 206 is configured to establish a communication channel between the edge devices 104A-N and the IoT cloud platform 102. Also, the communication interface 206 is configured to establish a communication channel between
20 the IoT cloud platform 102 and the user devices 114A-C.

The simulation module 106 is configured for simulating behavior of a software application on a plurality of simulation instances in a simulation environment. Each
25 simulation instance represents an edge device in the IoT environment 100. Each simulation instance is also configured with an unique resource configuration. The unique resource configuration corresponds to real-time resource capabilities of the edge device. The unique resource configuration
30 comprises resource constraints which has an influence on the behavior of the software application. In an exemplary implementation, the simulation module 106 executes the software application on the plurality of simulation instances in series or in parallel for a pre-determined time period.
35 The simulation module 106 is configured for computing an optimum resource configuration associated with the software application based on the outcome of the simulation of the software application. The optimum resource configuration

indicates least resources required for executing the software application on the edge devices 104A-N in the IoT environment 100. The simulation module 106 is configured for storing the optimum resource configuration in a metafile associated with the software application in the application repository 112.

The edge device management module 110 is configured for evaluating real-time resource capabilities of the edge devices 104A-N to execute the software application based on the optimum resource configuration associated with the software application. For example, the resources can be hardware as well as software resources. The resource can be network resources, storage resources, computational resources, etc. The optimum resource configuration indicates minimum resources required to execute the software application on the edge devices 104A-N in real-time. The provisioning module 108 is configured for provisioning the software application on one or more edge devices 104A-N if the one or more edge devices 104A-N has sufficient real-time resource capabilities to execute the software application. The steps performed by the IoT cloud platform 102 to simulate software applications and later provision the software applications on the edge devices 104A-N are explained in greater detail in following description.

The user devices 114A-C may be cloud-aware device such as laptop, desktop, tablet computer, smart phone and the like devices. The user devices 114A-C are configured for accessing the IoT cloud platform 102 using a cloud user interface via the Internet. In one embodiment, the user devices 114A-C are configured for viewing the raw/processed plant data and/or results of analysis of the plant data via the cloud user interface. In another embodiment, the user devices 114A-C are configured for uploading software applications to be provisioned on the edge devices 104A-N in the application repository 112. For example, the software applications are typically created by data scientists or software vendors for targeted use cases and uploaded on the IoT cloud platform 102

for provisioning on the edge devices 104A-N. In yet another embodiment, the user devices 114A-C are configured for requesting computation of optimum resource configuration through simulation of a desired software application and provisioning the software application on one of the edge devices 104A-N in the plant 103 based on the optimum resource configuration. In still another embodiment, the user devices 114A-C may act as a portable human-machine interface device for issuing commands to the actuators 116A-N via the edge devices 104A-N based on the results of the analysis of the plant data.

FIG 3 is a process flowchart 300 illustrating an exemplary method of provisioning a software application on the edge device 104A by the IoT cloud platform 102, according to the embodiment of the present invention. At step 302, a plurality of simulation instances capable of simulating behavior of a software application on the edge devices 104A-N is generated. For example, the software application may be analytical application for analyzing real-time video streams from infrared cameras deployed in the plant 103 to determine quality of a manufactured product. In the preferred embodiment, a set of unique resource configuration is selected from a plurality of unique resource configurations. The set of unique resource configurations are selected such that resource configurations correspond to real-time resource capabilities of the edge devices 104A-N in the IoT environment 100. Each of the selected unique resource configurations consists of resource constraints which has an influence on behavior of the software application during simulation. Exemplary resource constraints may include read-access memory (RAM) size, processor type, storage size, Bus size, etc. In this preferred embodiment, the plurality of simulation instances is generated based on the selected set of unique resource configurations. For example, the simulation instances are configured with corresponding unique resource configurations. The number of simulation instances and associated resource

configurations are formulated based on the type of software application to be simulated.

At step 304, the software application is processed on the
5 simulation instances using data from the plant 103. For
example, data from the plant 103 can be real-time data or
historical data collected from the plant 103. The data for
analyzing product quality may include real-time video streams
obtained from infra-red cameras. In a preferred embodiment,
10 behavior of the software application is simulated by
executing the software application on the simulation
instances using the data from the plant 103. In an exemplary
implementation, the software application is executed in
parallel on the simulation instances for a pre-determined
15 time period. Alternatively, the software application is
executed on the simulation instances for the predetermined
time period in a pre-defined order. In the preferred
embodiment, parameter values indicative of the behavior of
the software application during simulation are captured for
20 each simulation instance. For example, the parameter values
captured for each simulation instance are key performance
indicators which indicate performance of the software
application when executed on an edge device with the
associated unique resource configuration.

25
At step 306, an optimum resource configuration associated
with the software application is computed based on processing
the software application on the simulation instances. In a
preferred embodiment, a simulation instance is determined
30 from the plurality of simulation instance based on the
captured parameter values. In other words, the simulation
instance on which optimum behavior of the software
application is determined as best simulation instance.

35 In this preferred embodiment, the unique resource
configuration associated with the simulation instance is
determined. This unique resource configuration is considered
as the optimum resource configuration for executing the

software application. The optimum resource configuration is a resource configuration which resulted in satisfactory performance of the software application when the software application is simulated on the associated simulation
5 instance. The optimum resource configuration indicates least recommended resource configuration that shall be available on an edge device for satisfactory performance of the software application on the edge device. Thus, the optimum resource configuration is the least resource configuration that the
10 edge device must possess for provisioning the software application on the edge device.

At step 308, the optimum resource configuration is stored in a metafile associated with the software application in the
15 application repository 112. At step 310, real-time resource capabilities of at least one edge device 104A to execute the software application are evaluated based on the optimum resource configuration associated with the software application.

20 In the preferred embodiment, real-time resources available for execution of the software application are obtained from the edge device 104A. Also, the optimum resource configuration indicating minimum resources required for
25 satisfactory performance of the software application when deployed in the edge device 104A is obtained from the metafile associated with the software application. It is determined whether the real-time resources available are greater than or equal to the minimum resources required for
30 provisioning the software application. If the real-time resources available are greater than or equal to the minimum resources required, then at step 312, the software application is provisioned on the edge device 104A. The software application is then installed and executed in
35 runtime on the edge device 104A using the real-time resources available on the edge device 104A.

If the real-time resources available are less than the minimum resources required, then the edge device 104A may free-up resources by uninstalling undesired software applications. Alternatively, the IoT cloud platform 102 may
5 customize the software application in such a manner that the optimum resource configuration associated with the software application is less than or equal to the real-time resources available on the edge device 104A.

10 FIG 4 illustrates a schematic representation of a simulation environment 400 for computing an optimum resource configuration for deploying a software application on the edge devices 104A-N, according to the embodiment of the present invention.

15 Consider that an analytical application is developed for analyzing real-time video streams of a product obtained from infra-red cameras in the plant 103 and identifying deviations in product quality in real-time. The analytical application
20 may be implemented as a 3 layered Convolutional Neural Network (CNN) model with about hundred nodes in each layer. The CNN model is created by data scientists using training video data. The analytical application is designed for deploying on one or more edge devices 104A-N in the plant 103
25 which interfaces with the infra-red cameras to obtain infra-red video streams of a manufactured product and computes deviations in quality of the product. Typically, prior to deploying the analytical application, it is not clear as whether the analytical application would run smoothly without
30 any performance issues if deployed on an edge device 104A. Therefore, the present invention simulates the behavior of the analytical application in the simulation environment 400 and computes an optimum resource configuration for the analytical application required for deploying on the edge
35 device 104A as described below.

According to the preferred embodiment, the analytical application is added as a cloud container (e.g., Docker™

Hub) in the application registry 112 for provisioning on one or more edge devices 104A-N. The simulation module 106 prepares the simulation environment 400 comprising simulation instances 402A-N for simulating the behavior of the analytical application. For the purpose of illustration, consider that the simulation environment 400 includes a simulation instance 402A, a simulation instance 402B and a simulation instance 402C.

10 The simulation module 106 configures the simulation instance 402A with an unique resource configuration 404A. The unique resource configuration 404A comprises resource constraints: 64 MB RAM, 512 MHz with single core processor, 512 MB storage, and 128 MHz internal communication bus. Similarly, 15 the simulation module 106 configures the simulation instance 402B with an unique resource configuration 404B. The unique resource configuration 402B comprises resource constraints: 512 MB RAM, 1 GHz with dual-core processor, 2 GB storage, 1 GHz internal communication bus, and 512 MB Graphical 20 Processing Unit (GPU). Also, the simulation module 106 configures the simulation instance 402C with an unique resource configuration 404C. The unique resource configuration 404C comprises resource constraints: 2 GB RAM, 2 GHz with eight cores processor, 4 GB Solid State Drive, 2 25 GHz internal communication bus, 2 GB GPU and a tensor processing unit. It can be seen that the resource configuration 404A of the simulation instance 402A is typical to that of an embedded device with low resource capabilities. On the contrary, the simulation instance 402C has the unique 30 resource configuration 404C as that of a powerful edge device with built-in GPU and Tensor Processing unit and high resource capabilities.

The simulation module 106 obtains the cloud container of the analytical application from the application repository 112 35 and deploys the cloud container of the analytical application simultaneously on the simulation instances 402A-C. The simulation instances 402A-C executes the analytical

application according to the respective unique resource configuration for a specific time period (e.g., 48 hours) using infra-red video stream data. In an exemplary implementation, the simulation instances 402A-C may have
5 Docker™ runtime to run the cloud container of the analytical application. The cloud container may also contain the infra-red video stream data. During this time period, the simulation module 106 monitors the behavior of the analytical application being simulated on the simulation instances 402A-
10 C. The simulation module 106 computes parameter values indicative of the behavior of the analytical application on each of the simulation instances 402A-C. The parameter values may indicate accuracy of output of the analytical application, average/peak load on the processor, average/peak
15 read and writes, average/peak utilization of GPU/Tensor Processing Unit, and so on.

The simulation module 106 checks whether the parameter values associated with each of the simulation instances are within a
20 pre-determined threshold level. For example, if the parameter value indicates that average utilization of the processor is 90% on a particular simulation instance (e.g., the simulation instance 402A) whereas the predetermined threshold for utilization of processor is 75%. In such case, the behavior
25 or performance of the analytical application on the specific simulation instance (e.g., the simulation instance 402A) is said to be non-satisfactory or unacceptable. This may lead to performance issues when the analytical application is run on the edge device having such configuration as that of the
30 specific simulation instance. In the current example, the analytics application is computation expensive and hence the performance of the analytical application is non-satisfactory on the simulation instance 402A with low resource configuration 404A.

35 Based on the parameter values, the simulation module 106 identifies one or more simulation instances from the simulation instances 402A-C on which the behavior of the

analytical application is found to be satisfactory. In other words, the parameter values computed for the simulation instances are within the predetermined threshold level. Consider that, the simulation module 106 identifies the
5 simulation instances 402B and 402C on which the behavior of the analytical application is satisfactory.

The simulation module 106 determines a simulation instance which has lowest unique resource configuration among the
10 simulation instances 402B and 402C. In the current case, the simulation module 402 identifies the unique resource configuration 404B associated with the simulation instance 402B as an optimum resource configuration for deploying the analytical application on the edge devices 104A-N. The
15 optimum resource configuration is a least recommended resource configuration that an edge device shall have for being eligible to deploy the analytical application. The simulation module 106 stores the optimum resource configuration in a metafile associated with the analytical
20 application in the application repository 112.

When a request for provisioning the analytical application on an edge device 104A is received, the edge device management module 110 obtains real-time resource availability
25 information from the edge device 104A. The edge device management module 110 evaluates real-time resource capabilities of the edge device 104A to execute the analytical application in real-time based on the real-time resource availability information and the optimum resource
30 configuration. If the edge device 104A has the optimum resource configuration, then the provisioning module 108 provisions the analytical application to the edge device 104A. This helps ensure that the performance of the analytical application when executed on the edge device 104A
35 would be satisfactory.

The present invention can take a form of a computer program product comprising program modules accessible from computer-

usable or computer-readable medium storing program code for use by or in connection with one or more computers, processors, or instruction execution system. For the purpose of this description, a computer-usable or computer-readable
5 medium can be any apparatus that can contain, store, communicate, propagate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device. The medium can be electronic, magnetic, optical, electromagnetic, infrared, or semiconductor system
10 (or apparatus or device) or a propagation mediums in and of themselves as signal carriers are not included in the definition of physical computer-readable medium include a semiconductor or solid state memory, magnetic tape, a removable computer diskette, random access memory (RAM), a
15 read only memory (ROM), a rigid magnetic disk and optical disk such as compact disk read-only memory (CD-ROM), compact disk read/write, and DVD. Both processors and program code for implementing each aspect of the technology can be centralized or distributed (or a combination thereof) as
20 known to those skilled in the art.

While the present invention has been described in detail with reference to certain embodiments, it should be appreciated that the present invention is not limited to those
25 embodiments. In view of the present disclosure, many modifications and variations would be present themselves, to those skilled in the art without departing from the scope of the various embodiments of the present invention, as described herein. The scope of the present invention is,
30 therefore, indicated by the following claims rather than by the foregoing description. All changes, modifications, and variations coming within the meaning and range of equivalency of the claims are to be considered within their scope. All advantageous embodiments claimed in method claims may also be
35 apply to system/apparatus claims.