(12) STANDARD PATENT (11) Application No. **AU 2011289736 B2**
(19) AUSTRALIAN PATENT OFFICE

(54) Title
Methods and systems for extreme capacity management

(51) International Patent Classification(s)
G06F 9/06 (2006.01) ***G06F 9/44*** (2006.01)

(21) Application No: **2011289736** (22) Date of Filing: **2011.08.02**

(87) WIPO No: **WO12/021328**

(30) Priority Data

(31) Number	(32) Date	(33) Country
13/158,580	**2011.06.13**	**US**
13/152,341	**2011.06.03**	**US**
61/372,928	**2010.08.12**	**US**
13/158,571	**2011.06.13**	**US**
61/375,249	**2010.08.20**	**US**
13/152,349	**2011.06.13**	**US**

(43) Publication Date: **2012.02.16**
(44) Accepted Journal Date: **2016.06.30**

(71) Applicant(s)
Unisys Corporation

(72) Inventor(s)
Guarrieri, Stephen;Salsburg, Michael A.

(74) Agent / Attorney
Griffith Hack, GPO Box 4164, Sydney, NSW, 2001

(56) Related Art
US 2008/0027948
US 2005/0278439
US 2010/0088150
US 2010/0088205
US 7548843
US 2010/0198964
US 2010/0125845
CHRISTIAN, T. ET AL: "Automated Synthesis of Sustainable Data Centers", IEEE International Symposium on Sustainable Systems and Technology, 2009, ISSST '09, Piscataway, NJ, USA, 18 May 2009, pages 1-6
US 2010/0030877

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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
16 February 2012 (16.02.2012)

PCT

(10) International Publication Number
WO 2012/021328 A3

(51) International Patent Classification:
G06F 9/44 (2006.01) *G06F 9/06* (2006.01)

(21) International Application Number:
PCT/US2011/046200

(22) International Filing Date:
2 August 2011 (02.08.2011)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:

61/372,928	12 August 2010 (12.08.2010)	US
61/375,249	20 August 2010 (20.08.2010)	US
13/152,341	3 June 2011 (03.06.2011)	US
13/158,580	13 June 2011 (13.06.2011)	US
13/158,571	13 June 2011 (13.06.2011)	US
13/152,349	13 June 2011 (13.06.2011)	US

(71) Applicant *(for all designated States except US)*: **UNISYS CORPORATION** [US/US]; 801 Lakeview Dr., Suite 100, M/S 2NW, Blue Bell, PA 19422 (US).

(72) Inventors: **GUARRIERI, Stephen**; 115 Kings Road, Plymouth Meeting, PA 19462 (US). **SALSBURG, Michael, A.**; 133 Magnolia Drive, Phoenixville, PA 19460 (US).

(74) Agent: **GOEPEL, James**; Unisys Corporation, 801 Lakeview Dr., Suite 100, M/S 2 NW, Blue Bell, Pa 19422 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG,

[Continued on next page]

(54) Title: METHODS AND SYSTEMS FOR EXTREME CAPACITY MANAGEMENT

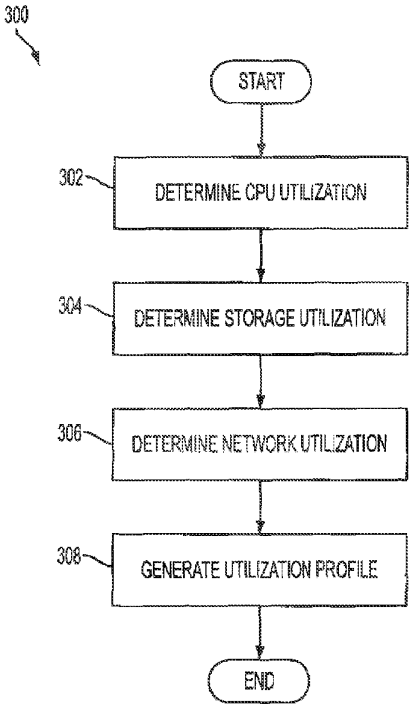

FIG. 3

(57) Abstract: Embodiments of the disclosed invention include an apparatus, method, and computer program product for. In one embodiment, a machine-readable tangible and non-transitory medium having instructions for managing resources is disclosed. The instructions when read by a machine, causes the machine to establish a workload profile for each tier within a plurality of tiers based on a computing request rate, a network request rate, and a storage request rate for each of the tiers. The machine also determines a configuration based on the workload profile for each of the tiers, wherein the configuration balances the computing request rate, the network request rate, and the storage request rate for each of the tiers.

WO 2012/021328 A3

ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

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))*

(88) Date of publication of the international search report:
12 April 2012

METHODS AND SYSTEMS FOR EXTREME CAPACITY MANAGEMENT

FIELD OF THE INVENTION

[0001] Embodiments of the present invention generally relate to a system and method for balancing, deploying, and managing a cloud computing environment. More specifically, embodiments of the invention provide aim to simplify, speed deployment, and optimize utilization of resources as well as drive interoperability of the three core datacenter components: servers, storage and network.

BACKGROUND

[0002] Three decades ago, capacity planning was handled by a team of experts who lovingly cared for a single mainframe. To justify the cost of a mainframe, every effort was made to wring out every CPU cycle. The complex and error-prone process included monitoring workloads, assessing business growth and requirements and correctly predicting when the mainframe should be upgraded. Upgrading too soon translated into excess costs due to the premium for cutting-edge technology, disturbance to the environment, downtime, and the expense of under-utilizing the costly server resources. Eventually, workloads disintegrated into multiple asynchronous workloads that could execute on multiple servers, including less expensive, commodity servers. The team was then faced with speeding up the capacity planning process so that servers could be monitored, analyzed, modeled and managed for capacity.

[0003] With the commoditization of server virtualization, another wave of disintegration has occurred. The number of virtual servers to be managed can be one or two orders of magnitude more than the physical servers that were being managed 5 years ago. In particular, with the advent of cloud computing, traditional management processes could no longer easily scale up where large numbers of servers are needed to "feed" a growing cloud within seconds. For instance, when planning to deploy a cloud computing environment, there are some unusual wrinkles in the standard approach for capacity planning. For example, the cloud allows users to provision their own resources (servers/storage/networks). Successful clouds keep up with demand in a way to present a façade of infinite elasticity. Cloud providers do not have control over what workloads will be using the cloud. Therefore, traditional approaches to capacity

2011289736 07 Jun 2016

planning based on careful measurements of workloads and their forecasted growth, cannot anticipate capacity in a timely manner in a cloud computing environment.

[0004] Accordingly, the disclosed embodiments provide a pragmatic approach to cloud computing aimed to simplify, speed deployment, and optimize utilization of resources in a cloud computing environment.

SUMMARY

[0005] The disclosed embodiments include a method, apparatus, and computer program product for managing resources such as servers of a cloud service provider. For instance, in one example, a computer implemented method for configuring a distributed computing system, the system comprising a plurality of servers organised into a plurality of tiers, comprises establishing, using a processor, a workload profile for each of the plurality of tiers based on a ratio between a computing request rate, a network request rate, and a storage request rate for each of the tiers; and determining, using the processor, a configuration of the system based on the workload profile expressed as a ratio between the rates for each of the tiers, such that the computing request rate, the network request rate, and the storage request rate for each respective tier are configured to reach maximum utilization at substantially the same time.

[0006] As another embodiment, a machine-readable tangible and non-transitory medium having instructions for managing resources in a distributed computing system, the system comprising a plurality of servers organized into a plurality of tiers, wherein the instructions, when read by the machine, causes a machine to perform the following: establish a workload profile for each of the plurality of tiers based on a ratio between a computing request rate, a network request rate, and a storage request rate for each of the tiers; and determine a configuration based on the workload profile expressed as a ratio between the rates for each of the tiers, such that the computing request rate, the network request rate, and the storage request rate for each respective tier are configured to reach maximum utilization at substantially the same time.

[0007] In still another embodiment, a system is disclosed. The distributed computing system comprises a plurality of servers organized into a plurality of tiers; a memory operable to store computer executable instructions; a processor configured to execute the computer executable instructions to establish a workload profile for each of the plurality of tiers based on a ratio between a computing request rate, a network request rate, and a storage request rate for each of the tiers; and determine a configuration based on the workload profile expressed as a ratio between the rates for each of the tiers, such that the computing request rate, the network request rate, and the storage

request rate for each respective tier are configured to reach maximum utilization at substantially the same time.

[0008] Additional advantages and novel features will be set forth in part in the description which follows, and in part will become apparent to those skilled in the art upon examination of the following and the accompanying drawings or may be learned by production or operation of the examples. The advantages of the present teachings may be realized and attained by practice or use of various aspects of the methodologies, instrumentalities and combinations set forth in the detailed examples discussed below.

BRIEF DESCRIPTION OF THE DRAWINGS

[0009] The accompanying drawings constitute a part of this specification and illustrate one or more embodiments of the disclosed system and methods, and further enhance the description thereof provided in this specification.

[0010] Figure 1 illustrates a network environment in which certain illustrative embodiments may be implemented;

[0011] Figure 2 illustrates a system in which certain illustrative embodiments may be implemented;

[0012] Figure 3 illustrates a process for generating a profile in accordance with an exemplary embodiment;

[0013] Figure 4 illustrates a profile generated for a web tier in accordance with an exemplary embodiment;

[0014] Figure 5 illustrates a profile generated for an application tier in accordance with an exemplary embodiment;

[0015] Figure 6 illustrates a profile generated for a data storage tier in accordance with an exemplary embodiment;

[0016] Figure 7 illustrates a Medium Platform Optimized Design (POD) for virtualized web tier configurations in accordance with an exemplary embodiment;

[0017] Figure 8 illustrates a Large Platform Optimized Design (POD) for virtualized application tier configurations in accordance with an exemplary embodiment;

[0018] Figure 9 illustrates an Enterprise Scalable (ESC) POD configuration in accordance with an exemplary embodiment; and

[0019] Figure 10 illustrates a small SAN and NFS high availability storage POD configuration in accordance with an exemplary embodiment.

DETAILED DESCRIPTION

[0020] The disclosed embodiments and advantages thereof are best understood by referring to Figures 1-10 of the drawings, like numerals being used for like and corresponding parts of the various drawings. Other features and advantages of the disclosed embodiments will be or will become apparent to one of ordinary skill in the art upon examination of the following figures and detailed description. It is intended that all such additional features and advantages be included within the scope of the disclosed embodiments. Further, the illustrated figures are only exemplary and are not intended to assert or imply any limitation with regard to the environment, architecture, design, or process in which different embodiments may be implemented.

[0021] Cloud computing, as referenced herein, refers to the provision of computational resources via a computer network. Cloud computing is an approach that enables organizations to leverage scalable, elastic and secure resources as services with the expected results of simplified operations, significant savings in cost and nearly instant provisioning. Some of the key tenets associated with cloud are elasticity and scalability, where resources can expand and contract as needed, and "Anything as a Service" (XaaS), where the details and concerns of implementation are abstracted for the customer.

[0022] Beginning with Figure 1, a cloud computing environment 100 in which certain illustrative embodiments may be implemented is depicted. The cloud computing environment 100 includes a plurality of client devices 102 that communicate over a network 110 with one or more systems of a cloud service provider 120. The network 110 may be any type of network including a wide area network, a local area network, a wireless network, one or more private networks, and the Internet. The client devices 102 may be any type of electronic device including, but not limited to, a laptop, personal computer, mobile phone, tablet, and personal digital assistant (PDA). The cloud service provider 120 provides computing resources, application services, and data storage to the plurality of client devices 102 using a plurality of servers. The plurality of servers may be located in one or more datacenters associated with the cloud service provider 120. The plurality of servers includes three main tiers/types of servers, a web tier 132, an application tier 134, and a data tier 136.

[0023] In support of the cloud computing principles, the disclosed embodiments recognize that the datacenter is undergoing a transformation driven by a "perfect storm" that is comprised of technology advances, extreme automation, and business shifts due to economic challenges. A major factor in this transformation is extreme automation and sense-and-respond systems that enable users to provision and migrate virtual machines (VMs) in minutes. Automation software centers on policy management of workload demand. Service monitoring focuses on optimizing the supply of resources for workloads. This optimization includes end-to-end transaction monitoring, environmental monitoring, resource correlation, performance and consumption monitoring.

[0024] Another contributor in this storm is a business shift that is driven by economic challenges and the need for agility. This business shift is movement toward Service Oriented Architecture (SOA), which is a methodology supported by data center transformation and embraced by cloud computing. SOA includes service governance, which leads to aligning current to future state. It leverages existing applications, provides for business value chain (BVC) alignment and enables future state planning. An SOA service infrastructure focuses on optimizing the supply of resources for workloads.

[0025] The disclosed embodiments recognize that transforming the datacenter requires new thinking regarding infrastructure economics as well as capacity planning and sizing. Current processes are relatively static and rigid, which makes planning and implementing them slow and ponderous. Workload patterns for the web-, application- and database tiers can be characterized by the ratio of compute, network and storage capacity and utilization. The disclosed embodiments recognize that providing simplified architecture that provides at least a minimal amount of performance and maintains the workload patterns while scaling for additional capacity would appeal to both datacenter architects and end users alike. Simplified architecture leads to simplified operations and significant savings as well as leading to greater elasticity and scalability.

[0026] Accordingly, the disclosed embodiments provide a methodology and a set of reference architectures, which integrate building blocks, referred to as Platform Optimized Designs (PODs). The methodology aims to simplify, speed deployment, and optimize utilization

of resources as well as drive interoperability of the three core data center components: servers, storage and network. A reference architecture, which includes the alignment and characterization of general data center workloads, supports a building block methodology that is both agile and scalable and necessary to meet the demands of the enterprise data center.

[0027] The following disclosure will describe the attributes of server and storage PODs that have been developed to create balanced systems among the three main tiers that form the pattern for today's applications— the web tier 132, the application tier 134, and the data tier 136. The notion of "balanced systems" arises where the system is properly balanced to handle the workload demands. In a perfectly balanced system, when the system reaches the maximum number of CPU arrival requests that the system can sustain, the system will also reach the maximum request rate for storage and networks. Additionally, a balanced system provides infrastructure capabilities to meet workload demands with adherence to the relative measures of compute, network and storage capacity

[0028] In contrast to a balanced system, if a workload predominately demands network resources, with little CPU or storage activity, then hosting that workload on a system that is configured for processor-intensive High Performance Computing (HPC) will waste CPU and memory resources and perhaps not keep up with the demands for network resources. If the system is not properly balanced, then, as more systems are added to address capacity, the costly underuse of CPU resources is compounded while the real bottleneck continues to plague the cloud provider.

[0029] The disclosed PODs provide building blocks that are independently scalable and therefore, deliver significantly greater 'efficiency' than alternate industry solutions. The PODS can be augmented with technologies to address specific customer requirements and Service Level Agreements (SLA) including availability and Quality of Service (QoS). In order to match the workload of an enterprise to these PODs, the disclosed embodiments explore and analyze compute-to-I/O ratios and workload characteristics that are common to each of these tiers to generate a profile for each tier.

[0030] The workload is the load that executes on the infrastructure based on business activity. For example, requirements for a SAP-based application development environment

might include business activity as defined by SAP transactions/sec and resource requirements as defined by CPU utilization, storage requests and network traffic. Quality requirements include service level requirements such as availability (for example, 99.99%) or quality of service (for example, response time). Meeting such requirements depends on the Reliability, Accessibility, and Scalability (RAS) characteristics of the underlying POD architecture and associated software.

[0031] The infrastructure of the reference configuration includes hardware plus operating system and virtualization software that are aligned with specific types of workloads. The infrastructure includes network support for the management subsystem but does not specify server or storage components required exclusively for management. For example, capabilities might include the compute capacity as determined by the number and type of VMs, transactions (of a specified workload) per second, I/O capacity in terms of I/O bandwidth, IOPS, storage and network bandwidth and latencies. Additional capabilities might include load balancing, fault tolerance or the functionality required by the customer (for example, server virtualization, support for .Net/Java and so on).

[0032] Figure 2 depicts a schematic diagram illustrating the basic components of an example architecture of a system 200 in which embodiments of the may be implemented. The system 200 includes a processor 200, main memory 202, secondary storage unit 204, and a communication interface module 208 for enabling the system 200 to communicate with the network 110. The processor 200 may be any type of processor capable of executing instructions for performing functions associated with the system 200 and the features associated with the claimed embodiments.

[0033] Main memory 202 is volatile memory that stores currently executing instructions/data, or instructions/data that are prefetched for execution.

[0034] The secondary storage unit 204 is non-volatile memory for storing persistent data (e.g., a hard drive). The secondary storage unit 204 stores the instructions associated with an operating system 212. The operating system 212 is software, consisting of programs and data, which manages the hardware resources of the system 200 and provides common services for execution of various applications 214.

[0035] In some embodiments, the system 200 may include an input/output interface module 206 that enables the system 200 to receive user input and to output information to a user or other devices. For example, the input/output interface module 206 may include a keyboard interface for receiving keyboard inputs from a user. The input/output interface module 206 may also enable external devices to be connected to the system 200.

[0036] In addition, in some embodiments, the system 200 may include a display module 210 such as a graphics card that enables information to be displayed on an internal or external display device.

[0037] Figure 3 depicts a flowchart describing a process 300 for generating a profile for each of the three main tiers (the web tier 132, the application tier 134, and the data tier 136). One of ordinary skill in the art will recognize that the process 300 may be written using any type of programming language and converted to machine readable instructions. These instructions may be stored in the secondary storage unit 204 and/or main memory 202 and executed by the processor 200 of the system 200. For example, in one embodiment, process 300 may be implemented directly on one or more systems in each of the three main tiers. In an alternative embodiment, process 300 may be implemented on a third party system that is configured to monitor the performance of one or more systems in each of the three main tiers.

[0038] At step 302, the process 300 determines CPU utilization for a particular machine. In one embodiment, the process retrieves CPU percent utilization from the statistics that are gathered by the operating system of the particular machine. To illustrate this, if CPU percent utilization is measured to be 75%, this means that, for each elapsed second, the CPU was busy for .75 of the second. The standard notation for utilization is ρ , which is a number between 0 and 1. Therefore, CPU activity per second is denoted as ρ_{CPU} ,where ρ_{CPU} = (%Processor Time)/100.

[0039] At steps 304 and 306, the process 300 similarly determines storage and network utilization. Again, in one embodiment, the process 300 the process determines storage and network utilization from the statistics that are gathered by the operating system of the particular machine. From the operating system's point of view, storage I/O utilization can be directly

measured for each physical disk. Utilizing the %Idle Time gathered from the operating system, and following equation may be utilized to determine storage I/O utilization.

[0040] $\sum_{i=1}^{n} \rho_{N_i}$ = Sum of (100-%Idle Time)/100 for all instances, for i = 1…n disks

[0041] It should be noted that the term physical disk includes storage subsystems that may be shared by multiple servers. This is possible through virtual storage subsystems or Storage Area Networks (SANs). For network devices, the operating system does not measure any delay because, from its vantage point, packets going in and out of the server will be transferred at the current bandwidth of the network interface device. So, although the operating system is reporting what it sees as classical disk utilization, the values may be skewed due to queuing by other servers in the back end subsystem.

[0042] For network activity, the process can measure the individual components of the equation using the Network Interface group of counters. For each network interface, measure Bytes Total/s and use the maximum bandwidth, as indicated by the network interface vendor.

[0043] $\sum_{j=1}^{m} \rho_{N_j}$ = Sum of (Bytes Total/sec / (Network Interface Bandwidth (bytes / sec)) for all instances for j = 1..m network interfaces

[0044] Although the above example utilizes statistics gathered from the operating system, in an alternative embodiment, third party software may be used gather the statistics necessary to determine the above performance parameters.

[0045] Based on the gathered statistics, the relationship between the CPU utilization, data storage utilization, and network utilization can be express as the following tuple:

[0046] $(\rho_{CPU}, \frac{\sum_{i=1}^{n} \rho_{N_i}}{n}, \frac{\sum_{j=1}^{m} \rho_{N_j}}{m})$

[0047] Where:

[0048] ρ_{CPU} is the utilization of CPU on a server;

[0049] $\frac{\sum_{i=1}^{n} \rho_i}{n}$ is the average utilization of all storage devices attached to the server; and

[0050] $\frac{\sum_{j=1}^{m} \rho_j}{m}$ is the average utilization of all network devices attached to the server.

[0051] Based on the gathered statistics, at step 308, the process generates a profile regarding the workload activity with the particular machine, with process 300 terminating thereafter.

[0052] Based on the profiles generated for each of the machines, a specific tier profile regarding the workload activity for each of the three main system tiers (web tier 132, application tier 134, and data tier 136) may be constructed. Although it is acknowledged that exceptions exist, at a base level, each of these tiers require a different balance of resources at the system level.

[0053] For example, Figure 4 illustrates an exemplary profile generated for the web tier systems. As can be seen, the web tier systems have low CPU and memory utilization; low disk capacity with low storage I/Os per second (IOPS) requirements. This environment requires that moderate-to-high network bandwidth be specified in both packets per second and MB per second. In this type of environment, the memory usage might scale linearly with the CPU utilization.

[0054] Figure 5 illustrates an exemplary profile generated for the application tier systems. The application tier profile shows moderate CPU and memory utilization; moderate disk capacity with moderate-to high storage and network IOPS requirements (depending on application and use case). This environment requires high network bandwidth both in packets per second and MB per second. The application tier has far less network activity than the web tier and more CPU activity, with more storage activity.

[0055] Figure 6 illustrates an exemplary profile generated for the database tier systems. The database tier has high CPU and memory utilization and requirements. This environment requires high disk capacity with high IOPS along with moderate-to high network bandwidth both in packets per second and MB per second. Requirements in this environment increase with the transaction workload; performance also depends on application and use case. Disk IOPS depend on read/write ratios and the layout of the database.

[0056] In each of the graphs of Figures 4-6, it is visually obvious that the three utilizations will not reach 100% utilization at the same time. Therefore, the current resources for the systems in each of these tiers are unbalanced because one of the resources will be exhausted before the others.

[0057] Of course, the above profiles for each of the tiers may change based on technological advances. For example, what if the CPU technologies differ so that the target servers can execute 25% more CPU cycles? Or what if network interfaces with 10 times the bandwidth are used? For storage utilization, what if spinning disks are replaced by solid state disk?

[0058] Therefore, in an alternative embodiment, the above approach may be slightly modified so that it is hardware independent. In other words, determining the workload demand irrespective of the target hardware. In order to do so, we modify the above statistics to requests per second and not the time per request. For instance, we defined:

[0059] λ_* = Arrival rate for resource *

[0060] $S[s]_*$ = Average service time per request for resource *

[0061] Then, the utilization of a resource such as CPU can be expressed as:

[0062] $\rho_{cpu} = \lambda_{cpu} S[s]_{cpu}$

[0063] We can then re-write the above tuple as:

[0064] $(\lambda_{cpu} S[s]_{cpu}, \frac{\sum_{i=1}^{n} \lambda_{D_i} S[s]_{D_i}}{n}, \frac{\sum_{j=1}^{m} \lambda_{N_j} S[s]_{N_j}}{m})$

[0065] Within the Physical Disk group, we can determine the arrival rate for each logical disk for i = 1 … n disks as:

[0066] λ_{D_i} = Disk Transfers/Sec

[0067] $S[s]_{D_i} = \frac{(100 - \%Idle\ Time)}{\lambda_{D_i}}$

[0068] Within the Network Interface group, we can determine the arrival rate for each network interface for j = 1…m network interfaces as:

[0069] λ_{Net_j} = Packets/Sec

[0070] $E[s]_{Net_j}$ =(Bytes Total/sec / (Network Interface Bandwidth (bytes / sec)) / λ_{Net_j}

[0071] In accordance with the disclosed embodiments, a system is considered balanced when the maximum number of CPU arrival requests that the system can sustain and the maximum request rate for storage and networks are reached at approximately the same time. Therefore, the relationship between λ_{CPU}, $\sum_{i=1}^{n} \lambda_{CPU_i}$, and $\sum_{j=1}^{m} \lambda_{Net_j}$ must be determined in order to balance the system for each of the three profiles.

[0072] Using the above process, the disclosed embodiments provide a rough/"good enough" infrastructure that aims to balance infrastructure capabilities and cost with workload requirements using standard building blocks (PODs) plus additional components to form Reference Architectures (RAs) and matches the customer workload requirements to the Infrastructure capabilities.

[0073] As referenced herein, a Solution Architecture (SA) is a collection of technical capabilities that provides business value. An SA serves as an architectural construct that identifies the technologies needed to support a specific project and identifies similar projects that have already been deployed in the environment. Additionally, an SA provides a baseline for immediately creating and deploying an infrastructure solution that meets a specific business need.

[0074] The disclosed embodiments provide a 'simpler' approach to capacity planning that helps clarify the proposed direction for the infrastructure and accelerates deployment as compared to the traditional approach that is based on careful measurements of workloads and their forecasted growth. In addition to accelerating deployment, the disclosed approach is easier to modify in response to a change in the workload, whereas the traditional approach cannot anticipate capacity changes in a timely manner. Therefore, the disclosed embodiments reduce cost and complexity associated with managing the datacenter.

[0075] In addition, the disclosed embodiments are easily scalable to newer technology. For instance, during a datacenter transformation, older infrastructure may be replaced with newer components. This can ultimately save on capital and operational expenses, as well as reducing power and cooling costs. For example, the disclosed embodiments may be easily scaled by establishing a change factor based on an estimation of a new CPU utilization against the current utilization. The change factor may be based on an industry standard benchmark, such as, but not limited to, benchmarks provided by SPECint. SPECint is a computer benchmark specification to determine a CPU's integer processing power and it is maintained by the Standard Performance Evaluation Corporation (SPEC). As an example, representing the SpecInt results for Server X and Server Y as α_X and α_Y and the average service time per request for Server X is $E[s]_{CPU_X}$ then we can estimate the CPU time on the target (Server Y) as:

[0076] $$E[s]_{CPU_Y} = \frac{\alpha_X}{\alpha_Y} E[s]_{CPU_X}$$

[0077] For example, assume Server X benchmark indicates a base value of 100 and Server Y's benchmark indicates 125. Since the SpecInt value increases as an inverse to the CPU service times, a server that is 25% faster according to the benchmark will yield:

[0078] $$\frac{\alpha_X}{\alpha_Y} = .80$$

[0079] Conversely, the number of CPU requests that can be sustained on a fully utilized system is 25% higher than Server X. Therefore, in order to maintain the same balance, the system of the disclosed embodiments need to also scale the system resources to support 25% more requests for network and storage resources.

[0080] As shown above, the disclosed embodiments provide an approach to deal with the uncertainties of the cloud-based workloads by suggesting a small number of profiles based on the current pattern of web, application and database tiers. The methodology supports an agile deployment model that is used early in the service delivery model and does not preclude the use of deeper forensics and analysis. The disclosed embodiments enable quick deployment and expansion of resources as is necessary in a cloud computing environment to provide the façade of infinite elasticity. As the cloud expands, additional infrastructure can be quickly deployed so

that it is balanced to handle the profiles without over-configuring in the areas of CPU, storage and networking resources.

[0081] To further expedite deployment, the disclosed embodiments recommend that the Reference Architecture (RA) be the lowest granularity of a deliverable product. The RA contains any combination of components, frameworks and services including third party products that are necessary to address specific customer requirements. RAs might contain a single component or POD; however, they usually contain more than one. Although the RAs may include optional components and reference configurations (such as, special I/O adapters, switch/firewall, and so on that might require services to validate), the goal is to encourage the solution-centric model. In this embodiment, only RAs are released and supported; and components are not optional within a solution. The components defined for the PODs or components in the aggregation layer relative to a solution might be replaced, added, or removed over time.

[0082] The disclosed embodiments define server (compute), network and storage PODs independently in order to balance the infrastructure capabilities and the workload requirements associated with the primary data-center tiers: the web tier 132, the application tier 134, and the data tier 136.

[0083] In one embodiment, the tier models are derived under the assumption that these environments could be and should be 100 percent virtualized in a secure, multitenant configuration. As such, priorities include VM density, I/O flexibility and an efficient yet resilient power and cooling solution. The reference configurations for the tier models target a single rack mounted cabinet design. That is, the compute, network and storage capacity for the base infrastructure is achievable within a standard rack.

[0084] For example, with reference now to Figure 7, an embodiment of a medium POD for virtualized web tier configurations 700 is presented, which targets application workloads that can leverage high-density, scale-out architecture. The medium POD configuration 700 supports VMs that require relatively lower CPU/memory resource utilization as well as resilience for the web tier. The configuration for the medium POD configuration includes a blade chassis 710,

which supports a maximum capacity of 14 blades with a total of 168 cores and 672 GB of memory (supporting 336 VMs, each with 2 GB of memory per VM and 2 VMs per core).

[0085] In the disclosed embodiment, two different types of storage options are available—FC SAN or iSCSI storage arrays. The storage capacity is sized accordingly with the maximum VMs. For instance, 336 VMs * 30 GB of storage per VM yields 10.08 TB of usable capacity. The configured raw capacity addresses operating system formatting and RAID considerations.

[0086] The medium POD configuration 700 optimizes network performance and includes multiple virtualization solutions. In one embodiment, the medium POD configuration 700 includes a VMware ESX hypervisor—vSphere 4 Enterprise Plus, vCenter for VM management (hosted on SPC management server) and vShield for secure zoning. In a VMware high availability (HA) environment, the usable configuration is 156 cores and 624 GB of memory (supporting 312 VMs each with 2 GB of memory per VM and 2 VMs per core).

[0087] Figure 8 illustrates an embodiment of a Large POD for virtualized high performance configurations 800, which target application workloads that can leverage high density scale-out architecture. The large POD configuration 800 hosts large memory/storage VMs and is a good general purpose system that is best suited for the application tier 134. The large POD configuration 800 includes a blade chassis that supports a maximum capacity of 168 cores and 1344 GB of memory (supporting 336 VMs, each with 4 GB of memory per VM and 2 VMs per core). In certain embodiments, the large POD configuration 800 supports different storage options that provide a maximum of 50 TB of raw storage.

[0088] In the VMware HA environment, the usable configuration consists of 156 cores and 1248 GB of memory (supporting 312 VMs, each with 4 GB of memory per VM and 2 VMs per core). This configuration optimizes processing, network and storage performance utilizing 10-Gb Ethernet connections and 8-Gb storage connections. In one embodiment, the large POD configuration 800 includes a connection to storage area network (SAN) storage having 42 TB of usable storage. Alternatively, the large POD configuration 800 could include a connection to Network-attached storage (NAS) storage.

[0089] In one embodiment, the software configuration of the large POD 800 configuration includes VMware ESX ™, vSphere™, vCenter™ and vShield™. vSphere™ is a cloud operating system that is able to manage large pools of virtualized computing infrastructure. The large POD configuration 800 may be paired with multiple storage options including RAID, HA configurations with redundant SAN switches with load balancing, and finally, fully automated DRS and vSphere clusters.

[0090] Figure 9 illustrates an embodiment of an Enterprise Scalable (ESC) high availability POD configuration 900, which targets enterprise-class, scale-up workloads with high-availability requirements (e.g., scale up beyond 12 cores). The ESC POD configuration 900 is best fitted for the back office/database tier 136. For example, the ESC POD configuration 900 is well positioned with additional memory and I/O capacity to address a wider range of workloads and configurations from multiple VMs per core to multiple cores per VM to physical servers. However, in a virtualized environment, additional ESC PODs should be considered in the detailed implementation to address HA requirements.

[0091] In one embodiment, the ESC POD configuration 900 includes an eight-socket scalable server with a maximum capacity of 64 cores, 1 TB of memory and 40 TB of usable storage. For HA capability, additional components may be added including redundant SAN switches with load balancing, redundant networking (NIC teaming, redundant switch fabric) and premier operating system software and hypervisor. Clustering may be selected between the two 4-socket cells that make up the eight-socket server. Although the ESC POD configuration 900 does not explicitly address the management subsystem, a software management stack is required.

[0092] With reference now to embodiments of storage PODs, Figure 10 illustrates examples of storage POD configurations in accordance with the disclosed embodiments. In particular, Figure 10 illustrates the configuration of a SAN Storage POD 1000 and a Network File System (NFS) Storage POD 1010.

[0093] The disclosed storage PODs are designed to have enterprise-class performance, availability and protection features. The current building block for these PODs is the EMC VNX Model 5300 Unified Storage System, which is optimized for virtualized environments. Along

with VNX, the Storage POD includes RecoverPoint software for business usage/disaster recovery (BU/DR) and Powerpath software for load balancing and failover.

[0094] The disclosed storage subsystem supports Enterprise Flash drives (EFDs), 15-rpm SAS disk drives and near-line 7200-rpm 1TB or 2TB SAS disks. These different types of disks are used by the FAST-VP tiered storage software for directing traffic to the right tiers for optimal performance.

[0095] The Small storage POD provides the storage capacity required by the Small Cloud RA blade or server PODs - 70 GB per VM of tiered storage for up to 168 VMs. The Medium storage POD provides the storage capacity required by the Medium Cloud RA blade or server PODs - 70 GB per VM of tiered storage for up to 336 VMs. The Large storage POD provides the storage capacity required by the Large Cloud RA blade POD - 125 GB per VM of tiered storage for up to 336 VMs. The Enterprise Scale-Up (ESC) Storage POD provides the storage capacity required by the ESC RA - 250 GB per VM of tiered storage for up to 128 VMs. Each of the storage PODs are independently configurable, with a range of capacities available. The capacities listed are the minimum suggested capacities, more drives can be added as required.

[0096] The default RAID configuration chosen for the storage PODs is RAID 5 arrays with 1 parity drive and a hot spare for each drive type (i.e. an EFD hot spare, a 15k SAS hot spare and a 7200rpm SAS hot spare).

[0097] The disclosed storage POD configurations provide a high level of fault tolerance and storage efficiency, which balances cost versus performance. Detailed sizing efforts and/or customer requirements influence the usable capacity (that is, other desired RAID levels) and must be considered in the final storage design implementation.

[0098] The server and storage reference configurations offer two different host connect options, FC (SAN) or NAS. The server RAs described in this document are the SAN option, with 8Gb FC HBAs in the host system and the corresponding 8Gb FC I/O modules in the VNX 5300 storage subsystem. For server and storage configurations where NAS storage is required, the VNX 5300 I/O modules will be configured for 1Gb or 10Gb modules depending on the desired speed and host I/O card option.

[0099] The disclosed storage POD configurations provide a high level of fault tolerance and storage efficiency, which balances cost versus performance. The capabilities of each configuration are stated in terms of the number of VMs supported and of the type of VMs (not all VMs are created equal). For example, a web tier VM might not be CPU or memory intensive (which allows for less memory/VM and more VMs/core) compared to a database VM. A database VM might require up to 4 GB of RAM per VM and have a limit of 2 VMs per CPU core, but a web tier VM can perform within an SLA with 2 GB of RAM per VM. The number and type of VMs that are optimal for a customer's environment are best determined with the workload analysis/migration and capacity planning tools. The analysis determines what type of server POD should be used for the VM profile, and what edge-connected components, as defined by customer requirements, are necessary to complete the infrastructure (for example, database accelerators, network and/or SAN load balancers, firewalls, and so on).

[00100] The I/O capacity of each server and storage POD is flexible enough to allow for a range of applications. However, the medium server POD is optimized for a web-tier environment with moderate-to-heavy network activity, where the I/O ratio is estimated to be around 65 percent network and 35percent storage traffic. The large server and storage PODs are optimized for database usage with moderate-to-heavy storage activity, where the I/O ratio is estimated to be around 65 percent storage and 35 percent network traffic. The large number of network ports in each configuration (56 total) allows for virtualization hypervisors such as VMware with vMotion to use two ports for operating system and VM migration—leaving two ports per blade available for the customer network.

[00101] Accordingly, the disclosed embodiments provides an approach to deal with the uncertainties of the cloud-based workloads by suggesting a small number of profiles based on the current pattern of web, application and database tiers. The methodology supports an agile deployment model that is used early in the service delivery model and does not preclude the use of deeper forensics and analysis. Using reference configurations, as described above, to design private or hybrid cloud configurations shortens both the development/test and sales cycles plus ensures that the major building blocks necessary to build an enterprise-class cloud configuration have been considered. As the cloud expands, additional infrastructure can be quickly deployed

so that it is balanced to handle the profiles without over-configuring in the areas of CPU, storage and networking resources.

[00102] The above embodiments are merely provided as examples and are not intended to limit the invention to any particular configuration.

[00103] Certain illustrative embodiments described herein can take the form of an entirely hardware embodiment, an entirely software embodiment or an embodiment containing both hardware and software elements. Furthermore, certain illustrative embodiments can take the form of a computer program product accessible from a computer-usable or computer-readable medium providing program code for use by or in connection with a computer or any instruction execution system. For the purposes of this description, a computer-usable or computer-readable medium can be any tangible apparatus that can contain, store, communicate, or transport the program for use by or in connection with the instruction execution system, apparatus, or device. The previous detailed description discloses several embodiments for implementing the invention and is not intended to be limiting in scope. Those of ordinary skill in the art will recognize obvious variations to the embodiments disclosed above and the scope of such variations are intended to be covered by this disclosure. The following claims set forth the scope of the invention.

CLAIMS

What is claimed is:

1. A computer implemented method for configuring a distributed computing system, the system comprising a plurality of servers organized into a plurality of tiers, the method comprising:

establishing, using a processor, a workload profile for each of the plurality of tiers based on a ratio between a computing request rate, a network request rate, and a storage request rate for each of the tiers; and

determining, using the processor, a configuration of the system based on the workload profile expressed as a ratio between the rates for each of the tiers, such that the computing request rate, the network request rate, and the storage request rate for each respective tier are configured to reach maximum utilization at substantially the same time.

2. The method of Claim 1, wherein the plurality of tiers comprises a web tier, an application tier, and a data tier.

3. The method of Claim 1, wherein the system comprises servers in a datacenter of a cloud service provider.

4. The method of Claim 1, wherein establishing the workload profile includes utilizing statistics gathered by an operating system.

5. The method of Claim 1, wherein determining the configuration includes selecting from a plurality of preconfigured platform optimized design components (PODs).

6. The method of Claim 5, wherein the plurality of preconfigured PODs includes a preconfigured server POD, a preconfigured network POD, and a preconfigured storage POD.

7. The method of Claim 5, wherein each of the PODs are uniquely preconfigured to match a workload profile for a particular tier.

8. The method of Claim 7, wherein the PODs are uniquely preconfigured with both hardware and software components.

9. The method of Claim 1, further comprising:

establishing a change factor based on an estimation of a new utilization rate against a current utilization rate for upgrading the system, the estimation based on a predetermined benchmark.

10. The method of Claim 1, wherein the established workload profile is hardware independent.

11. A machine-readable tangible and non-transitory medium having instructions for managing resources in a distributed computing system, the system comprising a plurality of servers organized into a plurality of tiers, wherein the instructions, when read by the machine, causes a machine to perform the following:

establish a workload profile for each of the plurality of tiers based on a ratio between a computing request rate, a network request rate, and a storage request rate for each of the tiers; and

determine a configuration based on the workload profile expressed as a ratio between the rates for each of the tiers, such that the computing request rate, the network request rate, and the storage request rate for each respective tier are configured to reach maximum utilization at substantially the same time.

12. The medium of claim 11, wherein the plurality of tiers consists of a web tier, an application tier, and a data tier.

13. The medium of claim 11, wherein the resources are servers in a datacenter of a cloud service provider.

14. The medium of claim 11, wherein establishing the workload profile includes utilizing statistics gathered by an operating system.

15. The medium of claim 11, wherein determining the configuration includes selecting from a plurality of preconfigured platform optimized design components (PODs).

16. The medium of Claim 15, wherein the plurality of preconfigured PODs includes a preconfigured server POD, a preconfigured network POD, and a preconfigured storage POD.

17. The medium of Claim 15, wherein each of the PODs are uniquely preconfigured to match a workload profile for a particular tier.

18. The medium of Claim 17, wherein the PODs are uniquely preconfigured with both hardware and software components.

19. The medium of claim 11, further comprising:

establishing a change factor based on an estimation of a new utilization rate against a current utilization rate for upgrading the resources, the estimation based on a predetermined benchmark.

20. A distributed computing system comprising:

a plurality of servers organized into a plurality of tiers;

a memory operable to store computer executable instructions;

a processor configured to execute the computer executable instructions to:

establish a workload profile for each of the plurality of tiers based on a ratio between a computing request rate, a network request rate, and a storage request rate for each of the tiers; and

determine a configuration based on the workload profile expressed as a ratio between the rates for each of the tiers, such that the computing request rate, the network request rate, and the storage request rate for each respective tier are configured to reach maximum utilization at substantially the same time.

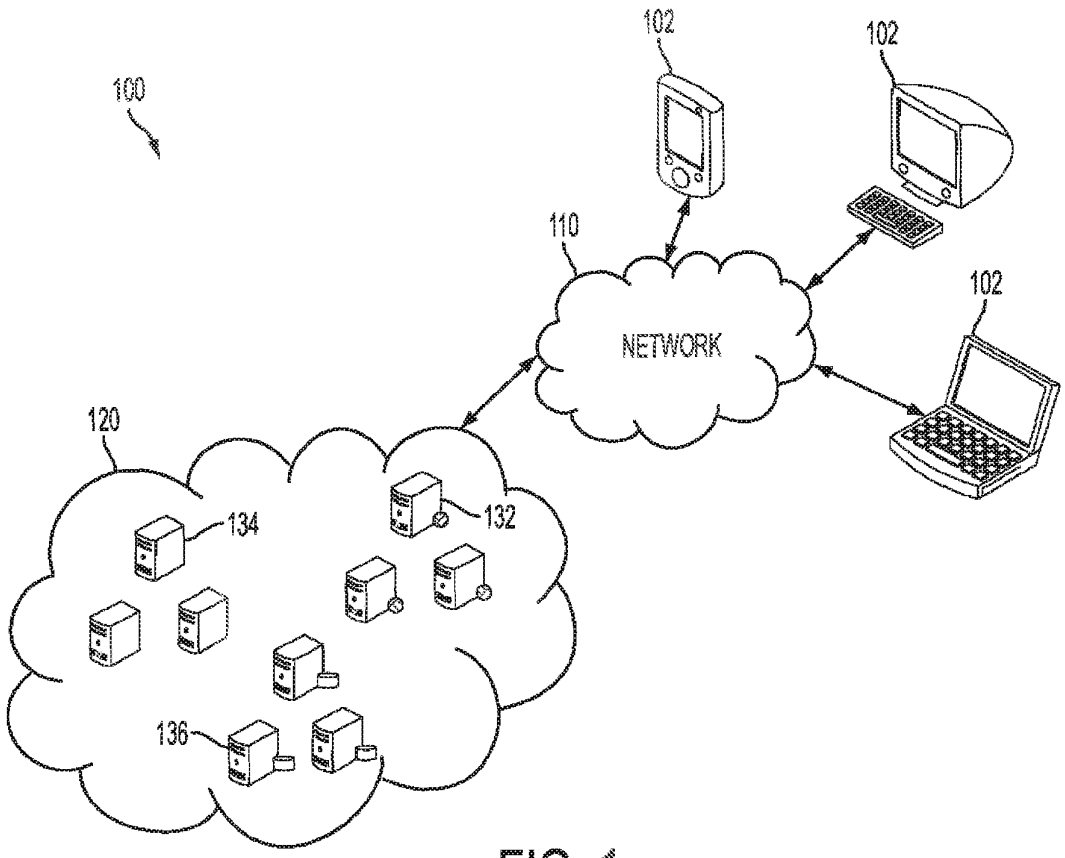
100
102
102
102
110
NETWORK
120
134
132
136

FIG. 1

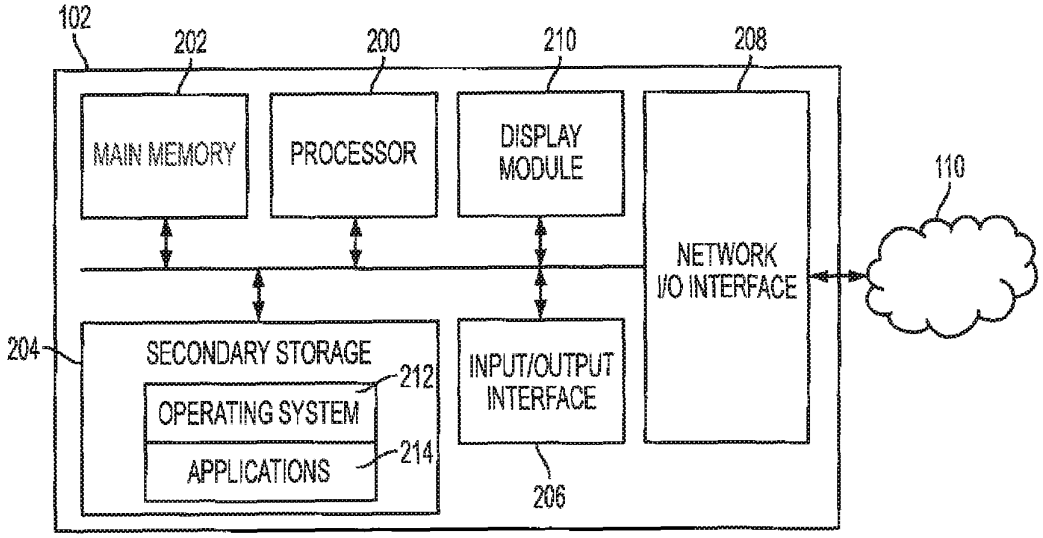
102
202
200
210
208
MAIN MEMORY
PROCESSOR
DISPLAY MODULE
NETWORK I/O INTERFACE
110
204
SECONDARY STORAGE
212
OPERATING SYSTEM
214
APPLICATIONS
INPUT/OUTPUT INTERFACE
206

FIG. 2

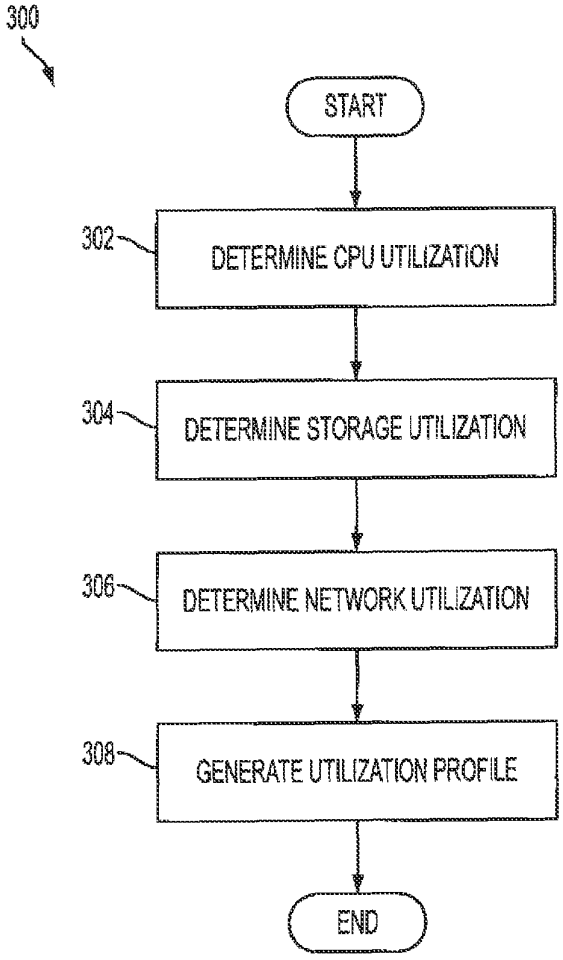
300
START
302
DETERMINE CPU UTILIZATION
304
DETERMINE STORAGE UTILIZATION
306
DETERMINE NETWORK UTILIZATION
308
GENERATE UTILIZATION PROFILE
END

FIG. 3

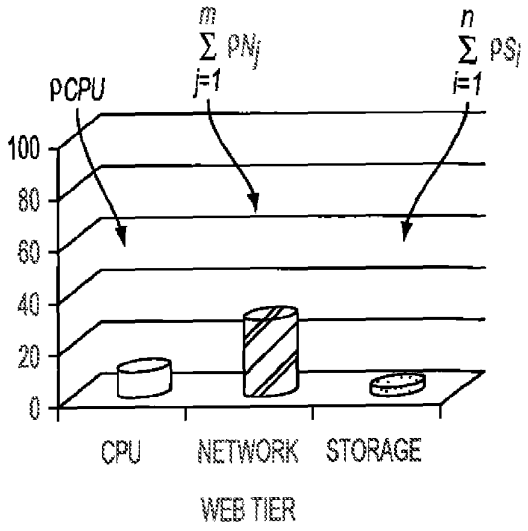

FIG. 4

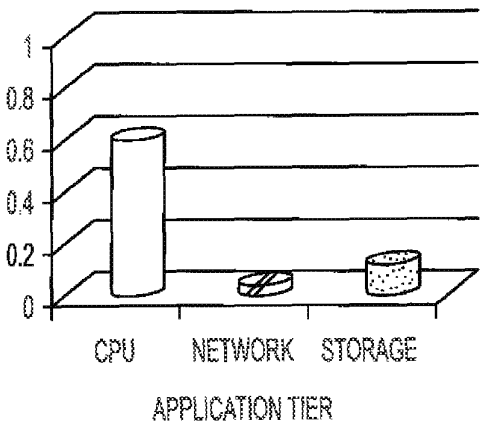

FIG. 5

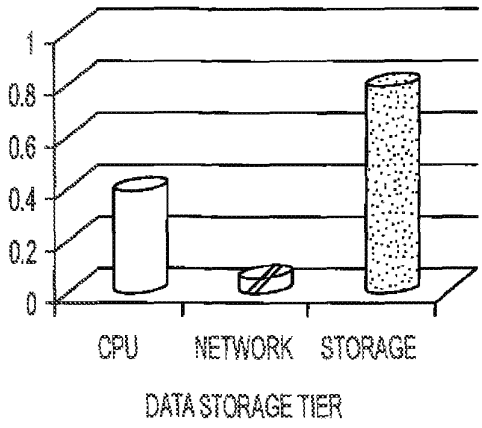

FIG. 6

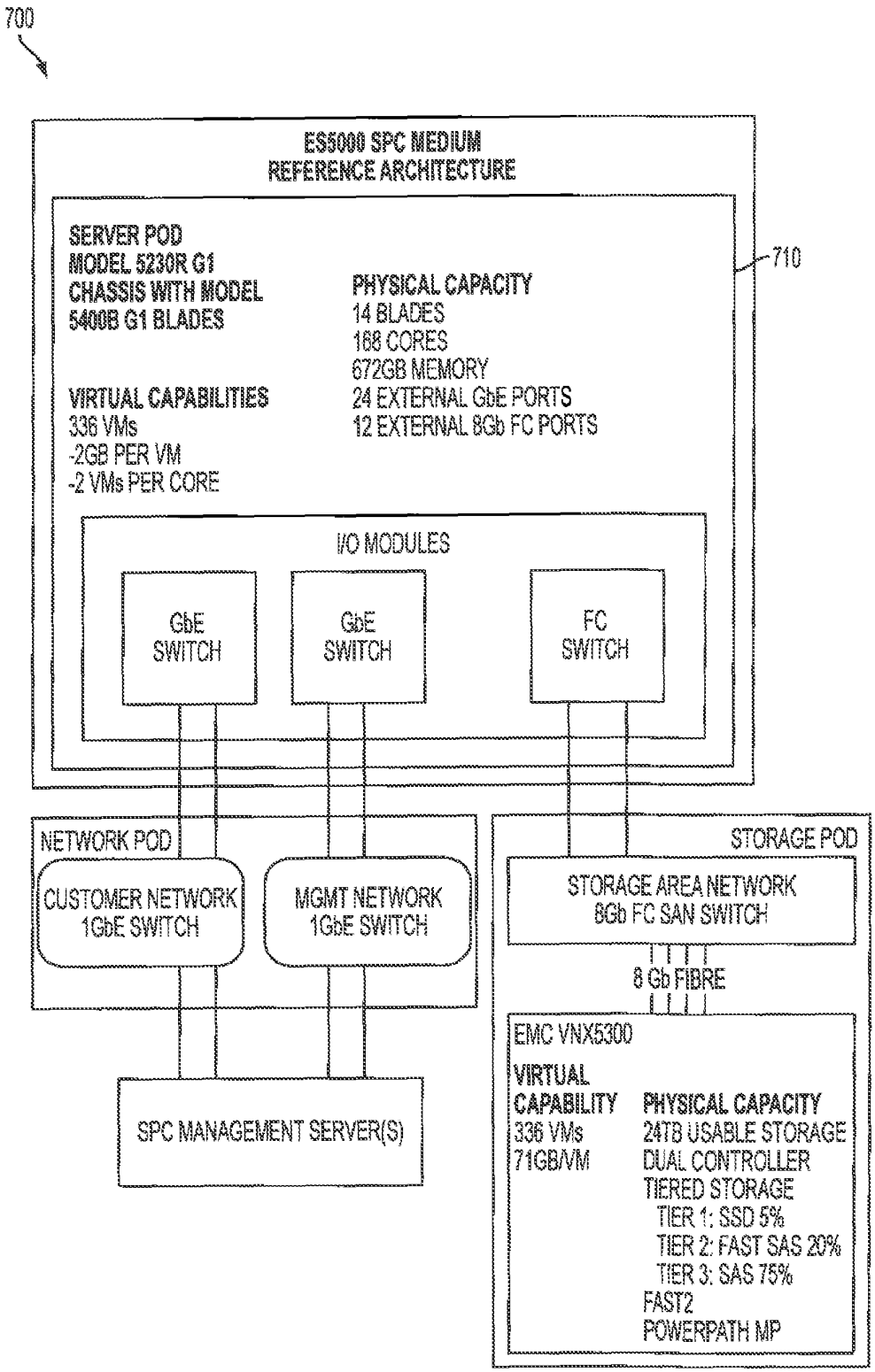
700
ES5000 SPC MEDIUM
REFERENCE ARCHITECTURE
710
SERVER POD
MODEL 5230R G1
CHASSIS WITH MODEL
5400B G1 BLADES
PHYSICAL CAPACITY
14 BLADES
168 CORES
672GB MEMORY
24 EXTERNAL GbE PORTS
12 EXTERNAL 8Gb FC PORTS
VIRTUAL CAPABILITIES
336 VMs
-2GB PER VM
-2 VMs PER CORE
I/O MODULES
GbE SWITCH
GbE SWITCH
FC SWITCH
NETWORK POD
CUSTOMER NETWORK 1GbE SWITCH
MGMT NETWORK 1GbE SWITCH
SPC MANAGEMENT SERVER(S)
STORAGE POD
STORAGE AREA NETWORK 8Gb FC SAN SWITCH
8 Gb FIBRE
EMC VNX5300
VIRTUAL CAPABILITY
336 VMs
71GB/VM
PHYSICAL CAPACITY
24TB USABLE STORAGE
DUAL CONTROLLER
TIERED STORAGE
TIER 1: SSD 5%
TIER 2: FAST SAS 20%
TIER 3: SAS 75%
FAST2
POWERPATH MP

FIG. 7

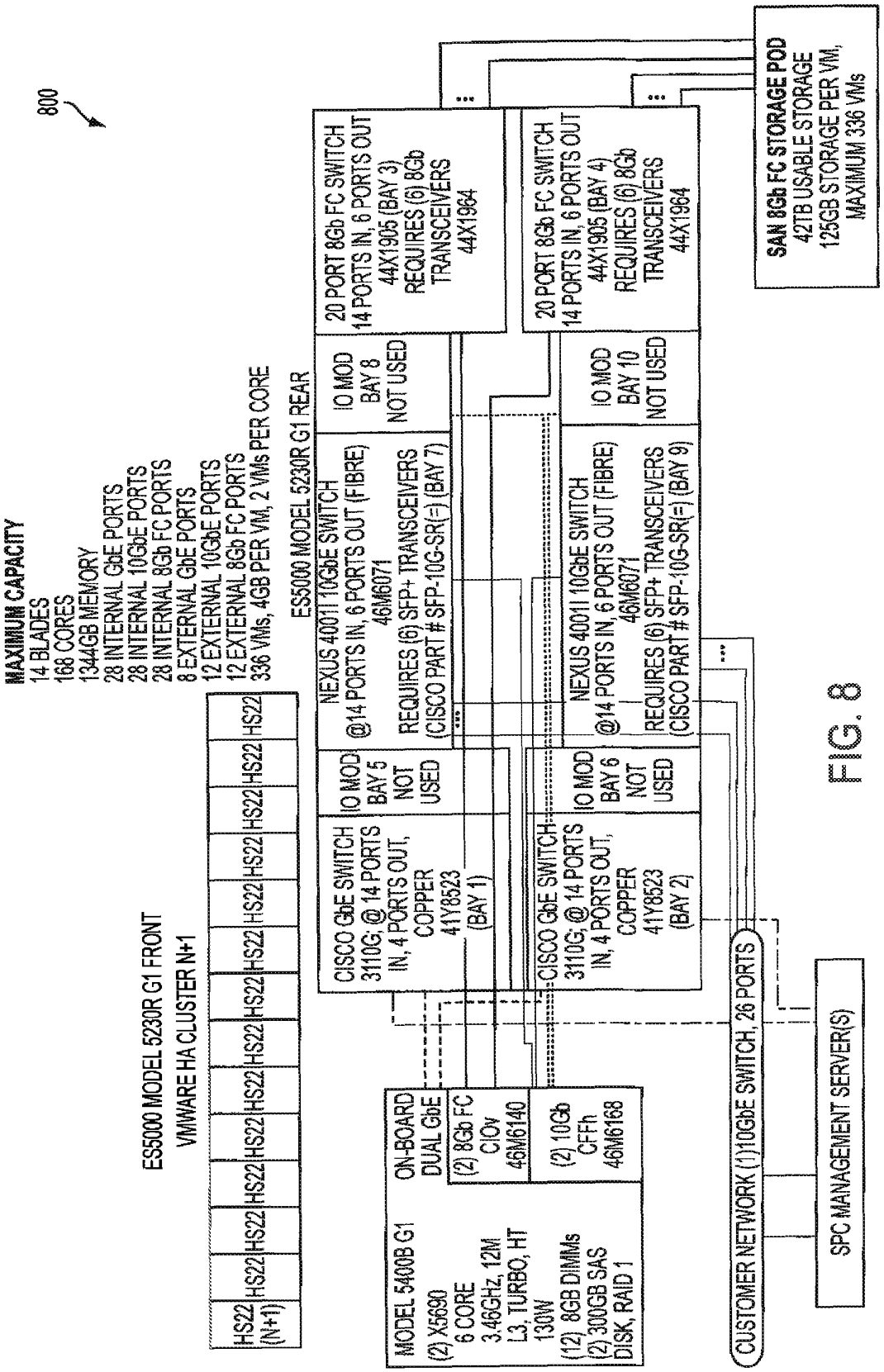
800
MAXIMUM CAPACITY
14 BLADES
168 CORES
1344GB MEMORY
28 INTERNAL GbE PORTS
28 INTERNAL 10GbE PORTS
28 INTERNAL 8Gb FC PORTS
8 EXTERNAL GbE PORTS
12 EXTERNAL 10GbE PORTS
12 EXTERNAL 8Gb FC PORTS
336 VMs, 4GB PER VM, 2 VMs PER CORE
ES5000 MODEL 5230R G1 FRONT
VMWARE HA CLUSTER N+1
HS22 (N+1)
HS22
ES5000 MODEL 5230R G1 REAR
CISCO GbE SWITCH 3110G; @ 14 PORTS IN, 4 PORTS OUT, COPPER 41Y8523 (BAY 1)
IO MOD BAY 5 NOT USED
NEXUS 4001I 10GbE SWITCH @14 PORTS IN, 6 PORTS OUT (FIBRE) 46M6071
REQUIRES (6) SFP+ TRANSCEIVERS (CISCO PART # SFP-10G-SR(=) (BAY 7)
IO MOD BAY 8 NOT USED
20 PORT 8Gb FC SWITCH 14 PORTS IN, 6 PORTS OUT 44X1905 (BAY 3) REQUIRES (6) 8Gb TRANSCEIVERS 44X1964
CISCO GbE SWITCH 3110G; @ 14 PORTS IN, 4 PORTS OUT, COPPER 41Y8523 (BAY 2)
IO MOD BAY 6 NOT USED
NEXUS 4001I 10GbE SWITCH @14 PORTS IN, 6 PORTS OUT (FIBRE) 46M6071
REQUIRES (6) SFP+ TRANSCEIVERS (CISCO PART # SFP-10G-SR(=) (BAY 9)
IO MOD BAY 10 NOT USED
20 PORT 8Gb FC SWITCH 14 PORTS IN, 6 PORTS OUT 44X1905 (BAY 4) REQUIRES (6) 8Gb TRANSCEIVERS 44X1964
SAN 8Gb FC STORAGE POD
42TB USABLE STORAGE
125GB STORAGE PER VM, MAXIMUM 336 VMs
MODEL 5400B G1
(2) X5690 6 CORE 3.46GHz, 12M L3, TURBO, HT 130W
(12) 8GB DIMMs
(2) 300GB SAS DISK, RAID 1
ON-BOARD DUAL GbE
(2) 8Gb FC CIOv 46M6140
(2) 10Gb CFFh 46M6168
CUSTOMER NETWORK (1)10GbE SWITCH, 26 PORTS
SPC MANAGEMENT SERVER(S)

FIG. 8

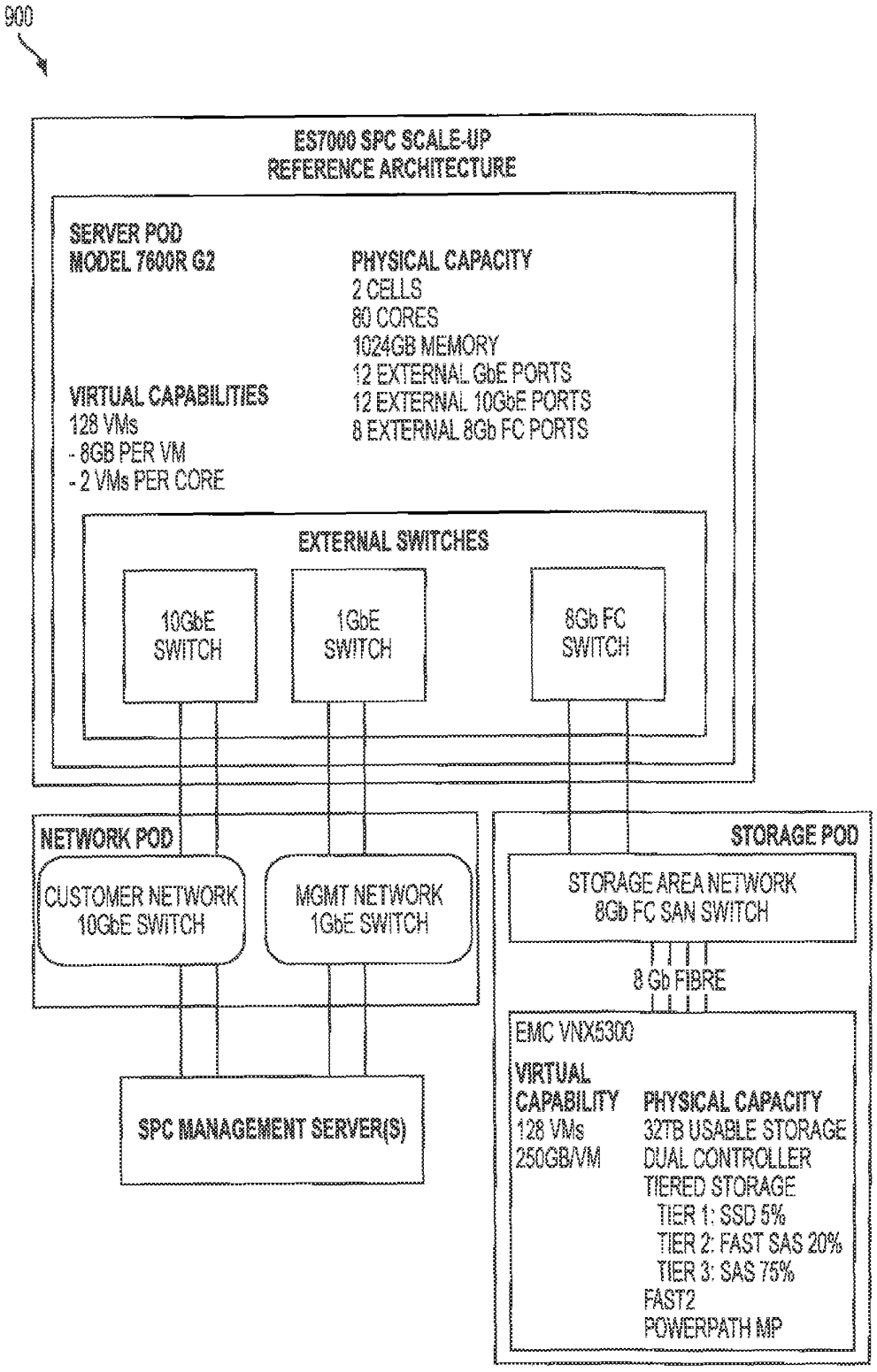
900
ES7000 SPC SCALE-UP
REFERENCE ARCHITECTURE
SERVER POD
MODEL 7600R G2
PHYSICAL CAPACITY
2 CELLS
60 CORES
1024GB MEMORY
12 EXTERNAL GbE PORTS
12 EXTERNAL 10GbE PORTS
8 EXTERNAL 8Gb FC PORTS
VIRTUAL CAPABILITIES
128 VMs
- 8GB PER VM
- 2 VMs PER CORE
EXTERNAL SWITCHES
10GbE SWITCH
1GbE SWITCH
8Gb FC SWITCH
NETWORK POD
CUSTOMER NETWORK 10GbE SWITCH
MGMT NETWORK 1GbE SWITCH
STORAGE POD
STORAGE AREA NETWORK 8Gb FC SAN SWITCH
8 Gb FIBRE
EMC VNX5300
VIRTUAL CAPABILITY
128 VMs
250GB/VM
PHYSICAL CAPACITY
32TB USABLE STORAGE
DUAL CONTROLLER
TIERED STORAGE
TIER 1: SSD 5%
TIER 2: FAST SAS 20%
TIER 3: SAS 75%
FAST2
POWERPATH MP
SPC MANAGEMENT SERVER(S)

FIG. 9

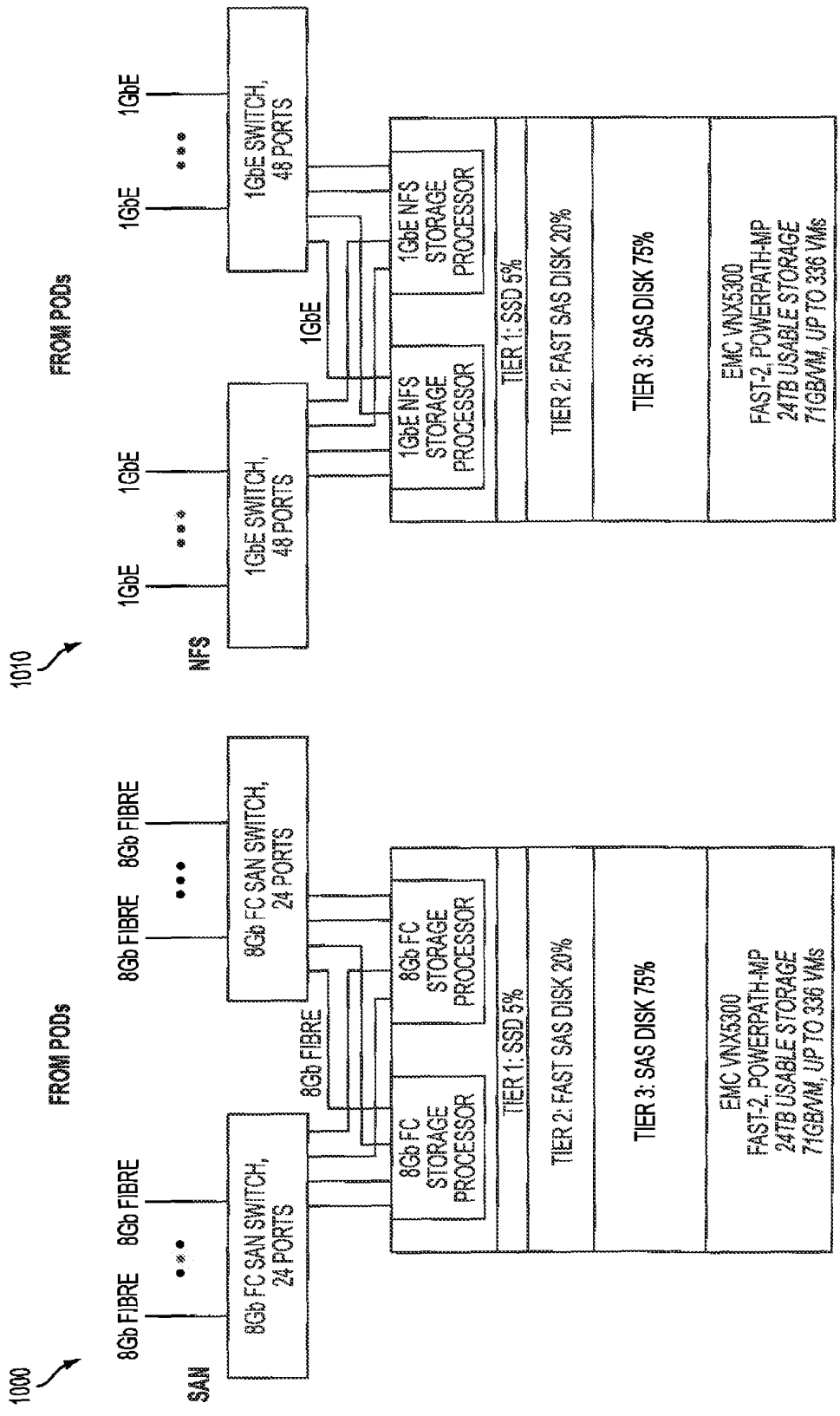
1000
SAN
8Gb FIBRE
8Gb FIBRE
8Gb FC SAN SWITCH, 24 PORTS
8Gb FIBRE
8Gb FIBRE
8Gb FC SAN SWITCH, 24 PORTS
8Gb FIBRE
8Gb FC STORAGE PROCESSOR
8Gb FC STORAGE PROCESSOR
TIER 1: SSD 5%
TIER 2: FAST SAS DISK 20%
TIER 3: SAS DISK 75%
EMC VNX5300
FAST-2, POWERPATH-MP
24TB USABLE STORAGE
71GB/VM, UP TO 336 VMs
FROM PODs
1010
NFS
1GbE
1GbE
1GbE SWITCH, 48 PORTS
1GbE
1GbE
1GbE SWITCH, 48 PORTS
1GbE
1GbE NFS STORAGE PROCESSOR
1GbE NFS STORAGE PROCESSOR
TIER 1: SSD 5%
TIER 2: FAST SAS DISK 20%
TIER 3: SAS DISK 75%
EMC VNX5300
FAST-2, POWERPATH-MP
24TB USABLE STORAGE
71GB/VM, UP TO 336 VMs
FROM PODs

FIG. 10