US 20140104778A1

(19) **United States**

(12) **Patent Application Publication**

Schnell et al.

(10) **Pub. No.: US 2014/0104778 A1**

(43) **Pub. Date: Apr. 17, 2014**

(54) **SYSTEM AND METHOD FOR FLEXIBLE STORAGE AND NETWORKING PROVISIONING IN LARGE SCALABLE PROCESSOR INSTALLATIONS**

(71) Applicant: **CALXEDA, INC.**, Austin, TX (US)

(72) Inventors: **Arnold T. Schnell**, Pflugerville, TX (US); **Richard Owen Waldorf**, Austin, TX (US); **David Borland**, Austin, TX (US)

(73) Assignee: **CALXEDA, INC.**, Austin, TX (US)

(21) Appl. No.: **14/106,698**

(22) Filed: **Dec. 13, 2013**

Related U.S. Application Data

(62) Division of application No. 13/284,855, filed on Oct. 28, 2011.

Publication Classification

(51) **Int. Cl.**
G06F 1/18 (2006.01)

(52) **U.S. Cl.**
CPC ***G06F 1/189*** (2013.01)
USPC **361/679.31**

(57) **ABSTRACT**

A system and method for provisioning within a system design to allow the storage and IO resources to scale with compute resources are provided.

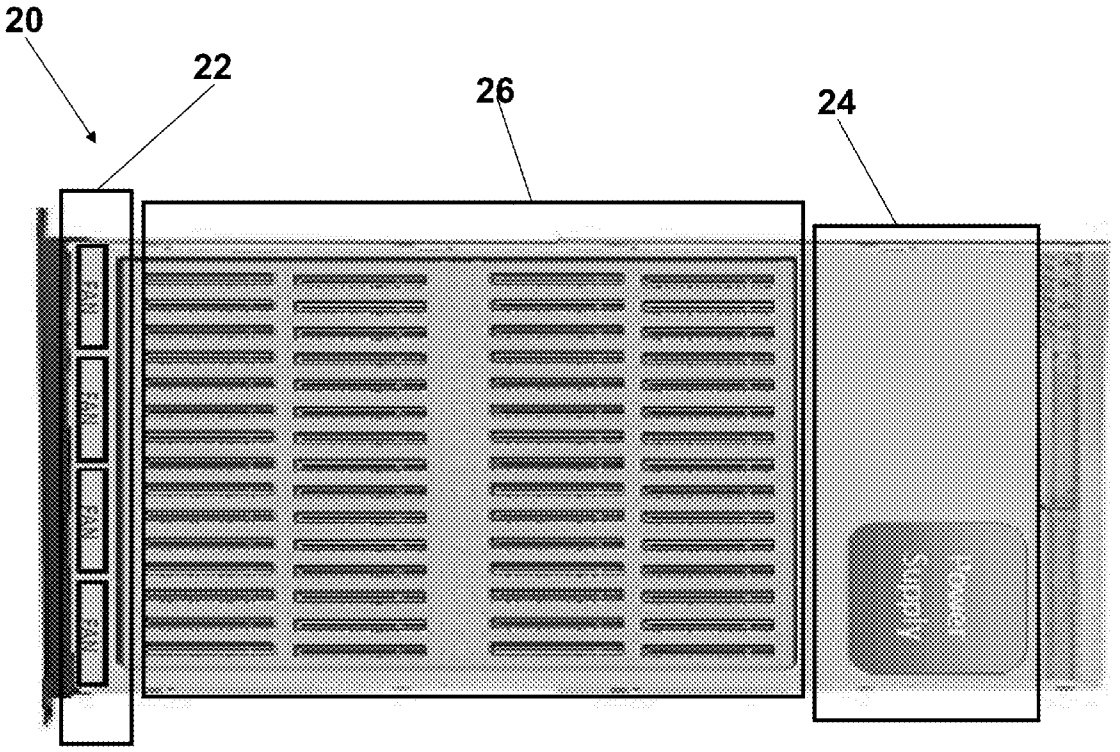

Figure 1

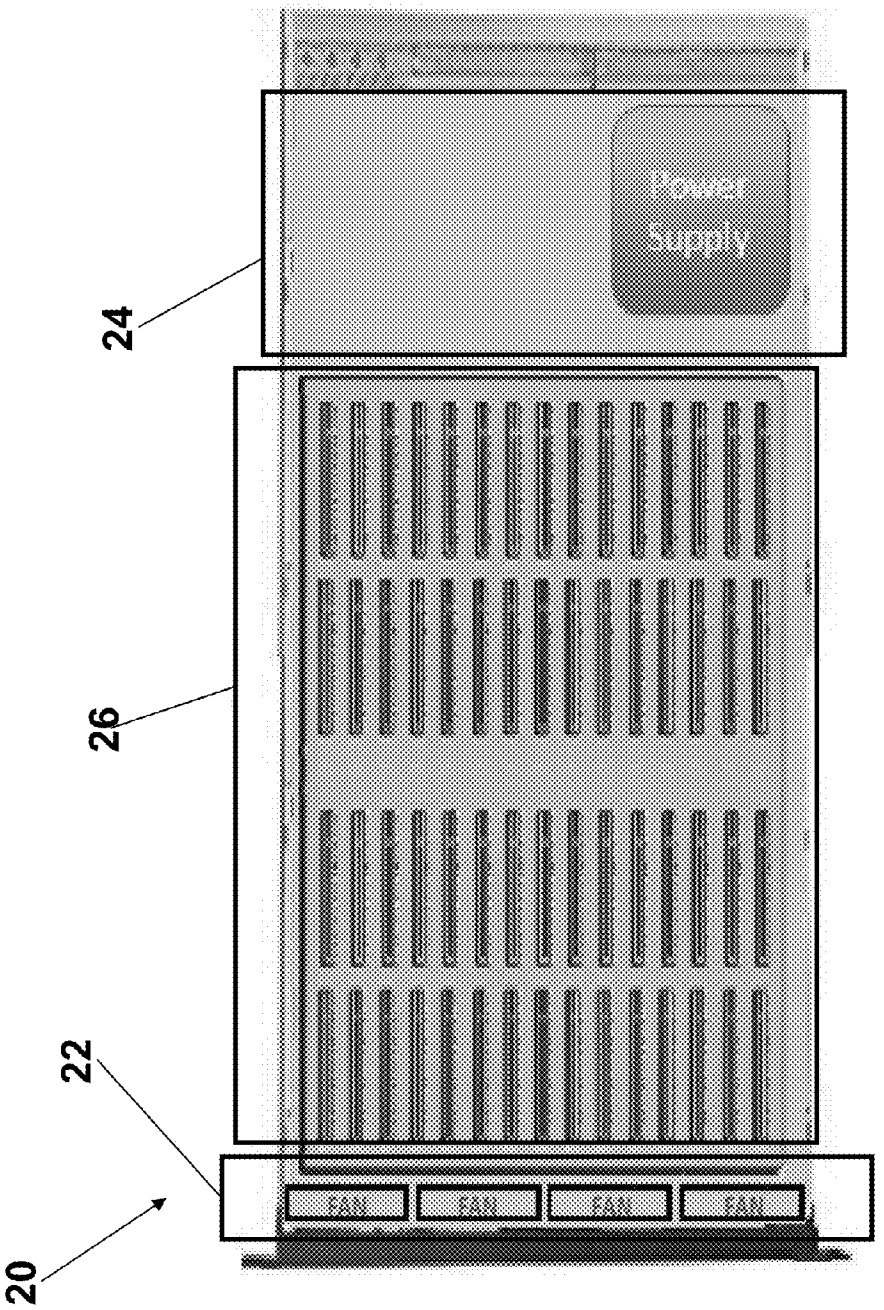

Figure 2

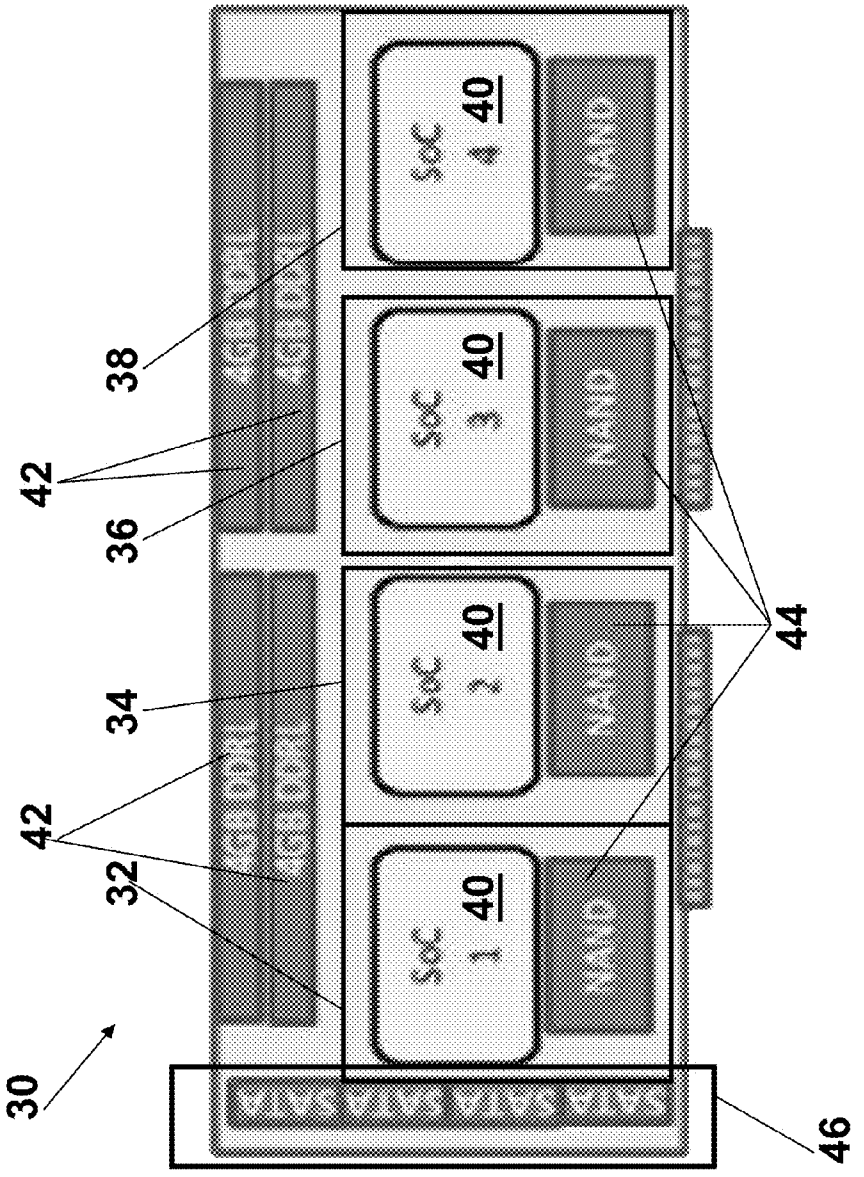
30
32
34
36
38
42
42
44
46
SoC 1 40
SoC 2 40
SoC 3 40
SoC 4 40
NAND
NAND
NAND
NAND
SATA SATA SATA SATA

Figure 3

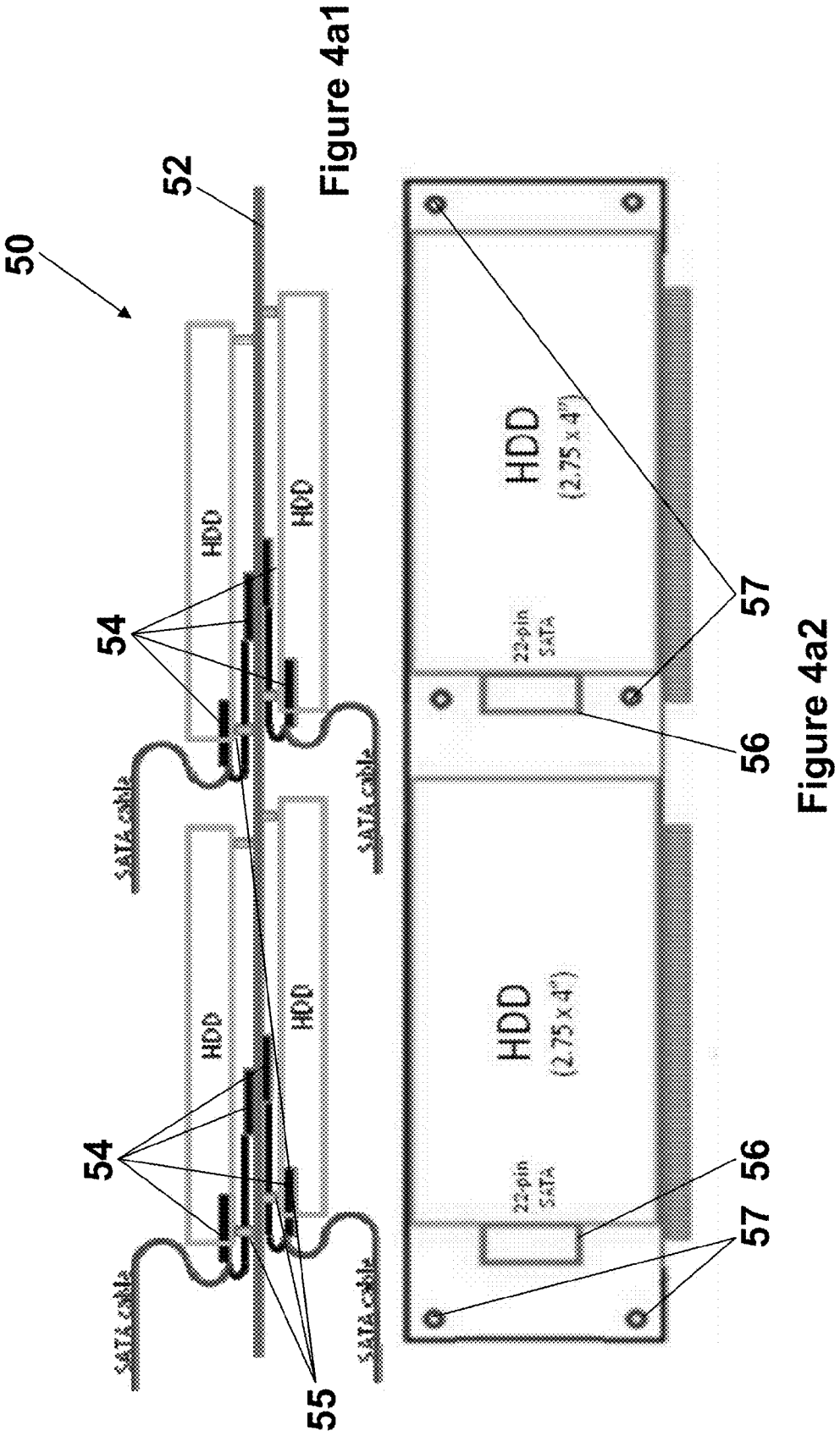

Figure 4a1

Figure 4a2

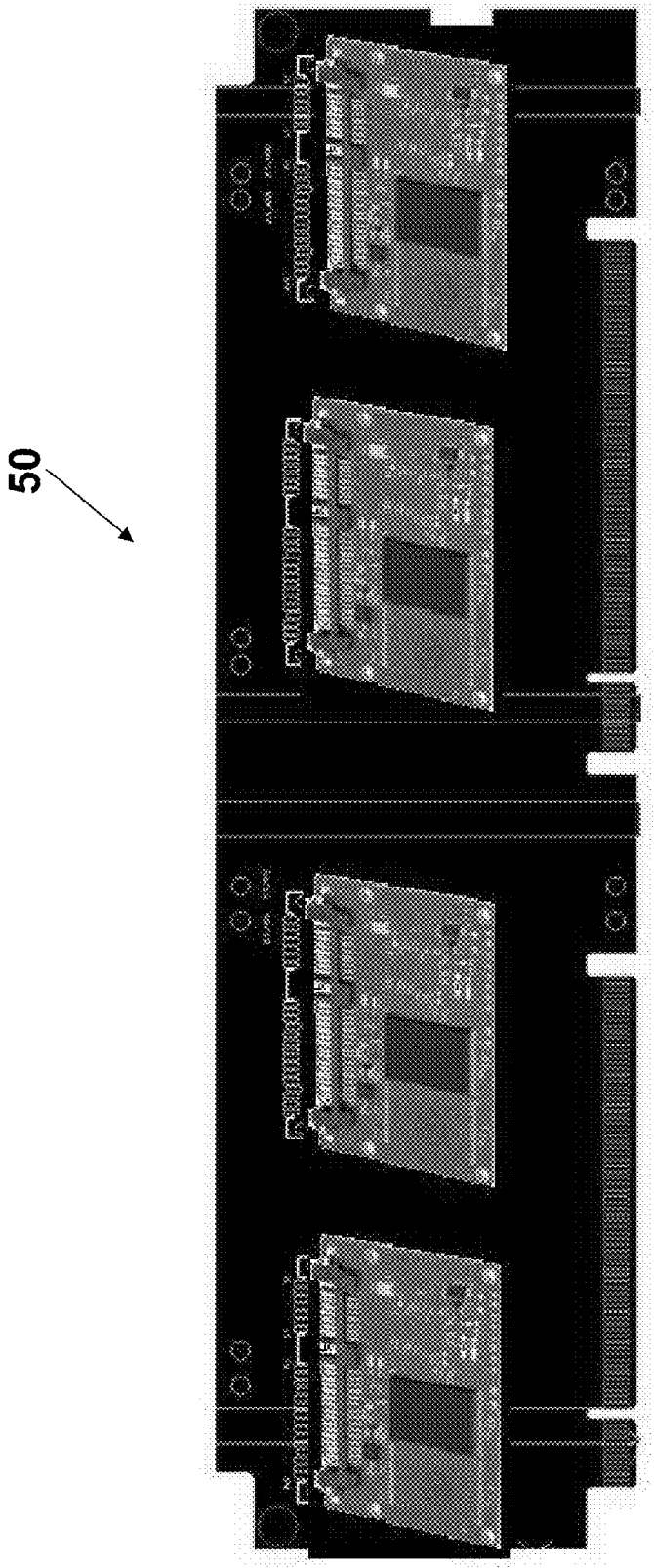

Figure 4b

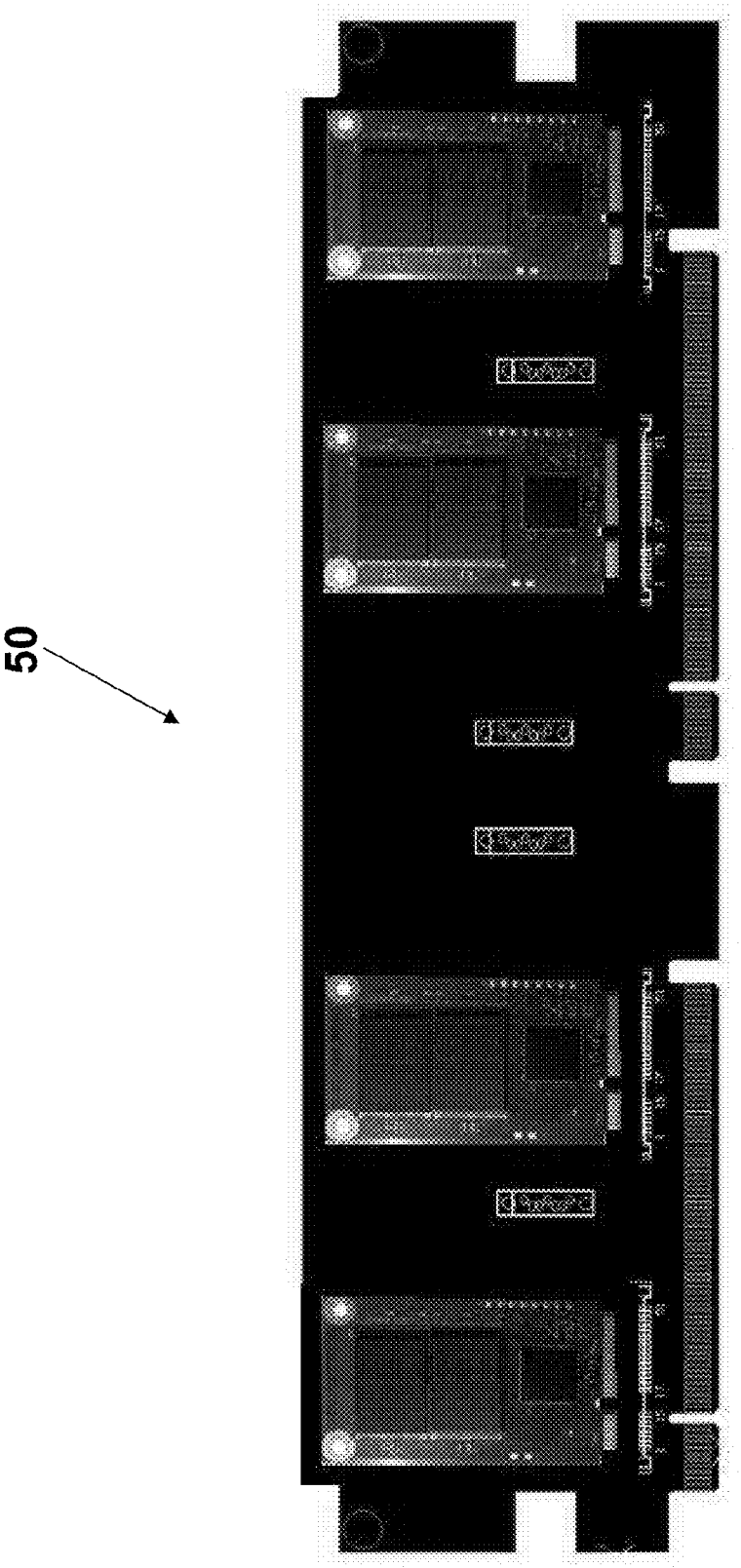

Figure 4c

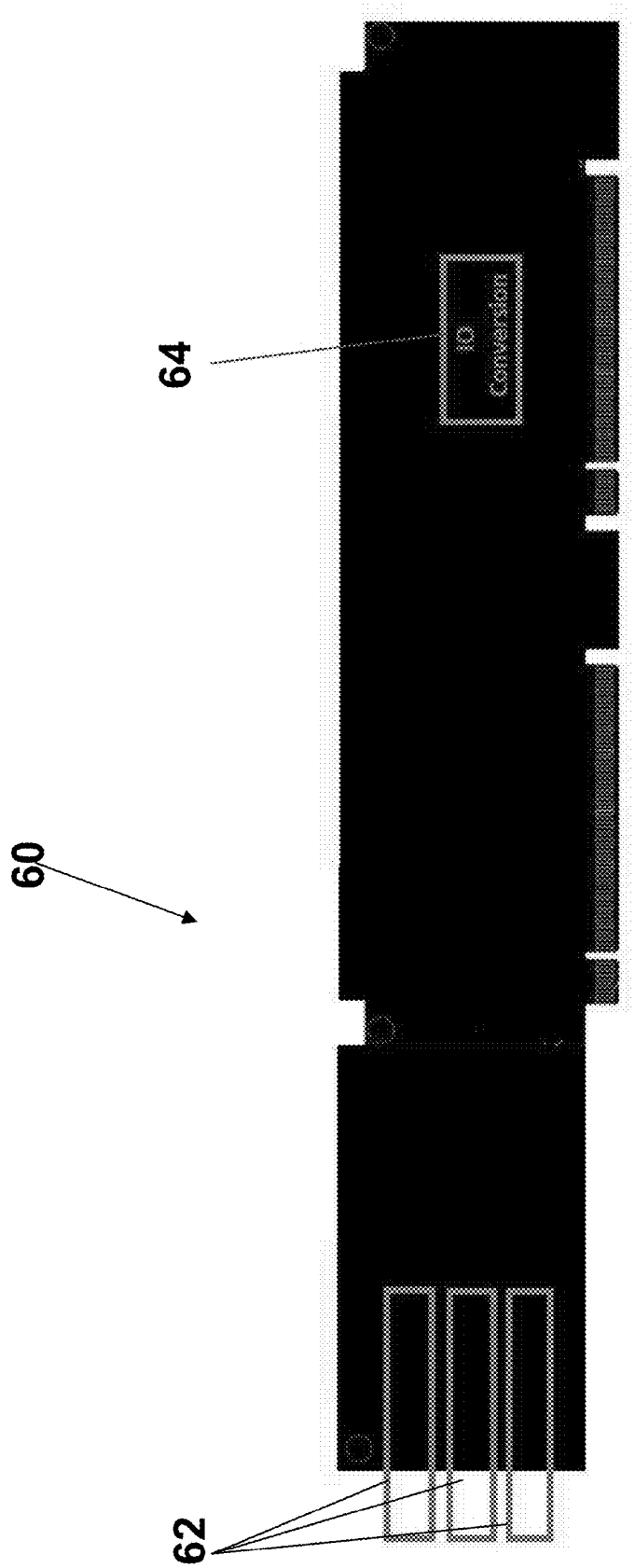

Figure 5

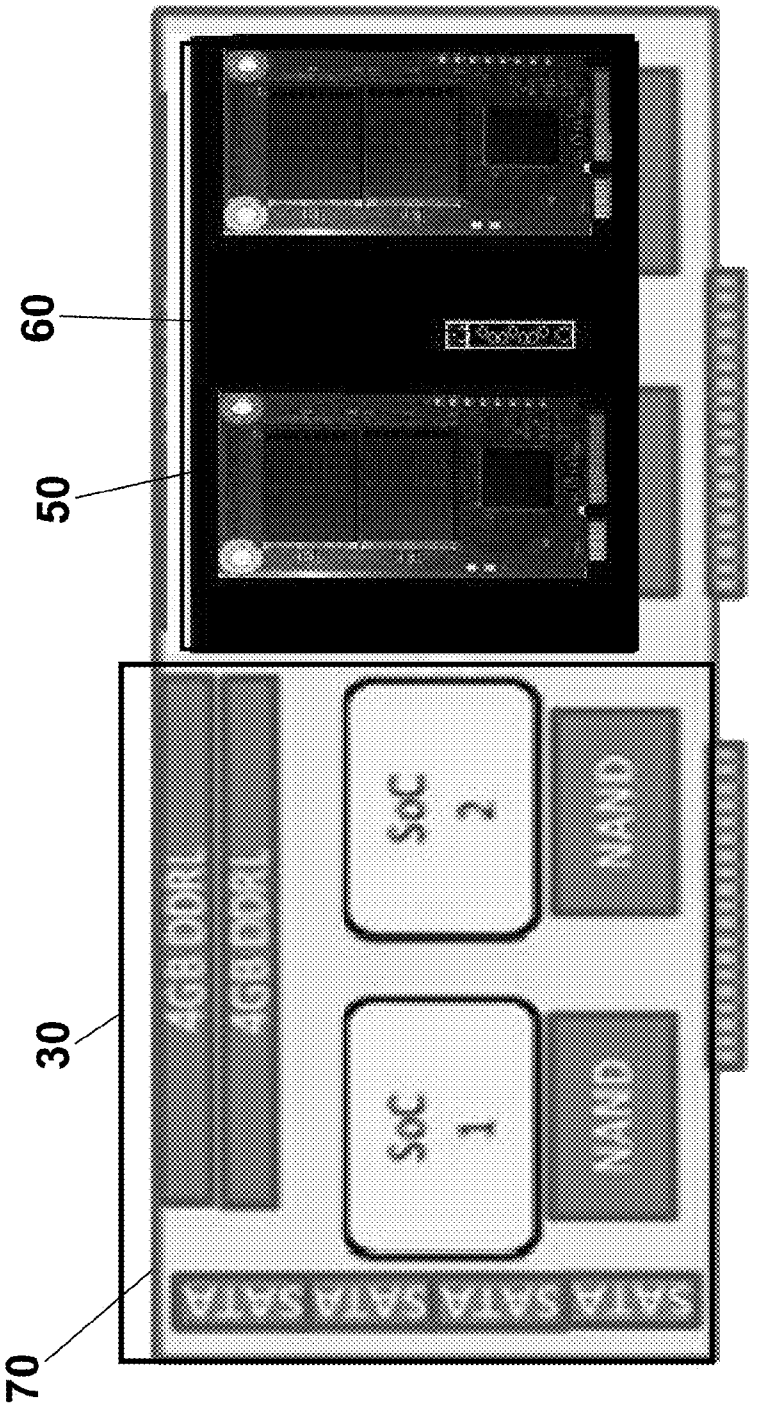

Figure 6

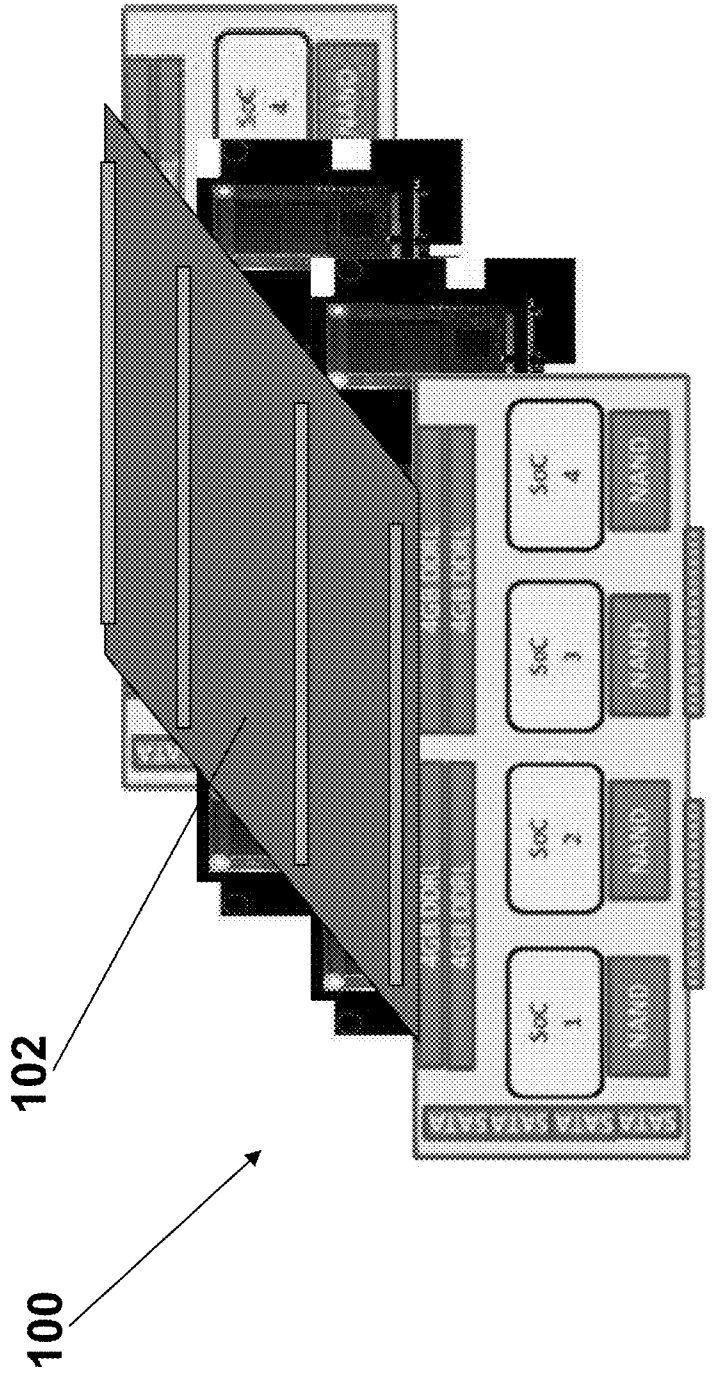

Figure 7

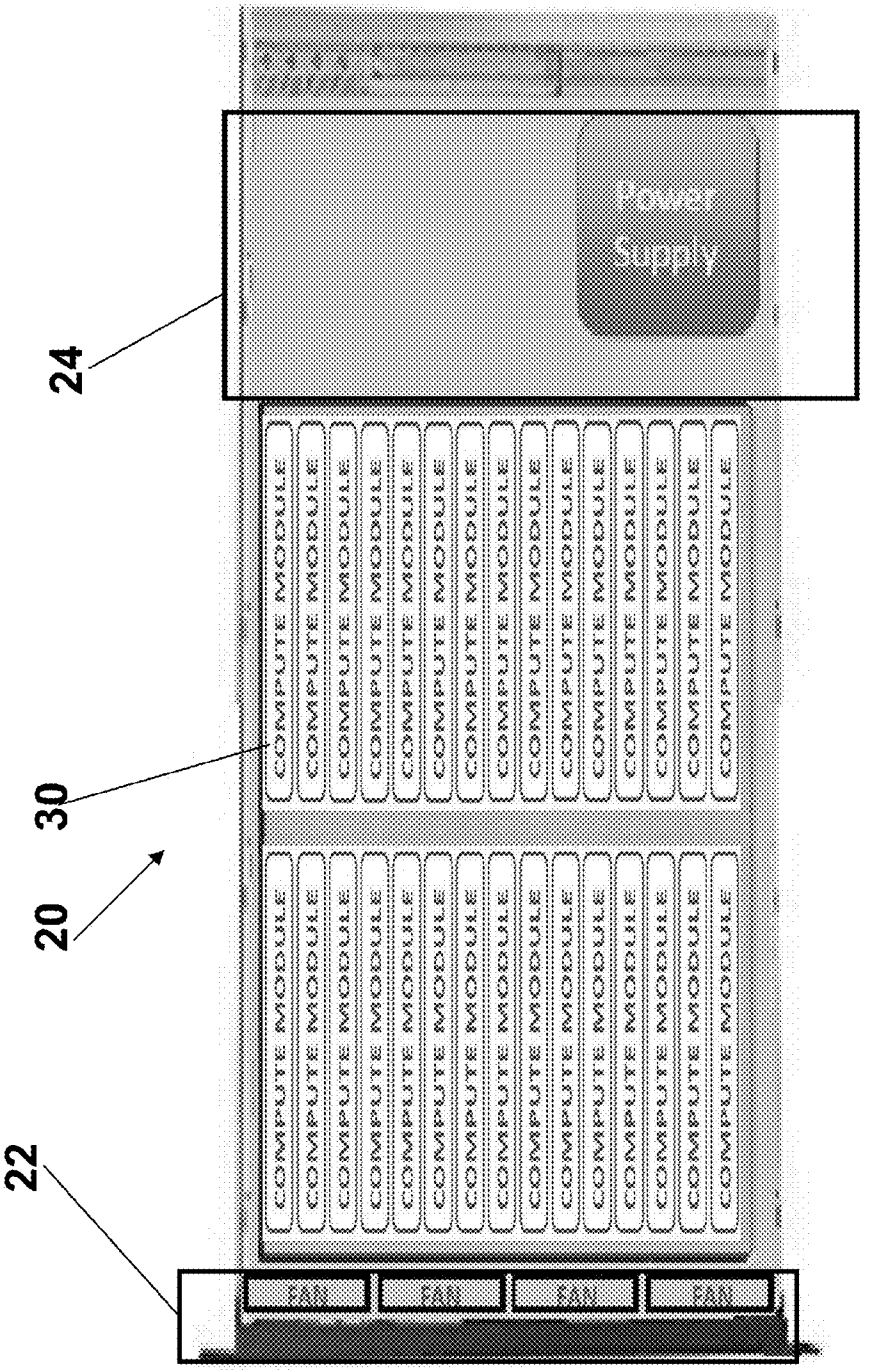

Figure 8a

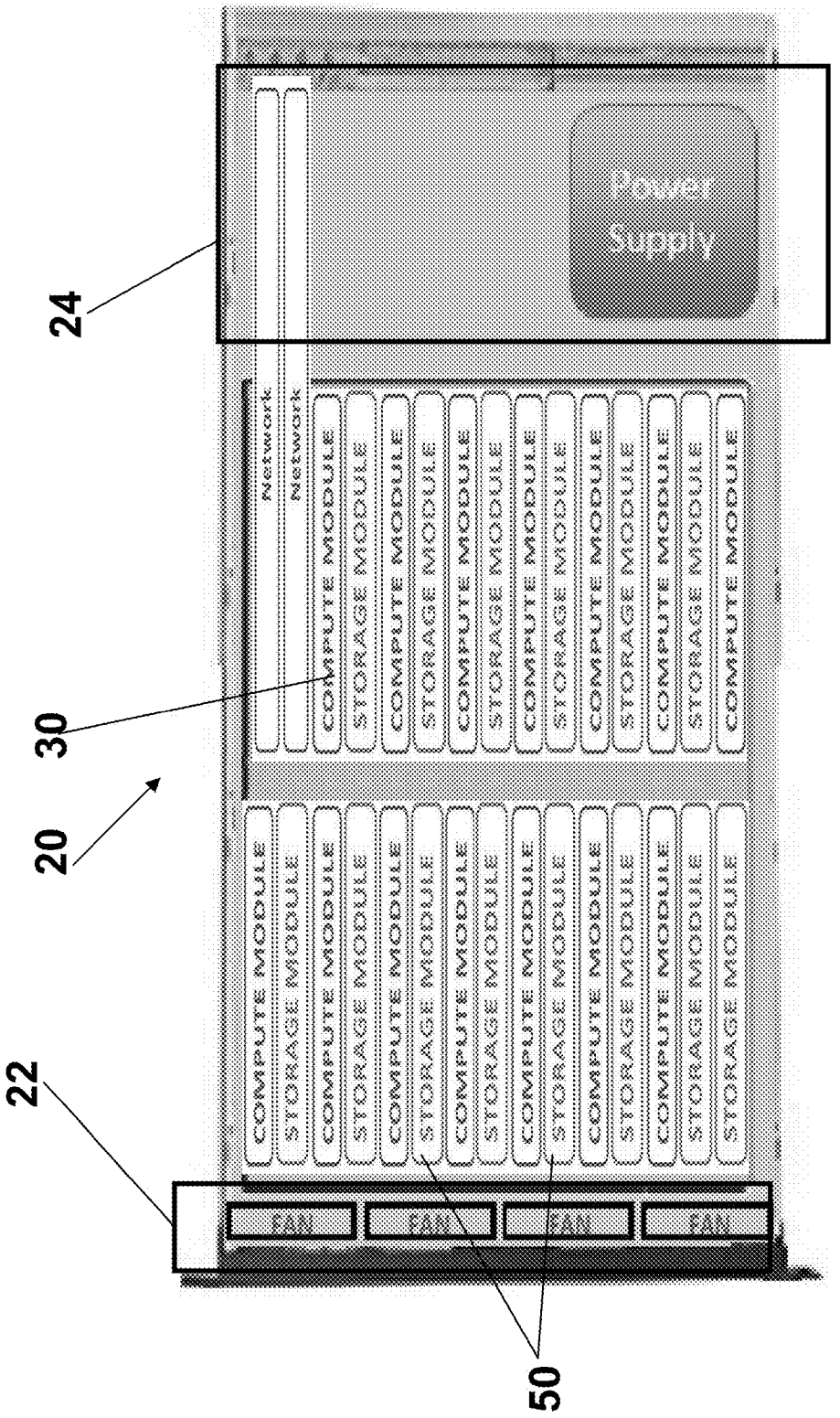
24
Power Supply
Network
Network
30
COMPUTE MODULE
STORAGE MODULE
20
22
FAN
50

Figure 8b

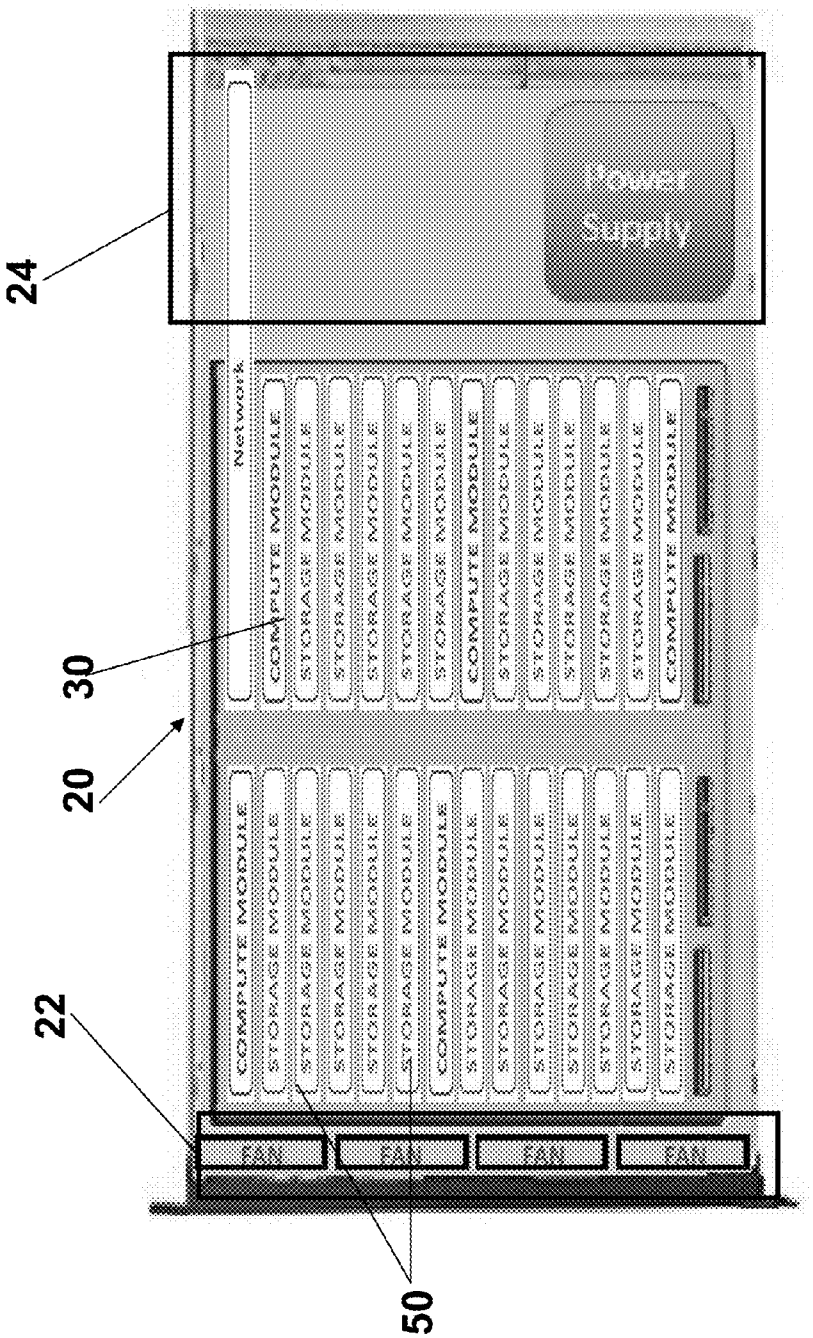

Figure 8c

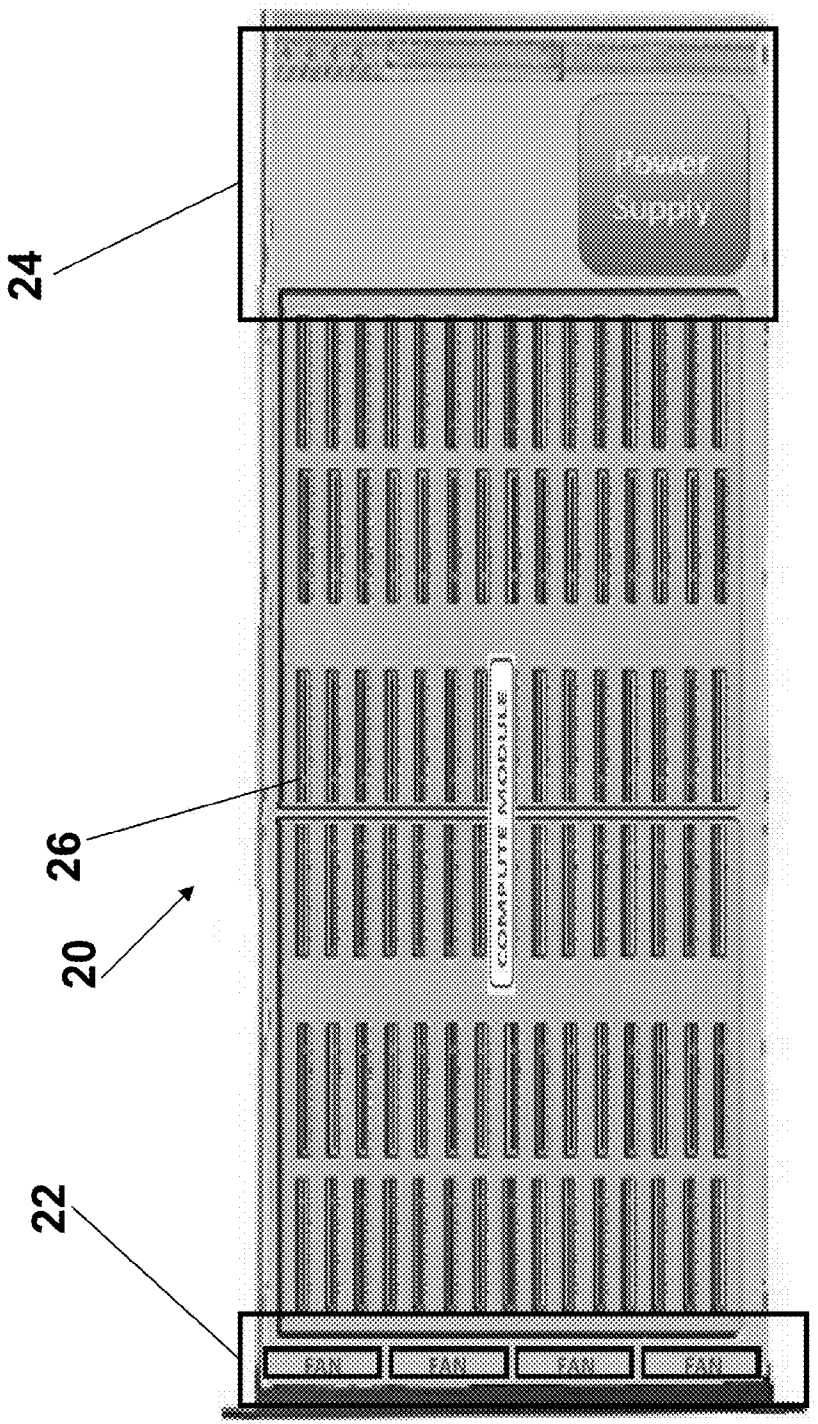

Figure 8d

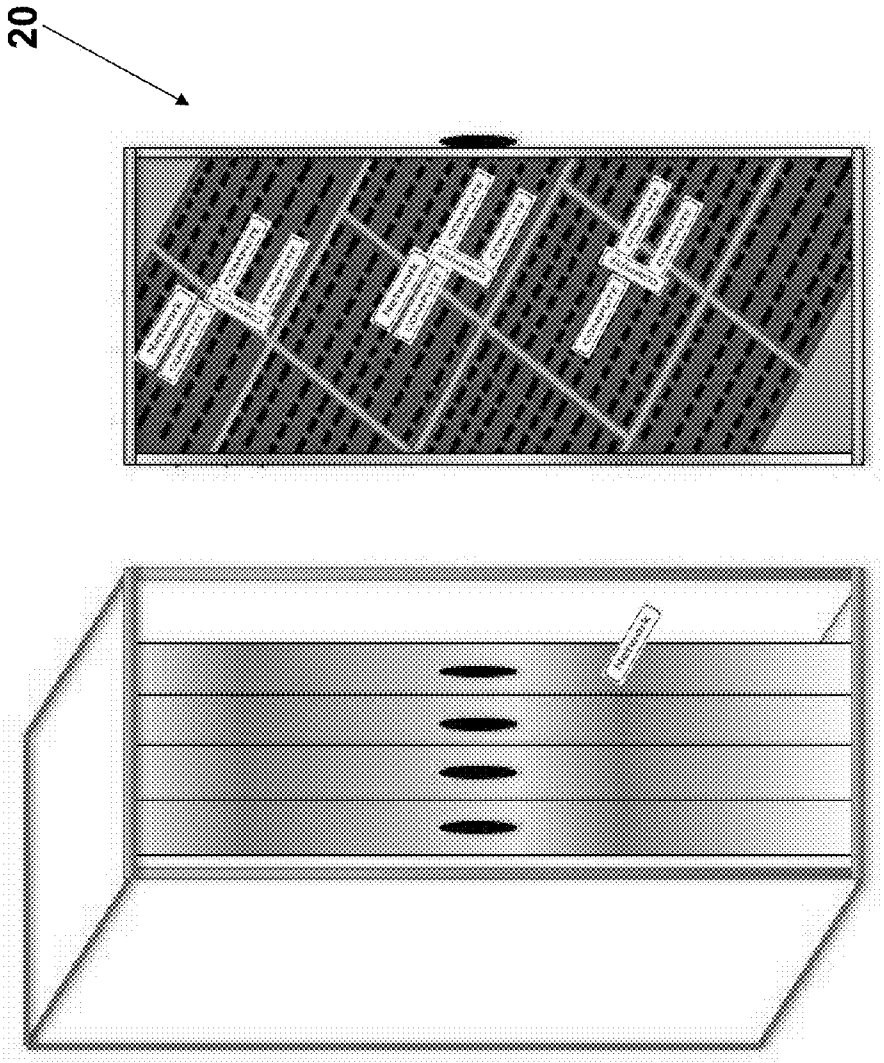

Figure 8e

SYSTEM AND METHOD FOR FLEXIBLE STORAGE AND NETWORKING PROVISIONING IN LARGE SCALABLE PROCESSOR INSTALLATIONS

[0001] This application is a divisional application and claims the benefit of U.S. patent application Ser. No. 13/284,855 filed on Oct. 28, 2011, the disclosure of which is incorporated herein by reference.

FIELD

[0002] The disclosure relates generally to provisioning within a system design to allow the storage and networking resources to scale with compute resources.

BACKGROUND

[0003] Server systems generally provide a fixed number of options. For example, there are a fixed number of PCI Express IO slots and a fixed number of hard drive bays, which often are delivered empty as they provide future upgradability. The customer is expected to gauge future needs and select a server chassis category that will serve present and future needs. Historically, and particularly with x86-class servers, predicting the future needs has been achievable because product improvements from one generation to another have been incremental.

[0004] With the advent of scalable servers, the ability to predict future needs has become less obvious. For example, in the class of servers within a 2U chassis, it is possible to install 120 compute nodes in an incremental fashion. Using this server as a data storage device, the user may require only 4 compute nodes, but may desire 80 storage drives. Using the same server as a pure compute function focused on analytics, the user may require 120 compute nodes and no storage drives. The nature of scalable servers lends itself to much more diverse applications which require diverse system configurations. As the diversity increases over time, the ability to predict the system features that must scale becomes increasingly difficult.

[0005] An example of a typical server system is shown in FIG. **1**. The traditional server system has fixed areas for 24 hard drives along its front surface and a fixed area for compute subsystem (also called motherboard) and a fixed area for IO expansion (PCI slots). This typical server system does not provide scalability of the various computer components. Thus, it is desirable to create a system and method to scale storage and networking within a server system and it is to this end that this disclosure is directed. The benefit of this scalability is a much more flexible physical system that fits many user applications.

BRIEF DESCRIPTION OF THE DRAWINGS

[0006] FIG. **1** illustrates a traditional server system, depicting fixed areas for 24 hard drives along its front surface and a fixed area for compute subsystem (also called motherboard) and a fixed area for IO expansion (PCI slots).

[0007] FIG. **2** illustrates an exemplary system with multiple slots that can house a compute module, a storage module, or an IO module.

[0008] FIG. **3** illustrates an exemplary compute module.

[0009] FIGS. **4***a***1** and **4***a***2** are a side view and a top view, respectively, of an exemplary storage module which implements industry standard 2.5" hard drives or SSDs (solid state drives).

[0010] FIG. **4***b* illustrates an exemplary storage module which implements SATA SSD modules.

[0011] FIG. **4***c* illustrates an exemplary storage module which implements mSATA SSD modules.

[0012] FIG. **5** illustrates an exemplary IO module.

[0013] FIG. **6** illustrates an exemplary hybrid module.

[0014] FIG. **7** illustrates a module block (or super module) made up of an integrated collection of modules connected together by way of a private interconnect.

[0015] FIG. **8***a* illustrates an example of how the exemplary system can be populated specifically for high compute applications which require no local storage.

[0016] FIG. **8***b* illustrates an example of how the exemplary system can be populated with a 1:1 ratio of mix of compute and storage. These are useful, for example, for Hadoop applications.

[0017] FIG. **8***c* illustrates another example of how the exemplary system can be populated specifically for storage applications.

[0018] FIG. **8***d* illustrates an example of a straddle slot. For long chassis', a practical limit is reached on system board size. The center columns of slots straddle across system boards.

[0019] FIG. **8***e* illustrates the use of straddle slots in systems with a much larger system board area.

DETAILED DESCRIPTION OF ONE OR MORE EMBODIMENTS

[0020] The disclosure is particular applicable to a 2U chassis which is the most widely favored form factor for PC-class servers. The concepts herein apply to any chassis form factor, such as tower and rack chassis' of varying customary sizes and any unconventional form. For example, FIG. **8***e* shows an unconventional form factor, the sliding door, which relies on rack rails at the top and bottom of a server rack, rather than left and right sides as used by conventional rack chassis'. The sliding door approach expands the usable space for system boards, but at the same time, it creates a new interconnect problem between system boards that should be solved by the flexible provisioning concepts herein.

[0021] Computer architecture have various components and those components can be categorized in three categories: compute, storage, and IO wherein the compute category may include computing related or processor components, the storage category are storage type devices and IO are input/output components of the computer architecture. Each category can be further subdivided, and each category can be defined to contain certain element types. For example, compute can be subdivided into an ALU, cache, system memory, and local peripherals. Also for example, the storage category can contain element types of hard drives, solid state storage devices, various industry-standard form factors, or non-standard devices. For this disclosure, the component level (compute, storage, IO) are used with the understanding that each component has dimensions and attributes to which the same concepts may be applied.

[0022] The system and method of the disclosure allow the same physical space to be used by any of the computer components: compute devices, storage devices, or IO devices. This provides the greatest flexibility in configuration of sys-

tems for different applications. In addition, devices within the computer system that support all three components, such as power supplies and fans, will be assumed to be stationary for simplicity in the examples provided. It is understood that these support devices do not have to be stationary, depending on the goals in differentiation of the system design, meaning that they also can scale as needed.

[0023] In this example, a "slot" consists of physical connectors and a defined volume of space above these connectors. In one implementation, two PCI Express x16 connectors are used, along with a volume of 10" length by 2.7" height by 1" width. This volume is selected based on associated component heights, the restrictions of a 2U chassis, and a length driven by the PCB space required to accommodate this implementation. It is understood that other connector types can be used, depending on the signaling frequency and quantity of pins required. It is understood that other volumes can be used, depending on the physical constraints that are acceptable for the application. The connector pin definitions are critical to accommodate the many needs of the computer components, both in power delivery and bandwidth of the electrical interfaces. FIG. **2** depicts the resulting example system **20** that has one or more fixed locations **22** in the system for fans, one or more fixed locations **24** for the power supplies, and one or more slots **26** (30 slots in this example) for processors, storage or IO components of the system in which

[0024] An exemplary compute module **30** is shown in FIG. **3**. In support of the principle of scaling, the compute module **30** has one or more nodes, such as four nodes **32-38** in this example. Each node consists of a highly integrated SOC (System On Chip) **40**, associated DIMM **42** for system memory, nonvolatile memory (NAND) **44** for local storage space, one or more known SATA channels **46** for connectivity to storage components and other necessary small devices which are necessary for general functions of the node (EEPROMs, boot flash memory, sensors, etc). The four nodes **32-38** have local IO connections to each other, which provide intercommunication and redundancy if an external IO connection fails. Each of the nodes runs an independent operating system, although as another example, a cache-coherent compute module is possible which would run one instance of an operating system on each node.

[0025] Examples of storage modules **50** that may be used in the system are shown in FIGS. **4***a*, **4***b*, and **4***c*. FIGS. **4***a***1** and **4***a***2** illustrate a storage module that leverages the existing industry-standard 2.5" drive form factor for hard drives (defined to contain spinning mechanical platters which store data) or for solid state drives (defined to have no moving parts and uses integrated circuits for its storage media). In this example, it is possible to use a printed circuit board (PCB) card edge connector for power delivery and/or data delivery using the necessary IO standard, such as SATA or SAS. The IO standard selected is purely a convenience based on support by the implemented devices. Any IO protocol can be routed through this card edge connector as long as the mechanical interface can support the necessary signaling frequency. Alternatively, directly connecting the IO for data delivery to the drive provides further flexibility in system configuration.

[0026] In FIG. **4***a***1**, a printed circuit board **52** is shown to which power/data connectors and voltage regulators are integrated for connection to subsequently attached storage devices. The storage modules also have one or more connectors **54**, such as SATA power connectors, and power cables to connect power from PCB power rails to the attached storage media (in this case, SATA 2.5" mechanical spindle hard drives). In this example, these cables are not needed for SATA SSD nor mSATA. The storage module may also have standoffs **55** that mount the 2.5" SATA HDD to the blue mounting holes in **4***a***2**. The storage module also has the SATA data cable **56** which do not convey power.

[0027] In FIG. **4***a***2**, the storage module has a set of SATA power/data connector **56** that are another method of attaching a hard drive to the PCB. The storage module in FIG. **4***s***2** may also have one or more mounting holes **57** for the standoffs **55** shown in FIG. **4***a***1**. They also include holes used for standard manufacturing of the PCB assembly.

[0028] FIG. **4***b* depicts a storage module that implements an industry-standard 22-pin SATA connector and interface, along with mechanical support features, to support SATA SSD modules per the JEDEC MO-297 standard. FIG. **4***c* depicts a storage module that implements an industry-standard xl PCI connector, along with mechanical support features to support the mSATA modules per the JEDEC MO-300 standard.

[0029] The example in FIG. **4***c* demonstrates an opportunity to expand beyond the industry standard to maximize the benefit of a storage module that can be very close to its associated compute module. The reuse of an xl PCI connector for the mSATA module left many pins unused, as the JEDEC standard had need for only one SATA channel through this interface. In fact, there is space for 5 additional SATA channels, even when allocating pins for sufficient grounding. This allows up to 6 SATA channels, each with smaller memories, as opposed to one SATA channel with one large memory block, although both scenario's can result in the same total storage space. The advantage of the multiple SATA channels is increased interface bandwidth, created by the possibility of parallel access to memory. Given that the operating system can stripe across multiple physical disks to create a single logical disk, the net change is a boost in SATA interface performance. Thus, mSATA modules with greater than one SATA channel can provide a new solution to IO bottlenecks to disks.

[0030] An exemplary IO module **60** for the system is shown in FIG. **5**. Unlike a Network Interface Controller (NIC) that would plug into a conventional server and tie into its operating system, this IO module **60** connects to the infrastructural IO of the system at its edge connectors **62** and provides a translation **64** (using an IO translation circuit) from the internal IO protocol to an external IO protocol, such as Ethernet. The IO module **60** operates independent of any particular operating system of any node. The IO module **60** can support one or many external IO ports, and can take on a form factor that is suitable for a particular chassis design. The benefit of modularity allows the quantity of IO modules to be determined by the bandwidth requirement for data traversing from this system to/from others.

[0031] An exemplary hybrid module **70** is shown in FIG. **6**, demonstrating that a combination of compute **30**, storage **50**, and IO **60** concepts can be implemented on a single module that are then incorporated into the system.

[0032] FIG. **7** illustrates a module block (or super module) **100** made up of an integrated collection of modules **70** connected together by way of a private interconnect **102**.

[0033] With the compute, storage, and IO module concepts described above, exemplary systems of FIG. **8** are now described. FIGS. **8***a*, **8***b*, and **8***c* depict different system configurations to address the basic categories of compute-inten-

sive applications, Hadoop applications, and storage applications respectively. Of course, many other combinations of modules are possible to form the recipe needed for specific applications. As shown, the module form factor is kept consistent for convenience, but when required, it can change also, as shown by the IO module labeled "Network". These degrees of flexibility allow creation of a family of modules that can be mixed and matched according to software application needs, with very little volume within the chassis tied to dedicated purposes. For example, FIG. **8***a* shows a system **20** that has the fans **22** and power supplies **24** and a plurality of compute modules **30** for a compute intensive system. In FIG. **8***b*, the system **20** has the same form factor and the fans and power supplies, but the slots **26** are filled with a combination of compute modules **20** and storage modules **50** as shown for a system that requires more storage than the system in FIG. **8***a*. FIG. **8***c* illustrates a system **20** has the same form factor and the fans and power supplies, but the slots **26** are filled a few compute modules **20** and many more storage modules **50** as shown for a system that requires more storage than computing power than the systems in FIGS. **8***a* and **8***b*.

[0034] FIG. **8***d* expands on the system **20** concepts by considering a chassis that is particularly long, such that the system board size is larger than the practical limit allowed by PCB fabrication factories. Typical PCB panel sizes are 18"×24" or 24"×24", although panels up to 30" are also available with limited sources. Given a typical 2U chassis that fits in a 19" wide rack, the 18"×24" PCB panel is the preferred size for most server motherboards today. To expand beyond the 24" limit, board-to-board connectors must be used to interconnect two assemblies. When high speed signaling must pass between the two assemblies, a relatively expensive interconnect solution must be implemented, such as FCI AirMax connectors. The use of these connectors complicates the electrical design by adding signal integrity considerations and complicates the mechanical design due to the volume required for these connectors. Alternatively, the two system boards do not need to be directly connected at all, relying instead on the IO fabric within a Compute module to traverse data between them, called a "straddle slot". In FIG. **8***d*, the left system board might be aligned based on controlled mounting points, while the right system board might be designed to "float" on its mounting points such that installed modules can control the alignment of associated edge connectors.

[0035] FIG. **8***e* breaks away from the 2U chassis example with an exemplary vertical system **20** that greatly expands the area possible for system boards. Each section on rails is referred to as a "vertical chassis". The black dashed lines represent module slots. Note the angled slot orientation enhances air flow due to natural convection, without the consequence of undue heat build-up caused in true vertical chimney rack designs. The straddle slot concept can be employed here to avoid the expense and space requirements of board-to-board high speed connectors. Power and cooling are not shown, as it is self-evident that space in the enclosure can be dedicated to these as needed.

[0036] While the foregoing has been with reference to a particular embodiment of the invention, it will be appreciated by those skilled in the art that changes in this embodiment may be made without departing from the principles and spirit of the disclosure, the scope of which is defined by the appended claims.

1. A printed circuit board, comprising:
- one or more PCIe connectors through which power is routed;
- one or more regulators connected to the printed circuit board that are powered by the one or more PCIe connectors and generate a regulated voltage;
- one of a SATA, mSATA and miniSATA connector connected to the printed circuit board that are powered by the regulated voltage; and
- wherein a storage component can be connected to the connector to power the storage component.

2. The printed circuit board of claim **1**, wherein the storage component is one of a 2.5" cased SATA drive, caseless SATA solid state device and an mSATA solid state device.

3. The printed circuit board of claim **1** further comprising one of a SATA connector, mSATA connector and a miniSATA connector connected to the storage component through which a set of SATA signals from the storage component are communicated.

4. The printed circuit board of claim **1** further comprising one of a SATA connector, mSATA connector and a miniSATA connector connected to the storage component through which a set of SATA signals from the storage component are communicated and the set of SATA signals are routed on the printed circuit board to the PCIe connectors.

5. The printed circuit board of claim **3** further comprising a compute component connected to the printed circuit board using a SATA connector and the set of SATA signals are communicated to the compute component.

6. The printed circuit board of claim **1** further comprising one or more digital enables that are routable through the PCIe connectors to allow external control of the one or more regulators.

7. The printed circuit board of claim **1** further comprising one or more of a power good signal and an acknowledge signal are routable through the PCIe connectors from the one or more regulators.

8. The printed circuit board of claim **6** further comprising a compute component connected to the printed circuit board and the compute component controls the digital enables.

9. The printed circuit board of claim **1** further comprising a temperature sensor attached to the printed circuit board and a temperature sensor interface is routed through the PCIe connector.

* * * * *