2015 15th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing

Statistical Characterization of Business-Critical Workloads Hosted in Cloud Datacenters

Siqi Shen[†], Vincent van Beek[†‡], and Alexandru Iosup[†]
† Delft University of Technology, Delft, the Netherlands.
‡ Bitbrains IT Services Inc., Amstelveen, the Netherlands.
Email: {S.Shen,A.Iosup}@tudelft.nl, vincent.vanbeek@bitbrains.nl

***Abstract*—Business-critical workloads—web servers, mail servers, app servers, etc.—are increasingly hosted in virtualized datacenters acting as Infrastructure-as-a-Service clouds (cloud datacenters). Understanding how business-critical workloads demand and use resources is key in capacity sizing, in infrastructure operation and testing, and in application performance management. However, relatively little is currently known about these workloads, because the information is complex—large-scale, heterogeneous, shared-clusters—and because datacenter operators remain reluctant to share such information. Moreover, the few operators that have shared data (e.g., Google and several supercomputing centers) have enabled studies in business intelligence (MapReduce), search, and scientific computing (HPC), but not in business-critical workloads. To alleviate this situation, in this work we conduct a comprehensive study of business-critical workloads hosted in cloud datacenters. We collect two large-scale and long-term workload traces corresponding to requested and actually used resources in a distributed datacenter servicing business-critical workloads. We perform an in-depth analysis about workload traces. Our study sheds light into the workload of cloud datacenters hosting business-critical workloads. The results of this work can be used as a basis to develop efficient resource management mechanisms for datacenters. Moreover, the traces we released in this work can be used for workload verification, modeling and for evaluating resource scheduling policies, etc.**

I. Introduction

Spurred by a rapid development of hardware and of resource management techniques, cloud datacenters are hosting an increasing number of application types. Over a billion people access daily a diverse collection of free or paid cloud utilities, from search to financial operations, from online social gaming to engineering [1]. To continue the adoption of cloud datacenters, and to improve the ability of the datacenter operators to tune existing and to design new resource management techniques, understanding of the workload characteristics and the underlying datacenters is key for both datacenter operators and for cloud service providers. Although some of the largest datacenter operators, that is, Google, Facebook, Microsoft, and Yahoo, have contributed small subsets of workload information that were later used in valuable studies [2]–[5], the information they have contributed represents a relatively small part of the cloud service market. To better understand the workloads of cloud datacenters, in this work we collect and analyze workload traces from a distributed datacenter servicing a fundamentally different workload, that of business-critical applications of financial institutions and engineering firms.

The rapid adoption of cloud datacenters is leading to significant changes in workload structure and, as a consequence, in system design and operation. It is likely that datacenter workloads are becoming increasingly data-intensive, which may put increasingly more stress on the networking, storage, and memory resources of the datacenter [6]. In a previous study of tens of grid workloads [7], we observed that the workload units (jobs, requests, etc.) have decreased in size, and increased in amount and possibly also in interdependency, over the last decade; this could be continuing in cloud datacenters [8]. In response, resource management techniques have also evolved rapidly, with new approaches in computing [9], networking [10], storage [11], and memory [12] management.

The characterization of workload traces is a long-established practice that supports innovation in the design, tuning, and testing of resource management approaches. A recent study [16] uses the characteristics of workloads observed in the Microsoft datacenters to propose and validate an energy-efficient scheduler. Others describe how workload characteristics could help test the robustness of stateful cloud services [17], how MapReduce workload traces could help understand the performance of big data frameworks [3], how the characteristics of traces could help the automated selection of the datacenter scheduling policies [9], etc.

Although actual data and knowledge about workload characteristics are often beneficial for datacenter operation, remarkably few workload traces are publicly available or have even been publicly characterized. Moreover, the few existing examples, albeit seminal, are not comprehensive and, because of their source, may not be representative for the cloud datacenter industry in general. Table I (which we will discuss in detail in Section VII) presents an overview of several of the most-cited studies of cloud workload traces. Overall, the traces originate from Google, Microsoft, and other giant datacenter operators (column TS), and represent workloads that may be typical for the MapReduce and other operations specific to these companies (column Workload). We also observe that few studies include information about requested resources, and rarely include network and disk information at all.

To address the paucity of data and knowledge about datacenter workloads, in this work we aim to characterize the workload of a distributed datacenter servicing enterprise customers with business-critical applications (detailed in Section II-A). We analyze the requested and used resource in (Section III, IV,

978-1-4799-8006-2/15 $31.00 © 2015 IEEE
DOI 10.1109/CCGrid.2015.60

Table I

Previous work in workload trace analysis, in contrast to this study. The trace source (TS) column: F=Facebook, C=Cloudera, G=Google, Y=Yahoo, T=Taobao, I=IBM, BB=Bitbrains. Nodes (N) column: N counts the peak number of VMs; k indicates thousands of items. The traces (Tr) column lists the number of traces. Time (T) column: y/m/d stand for year/month/day. Resources: Mem=Memory, Net=Network.

			Scale			Requested resources		Used resources			
Study	**TS**	**Workload**	**N**	**Tr**	**T**	**CPU**	**Mem**	**CPU**	**Mem**	**Disk**	**Net**
Chen et al. [13]	F/C	MapReduce	5k	7	1 y	—	—	yes	—	yes	—
Reiss et al. [2]	G	Mixture	12.5k	1	1 m	yes	yes	yes	yes	—	—
Chen et al. [3]	F/Y	MapReduce	2.6k	2	7 m	—	—	yes	—	yes	—
Mishra et al. [6]	G	Mixture	?	5	4 d	—	—	yes	yes	—	—
Ren et al. [14]	T	MapReduce	2k	1	2 w	—	—	yes	yes	yes	yes
Di et al. [8]	G	Grid vs Google	12.5k	1	1 m	—	yes	yes	yes	—	—
Birke et al. [15]	I	Industry workloads	?k	1	2 y	yes	yes	yes	yes	yes	—
This study	**BB**	**Business critical**	**1.75k**	**2**	**4 m**	**yes**	**yes**	**yes**	**yes**	**yes**	**yes**

and V), and discuss the limitations and implications of our work (Section VI). Our work can be used as a basis to build workload models, resource usage predictors, efficient datacenter schedulers, etc. Our major contribution is four-fold:

1) We collect long-term and large-scale workload traces from a distributed cloud datacenter (Section II). The traces include information about CPU, memory, disk I/O, and network I/O. We make available these traces through the public Grid Workloads Archive [18][1].
2) We analyze the basic statistics of the requested and actually used resources (Section III). We report the basic statistics, such as quartiles, mean, and standard deviation. We also contrast the basic statistics of business-critical traces with those of parallel production environments, grids, and the search and data-mining workloads of Google, Microsoft, etc.
3) We investigate the time patterns occurring in the resource consumption (Section IV). Specifically, we investigate the peak to mean ratio in resource usage, which we compare with previous datacenter data, and conduct an autocorrelation study of each of the recorded characteristics.
4) We conduct a correlation study to identify possible relationships between different resources (Section V). We also contrast the results with results of previous datacenter studies.

II. Dataset Collection and Method of Characterization

In this section, we introduce two traces representative of business-critical workloads, which we have collected from a distributed cloud hosting datacenter. We also present a method for characterizing such traces.

A. A Typical Cloud-Hosting Datacenter for Business-Critical Workloads

In this work, we study operational traces representative of business-critical workloads, that is, workloads comprised of applications that have to be available for the business to not suffer significant loss. We define business-critical workloads as the user-facing and back-end enterprise services, generally supporting business decisions and generally contracted under strict SLA requirements, whose downtime or even just low performance will lead to loss of revenue, of productivity, etc., and may incur financial loss, legal action, and even customer departure. Business-critical workloads often include applications in the solvency domain; these are often Monte-Carlo simulation based, financial modeling applications. Other applications that characterize business-critical workloads are email, database, CRM and collaborative, and management services, when used in conjunction with the other workloads.

By nature, business-critical workloads are significantly different from the applications that are running in datacenters used by Google's web search/services and data analysis workloads [2], intuitively because these are not contracted under strict SLA requirements; and by Microsoft's Messenger, shared cluster, and Azure [4] datacenters, intuitively because the former does not offer full support for business decisions and the latter two also run other types of workloads. (Our study quantifies this difference.)

A typical mid-size datacenter hosting business-critical workloads is managed by Bitbrains, which is a service provider that specializes in managed hosting and business computation for enterprises. Customers include many major banks (ING), credit card operators (ICS), insurers (Aegon), etc. Bitbrains hosts applications used in the solvency domain; examples of application vendors are Towers Watson and Algorithmics. These applications are typically used for financial reporting, which is used predominately at the end of financial quarters. The workloads are typically of a master worker model where the workers are used to calculate Monte-Carlo simulations. For example, a customer would request a cluster of compute nodes to run such simulations. The following requirements would come with this request: data-transfers between the customer and the datacenter via secure channels, compute nodes leased as virtual machines (VM) in the datacenter that deliver predictable performance, and high availability for running business-critical simulations.

The studied datacenter used VMwareś vCloud suite to host virtualized computing resources for its customers. Bitbrains uses standard VMware provisioning mechanisms, such as Dynamic Resource Scheduling and Storage Dynamic Resource Scheduling, to manage computing resources. One common policy is that memory is not over-committed. This means that the amount of memory requested for a VM can be guaranteed. BitBrains use a power policy, High Performance, that maximizes performance by not using dynamic frequency

[1] These traces can be accessed at http://gwa.ewi.tudelft.nl/datasets/Bitbrains

Table II
Business-critical workload traces collected in this work.

Name of the trace	# VMs	Period of data collection	Storage technology	Total memory	Total cores
`fastStorage`	1,250	1 month	SAN	17,729 GB	4,057
`Rnd`	500	3 months	NAS and SAN	5,485 GB	1,444
Total	1,750	5,446,811 CPU hours		23,214 GB	5,501

scaling features. Bitbrains adopts pricing models that can be usage-based or subscription-based.

In general, Bitbrains hosts three types of VMs: management servers, application servers, and compute nodes. Management servers are used for the daily operation of customer environments (e.g., firewalls). Examples of application servers are database servers, web servers, and head-nodes (for compute clusters). Compute nodes are mainly used to do simulation and other compute-intensive computation, such as Monte-Carlo-based financial risk assessment.

B. Collected Traces

From the distributed datacenter of Bitbrains, we collect two traces of the execution of business-critical workloads. For this we use the monitoring and management tools provided by VMware, such as vCloud suite. For each trace, the vCloud Operation tools record 7 performance metrics per VM, sampled every 5 minutes: the number of cores provisioned, the provisioned CPU capacity, the CPU usage (average usage of CPU over the sampling interval), the provisioned memory capacity, the actual memory usage (the amount of memory that is actively used), the disk I/O throughput, and the network I/O throughput. Thus, we obtain traces that cover both requested and actually used resources, for four resource types (CPU, memory, disk, and network).

We collected two traces between August and September 2013, whose overview we present in Table II. Combined, the traces include data for 1,750 nodes, with over 5,000 cores and 20 TB of memory, and over 5 million CPU hours accumulated over 4 operational months; thus, the traces we collected are long-term and large-scale time series. The first trace, `fastStorage`, consists of 1,250 VMs that are connected to fast storage area network (SAN) storage devices. The second trace, `Rnd`, consists of 500 VMs that are either connected to the fast SAN devices or to much slower Network Attached Storage (NAS) devices. The `fastStorage` trace includes a higher fraction of application servers and compute nodes than the `Rnd` trace, which is due to the higher performance of the storage attached to the `fastStorage` machines. Conversely, for the `Rnd` trace we observe a higher fraction of management machines, which only require storage with lower performance and less frequent access.

The two traces include a random selection of VMs from the Bitbrains datacenter, using a uniform distribution for the probability of selecting each VM. This is motivated by the need to guarantee the absolute anonymity of individual Bitbrains customers and to not reveal the actual scale of the Bitbrains infrastructure. A similar process is used by related work characterizing Google workloads [2], [8], where the anonymization is achieved through a normalization of resource scales and by a selection of only a part of the infrastructure; in contrast, our study is more revealing, in that it presents the full characteristics of the virtualized resources.

Our traces do not include data about arrival processes, which in a cloud datacenter could be used to describe the lifetime of user jobs or of VMs. Instead, we investigate *resource consumption*, which replaces the notion of user jobs with resource usage counters. This also protects the anonymity of Bitbrains' users and is in line with the approach adopted by many previous studies [2], [8]. For VMs, business critical workloads often use the same VMs for long periods of time, typically over several months. Because the VMs we study run throughout the duration of our traces, we do not have a proper arrival process to report on.

C. Method for Workload Characterization

In this work, we conduct a comprehensive characterization of both requested and actually used resources, using data corresponding to CPU, memory, disk, and network resources. Although VMs may change configuration during the trace, the frequency at which this happens is rare in our trace (under 1%), so we show only the initial configuration of each VM present in our traces.

We use three main statistical instruments for statistical characterization: basic statistics, correlations, and time-pattern analysis. For the basic statistics, we report the min and the max, the quartiles, the mean and the standard deviation (SDev), and the unitless "Coefficient of variation" (CoV, defined as the ratio of standard deviation and mean). We also report the cumulative distribution function (CDF) of the values observed for all VMs, and for the CoV observed per VM (a measure of *dynamicity* that extends previous work [2]).

To identify time patterns in our time series, for each resource type we analyze its aggregate usage over time, by summing, each hour, the average resource usage observed for all the VMs. This *aggregate resource usage* can be used to assist resource capacity planning. We plot the auto-correlation function (ACF, a strong indicator for the existence repeating patterns) of the workload traces for each aggregate resource usage. In addition, we analyze *dynamicity* [2], expressed as the ratio of peak to mean values, which we compute for hourly and daily intervals.

To understand the dependency between the different resources, and between requested and used resources, we look at two traditional instruments: the Pearson correlation coefficient (PCC), which measures the linear relationship between two variables, and the Spearman rank correlation coefficient (SRCC), which measures the dependence between two ranked series (e.g., ranked by time). We report here overall results that summarize all VMs but also, where the process is dynamic (e.g., resource usage), the CDF and the probability density function (PDF).

Table III
STATISTICS OF REQUESTED AND USED RESOURCES FOR BITBRAINS, AND REQUESTED CPU CORES (WHICH AS THE SAME AS USED) FOR GRID AND PARALLEL TRACES. THE SAME INFORMATION IS VERY DIFFICULT TO ASSEMBLE FOR THE STUDIES LISTED IN TABLE I.

Trace source	Properties	Mean	Min	Q1	Median	Q3	Max	CoV	SDev
Bitbrains	CPU requested [GHz]	8.9	2.4	2.93	5.20	10.40	86	1.3	11.1
Bitbrains	CPU usage [GHz]	1.4	0.0	0.02	0.08	0.20	64	3.3	4.4
Bitbrains	Memory requested [GB]	10.7	0.0	1.27	3.98	15.59	511	2.8	29.3
Bitbrains	Memory usage [GB]	0.6	0.0	0.03	0.10	0.29	384	3.0	1.8
Bitbrains	Disk, Read throughput [MB/s]	0.3	0.0	0.00	0.00	0.00	1,411	14.9	5.2
Bitbrains	Disk, Write throughput [MB/s]	0.1	0.0	0.00	0.00	0.01	188	14.4	1.1
Bitbrains	Network receive [MB/s]	0.1	0.0	0.00	0.00	0.00	859	11.3	0.7
Bitbrains	Network transmit [MB/s]	0.1	0.0	0.00	0.00	0.00	3,193	24.0	1.5
Bitbrains	CPU cores	3.3	1	1	2	4	32	1.2	4.0
Bitbrains	`Rnd` CPU cores	2.8	1	1	2	4	32	1.1	2.9
Grid	`DAS2` [18] CPU cores	4.3	1	1	2	4	128	1.5	6.4
Grid	`Grid5000` [18] CPU cores	5.8	1	1	1	2	342	3.6	21.0
Grid	`NorduGrid` [18] CPU cores	1.1	1	1	1	1	64	1.2	1.3
Parallel	`CEA Curie` [19] CPU cores	713.3	1	4	32	256	79,808	5.8	4,116.7
Parallel	`LLNL Atlas` [19] CPU cores	423.4	8	8	64	256	9,120	2.9	1,249.0

III. RESOURCE REQUESTED AND USAGE

In this section, we analyze, in turn, the requested resources, the used CPU and memory resources, and the used disk and network resources. Understanding the basic statistics can lead to interesting insights into the operation of the datacenter and into the structure of business-critical workloads, and can help create benchmarks and tune resource-management approaches. Table III summarizes the results, which are further analyzed in this section. Unless otherwise specified, for Bitbrains we present here only results obtained for the `fastStorage` trace; for `Rnd` results, which are very similar (for example, see Figure 1), we refer to our technical report [20]. The main findings are:

1) Over 60% of VM requests are for no more than 4 CPU cores and 8 GB of memory (Section III-A).
2) The resource usage for most VMs is dynamic. The mean CoV for resource usage ranges from less than 1 to more than 20. The lowest CoV is observed for CPU and memory—CoV values under 5 (Section III-B).
3) On average, VMs read 3 times more than write, and use the network to send as much as they receive (Section III-C).

A. Requested Resources

In this section, we analyze the requested resources (only CPU and memory, as disk and network do not record such requests). We find that VMs in our traces require on average similar amounts of CPU cores as typical grid workloads, that most of VMs have modest requirements for CPU cores (at most 4) and allocated memory (at most 8 GB), and that power-of-two requests are common.

First, we compare the CPU characteristics of VMs supporting business-critical workloads (rows labeled Bitbrains in Table III) and of representative traces from grid and parallel production environments. The rows including "CPU cores" in Table III list the number of CPU cores requested (and reported as used by all resource managers) in these workloads: `fastStorage` and `Rnd` representing business-critical workloads; the `DAS2`, `Grid5000`, and `NorduGrid` datasets representing grid workloads [18]; and the `CEA CURIE` and `LLNL Atlas` datasets representing production parallel workloads [19]. If we view the VM as the unit of submitting workload, our workloads require, on average, slightly more CPU cores than production grids (`NorduGrid`) and slightly less cores than experimental grids (`DAS2` and `Grid5000`), but significantly less CPU cores than the parallel workloads.

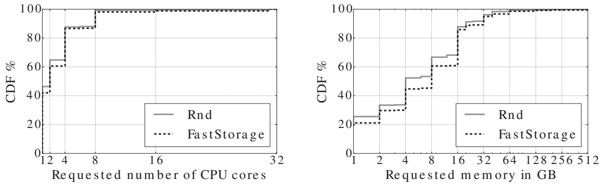

Figure 1. CDF of the (left) number of requested CPU cores and (right) amount of requested memory.

We further characterize the requested resources, in terms of number CPU cores and bytes of memory provisioned to each VM. Figure 1 (left) shows the cumulative distribution function (CDF) of the number of CPU cores requested per VM. For both the number of CPU cores and the amount of memory our results show that a large percentage (more than 60%) of VMs have low requirements (2 or 4 CPU cores for our two traces, and less than 8 GB of memory). Most VMs (over 90%) use power-of-two cores.

Other studies [21] show the power-of-two scale-up behavior, which seems to be historically an artifact of the architecture of parallel architectures and algorithms. VMs in our datasets use from 1 to 32 cores, but more than 85% VMs use 4 or fewer cores. On average, VMs in the `Rnd` dataset use slightly fewer cores, which we ascribe to the higher density of management VMs in the `Rnd` trace—typically, management VMs require only 1 core, and rarely more.

Regarding memory requirements, we observe similar patterns as for CPU requirements. Figure 1 (right) shows the CDF for the requested memory of each VM. Memory is often provisioned in power-of-two quantities (around 90% for memory). For the `fastStorage` dataset, the requested memory can range from 1 GB to 512 GB per VM, but most VMs use a relatively small amount of memory: over 70% VMs use at most 8 GB of memory. The VMs in the `Rnd` dataset

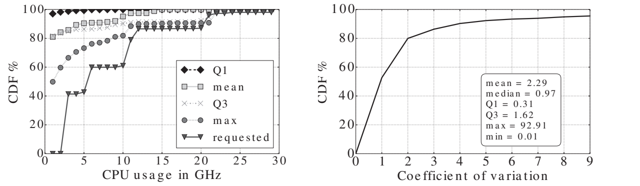

Figure 2. CPU usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

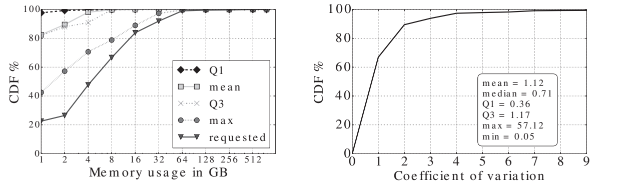

Figure 3. Memory usage: (left) CDF for all VMs, and (right) CDF of CoV observed per VM.

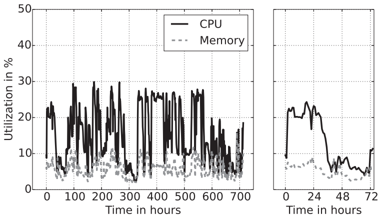

Figure 4. CPU and memory usage over time.

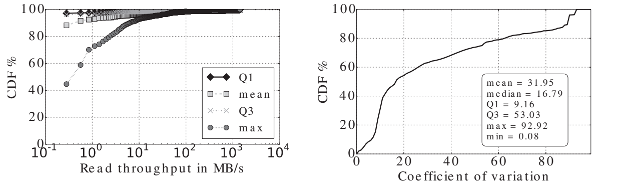

Figure 5. Disk read usage: (top) CDF for all VMs, and (bottom) CDF of CoV observed per VM.

demand slightly less memory than in the `fastStorage` dataset, which we ascribe again to the difference in management VM density—typically, management VMs use 1 GB or less memory.

B. CPU and Memory Usage

In this section, we analyze the CPU and memory resource usage, for which we report both the CDF observed across all VMs, and the CDF of CoV observed per VM. We find that CPU usage is low on average and can be dynamic, and much lower (around 10% for most VMs) than the requested CPU capacity. We also find that memory usage is even lower on average but less dynamic than CPU usage.

Following modern datacenter practice, for example in all VMware-powered clusters, we define the *CPU utilization* as the percentage of CPU cycles used by the VMs from the total CPU cycles allocated to the VMs. We also define a related metric, the *CPU usage*, as the number of cycles provided per second by the VM manager and actually used by the leased VM.

We study first the CDF of CPU usage, across all VMs, and depict the results in Figure 2 (left). The curves in the figure refer to the observed CPU usage—the first-quartile ("Q1" in Figure 2), mean, third-quartile (Q3), and maximal (Max) CPU usage. We also include in Figure 2 a curve, "requested", for the CPU capacity requested for each VM, computed as the product of the number of CPU cores and the speed requested of each core, e.g., 4×2.6 GHz for 4 cores at 2.6 GHz each).

As Figure 2 (left) shows, for most (about 80%) VMs, the mean CPU usage (curve "mean") is lower than 0.5 GHz. Their mean CPU utilizations are lower than 10%. Moreover, only 30% of VMs have a maximal usage higher than 2.8 GHz. In addition, less than 5% of VMs have mean CPU utilizations that are higher than 50%. These observations suggest that for most VMs *the usage is low most of the time.*

Figure 2 (right) indicates that half (50%) of the CoV for CPU usage is lower than 1. This shows that, for half of VMs in our traces, the CPU usage is stable and centered around the mean—these VMs have predictable CPU usage. However, there is still a significant amount (about 20%) of VMs whose CoV for CPU usage is higher than 2—the CPU usage of these VMs is dynamic and unpredictable.

We now study the CDF of memory usage, across all VMs. We construct Figure 3 (left) similarly to Figure 2 (left), but with data about memory usage. We find that the memory usage is low: on average, 80% of VMs use less than 1 GB of memory, and most (about 80%) of VMs have maximal memory usage lower than 8 GB of memory. In Figure 3 (left), the large gap between the "mean" and the "max" curves indicates that the peak memory usage of each VM is much higher than its average usage; we investigate this in more detail in Section IV-A.

Similarly to our study for CPU usage, we investigate next the CDF of the CoV in the observed memory usage, per VM. As Figure 3 (right) shows, the memory usage is less dynamic than CPU usage: about 70% of VMs (vs only 50% for CPU usage) have a CoV for memory usage lower than 1. A similar observation has been reported for the Google trace [2].

It is interesting to study the CPU and memory usage, together; their progress over time can indicate opportunities for VM consolidation and datacenter efficiency. We depict these metrics, over time, in Figure 4. We find that CPU utilization is higher than memory utilization, which is the opposite of the finding of Di et al. [8] for the Google trace. This may suggest that memory resources are more over provisioned than CPU resources in the studied datacenter.

C. Disk and Network Usage

Similarly to Section III-B, in this section we analyze the disk and network resource usage. We find that most VMs have bursty disk and network accesses.

We study the CDF of disk read usage, across all VMs, which we depict in Figure 5 (left). We find that most of VMs only read sporadically: about 95% of VMs reads disk with a speed less than 0.1 MB/s 75% of the time. The mean value and especially the maximal value of disk reads of most VMs is much higher than the Q3 value, which indicates that disk

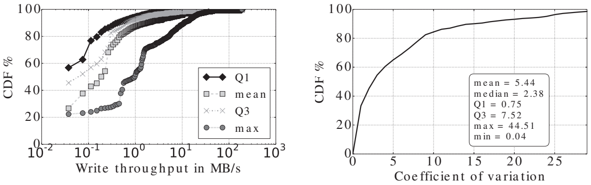

Figure 6. Disk write usage: (top) CDF for all VMs, and (bottom) CDF of CoV observed per VM.

reads are bursty. The CDF for CoV of disk reads is plotted in Figure 5 (right). The disk read usage is much more dynamic than the CPU usage: about 50% of VMs have their disk-read CoV higher than 2. This may be due to application behavior, e.g., backup tools may act periodically, financial modeling tools read large volumes of financial data into memory at the start of simulations, etc.

Similarly to disk reads, we study disk-write usage. The results, which we depict in Figure 6 (left), are similar in trend for disk reads and writes: most of VMs do not write most of the time, but some VMs show very high peak disk write usage. On average, each VM's disk write usage is about 0.1 MB/s, which is about one third of the disk read usage (0.3 MB/s). Comparing to disk reads, we observe that disk writes are less dynamic, as shown in Figure 6 (right). Different from [15], the disk activity of the Bitbrains workload are much more dynamic.

Similarly to disk behavior analysis, we study network usage, expressed in terms of received and transmitted data. On average, for most VMs the amount of data received or transmitted over the network is low. About 80% of VMs receive less than 30 KB/s and transmit less than 10 KB/s. The large gap between the max and the other percentiles, as observed per VM, indicates the bursty nature of network traffic. The amounts of both the received and transmitted are much more dynamic than the CPU usage.

In summary, a typical server in the studied datacenter has rather low but highly variable CPU, memory, disk and network usage. The variations of memory usage are lowest, whereas the variation of disk activities are highest. Such information can be used for designing resource schedulers. Furthermore, as we have shown in this section that there are quite a few differences between the studied workload and others, optimal schedulers designed specially for the workloads of [8], [15] may obtain sub-optimal results for our studied workload.

IV. Time-Patterns in Resource Usage

In this section, we analyze the time patterns of resource usages. Understanding the time patterns of resource usage can help to build smart predictors that estimate upcoming resource usage, and can lead to improved datacenter efficiency. The main findings are:

1) The aggregate resource usage of VMs fluctuates significantly over time.
2) The peak CPU resource usage is 10–100 times higher than the mean (Section IV-A).
3) CPU and memory resource usage can be predicted in short-term.

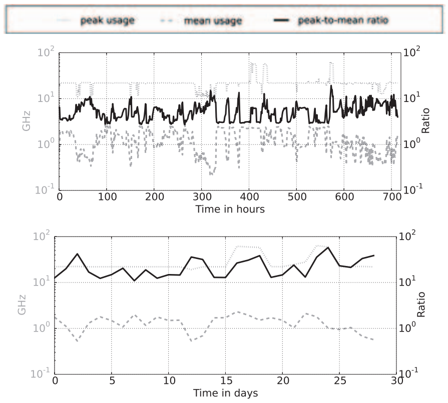

Figure 7. Peak to Mean CPU usage, over time: (top) hourly data; (bottom) daily data.

4) The usage of disk I/O and network I/O show daily patterns, for the `fastStorage` dataset. (Section IV-B).

A. Peak vs Mean Resource Usage

In this section, we analyze how dynamic the business-critical workloads are, and contrast our findings with previously described workloads. To this end, following [22] and [2], we study the peak and mean resource usage, and their ratio, over time. We report both hourly and daily intervals, for all the resources investigated in this work. (Previous studies report this value of intervals that range from 30 seconds [4] to 1 day [2], which makes it difficult to compare results across studies.) Overall, we find that workloads in the studied datacenter are much more dynamic than most previously described datacenter workloads, and more in line with volatile grid workloads. This emphasizes the opportunity to design more efficient resource management approaches, such as dynamically changing the number of active physical resources underlying the leased VMs.

We begin with a focus on CPU usage. Figure 7 shows the peak and mean CPU usage, and their peak-to-mean ratio, per hour and per day. CPU usage fluctuates significantly over time. The daily peak usage can be 10 to 100 times higher than the daily mean usage. This phenomenon is commonly observed in other related workloads: in the Google trace (daily peak-to-mean ratio, 1.3), in the Microsoft Azure trace (15-minute samples, peak-to-mean ratio, 1.7), and in the Microsoft Messenger trace (30-second samples, peak-to-mean ratio ranges from 2.5 to 6.0). The peak-to-mean ratios observed in the studied workload are even higher than the ratios observed in these traces. Iosup et al. [22] analyze 5 grid traces and find hourly peak-to-mean ratios of up to 1,000:1. Similarly, Chen et al. [13] analyze 7 workload traces (from Facebook and Cloudera) and find peak-to-mean ratios ranging from 9:1 to 260:1. These ratios are more in line with the ratios we observe.

Similarly to CPU usage, we analyze the other resources, and find similarly high or even higher peak-to-mean ratios. Both the hourly and daily ratios for disk-read usage are much higher than the ratios observed for CPU usage: we observe 1:1000 and even 1:10,000 ratios. We find similar numbers for

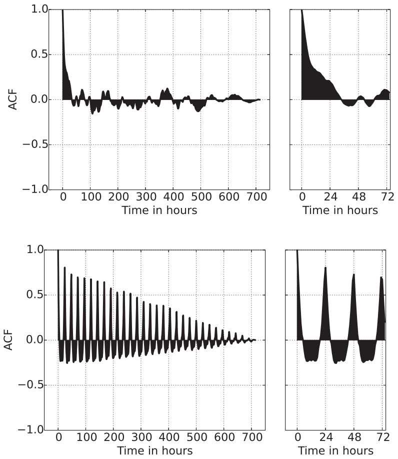

Figure 8. Auto-correlation function: (top) CPU usage and (bottom) Disk read.

disk-write usage [20]. Moreover, ratios for network-transmit usage have the same order of magnitude as for disk usage (including the occasional 1:10,000).

B. Time Patterns through Auto-correlation

In this section, we investigate the presence of time patterns in the usage of resources observed for the studied datacenter. To this end, we conduct an analysis using the auto-correlation (ACF) tool. For all resources, we identify high ACF for small lag, which indicates predictable resource usage in the short term (that is, a few hours). We also find strong daily patterns in disk activity and, somewhat less strong, in network activities.

We analyze the ACF for all types of resource usage, for lag values from 0 hours up to 1 month, with a 1-hour step. Figure 8 (top) depicts the ACF values for CPU usage. The ACF values for the first 10 lags ranges, for all resource usage types, from 0.7 to 0.8, which is high and indicates strong auto-correlation. This indicates that, for all resource usage types, the resource usage is predictable in the short-term (up to a few hours). For disk read, as is shown in Figure 8 (bottom), the ACF curve has local peaks at lag multiples that correspond to days; this indicates that the disk read has a strong daily pattern. We also observe that the disk write and the network I/O follow daily patterns, albeit less pronounced for the network I/O.

V. Dependency Among Resources

Understanding the dependency between resources can help researchers develop better VM consolidation techniques. Based on the observation that the peak resource usage can be much higher than mean resource usage, researchers propose to allocate physical resources to VMs based on the 95-th percentile of usage [23]. This approach overlooks correlations between the usage of different VM resources, and may thus ignore an important cause of overloads in consolidated VMs. Verma et al. [24] propose a method to consolidate VMs based on correlation of CPU usage between VMs. However, they consider only the CPU resource, which can lead to sub-optimal VM consolidation. To achieve better VM consolidation, an in-depth understanding of the dependency between the usage of different resources is needed.

In this section, we analyze the pair-wise dependency between the requested resources (e.g., requested CPU and memory), the dependency between the request and the actual resource usage, and the pair-wise dependency between used resources (e.g., between the CPU and memory usage). We study the dependency using two correlations: PCC and SRCC (described in Section II-C). The main findings are:

1) CPU and memory are strongly correlated for requests (Section V-A), but much less correlated for usage (Section V-B).
2) Request and use are very weakly correlated (Section V-A).

A. Correlation of Requested Resources

In this section, we investigate the correlation between the two types of requested resources, CPU and memory, and find a strong correlation between them. We also investigate the correlation between requested and used resources, and find a very weak correlation.

For the `fastStorage` dataset, the PCC and SRCC between the number of CPU cores and memory are 0.81 and 0.90, respectively. For the `Rnd` dataset, the PCC and SRCC are 0.82 and 0.85, respectively. This indicates that VMs with high values for the requested CPU tend to also have high values for the requested memory, especially for VMs in the `fastStorage` dataset. We confirm this result through an interview with the engineers of the studied datacenter, confirming that operators of the datacenter typically maps either 2 GB or 4 GB memory to a CPU core, depending on the physical CPU-to-memory ratio of the underlying physical infrastructure. For memory-intensive workloads, they set the memory to 16 GB per core. At the other extreme of the CPU-to-memory ratio, small VMs (1 GB or less memory) are typically management VMs that are needed to operate the customer environments.

For both the `fastStorage` and the `Rnd` datasets, the requested and the used resources are weakly correlated. This is indicated visually by the left plots of Figures 2, 3, and Figure 4: the CPU and memory utilizations are low most of the time.

B. Correlation of CPU and Memory Usage

We analyze the correlation of CPU usage and memory usage, for which we report an average correlation. Because both CPU and memory usage vary over time, we also report CDFs and PDFs of the correlation observed over time, per VM. We find strong correlation between high CPU and memory usage, that is, VMs that exhibit high CPU usage are very likely to also exhibit high memory usage. However, the temporal correlation is much weaker: it is less likely that VMs exhibit high CPU and memory usage *at the same time*. This gives, for the future, interesting opportunities to host business-critical workloads more efficiently inside the datacenter.

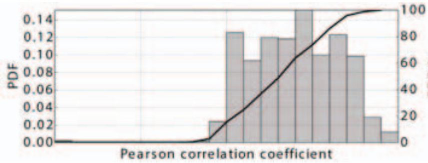

Figure 9. Correlation between CPU usage and memory usage, CDF and PDF of PCC over time.

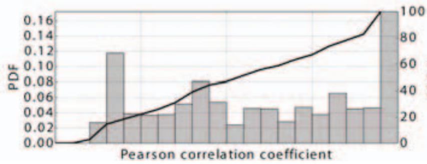

Figure 10. Correlation between network receive usage and network transmit usage, CDF and PDF of PCC over time.

We analyze first the average correlation, that is, the correlation between mean CPU and mean memory usage. For the `fastStorage` dataset, the PCC and SRCC of the mean CPU usage and mean memory usage, per VM, are 0.83 and 0.84, respectively. For the `Rnd` dataset, the PCC and SRCC are 0.72 and 0.83, respectively. This indicates that VMs with high CPU usage tend to have high memory usage. Ren et al. [14] report that for the Taobao system the PCC between CPU usage and memory usage is 0.76, which falls within the same range as our overall result.

Although the average correlation is strong, the temporal nature of both CPU and memory usage requires more in-depth analysis. We thus report CDFs and PDFs of the correlation observed over time, per VM, e.g., we collect data about the CPU-memory correlation for each sampling point (every 5 minutes, as indicated in Section II-B), and we analyze the CDF and the PDF of this dataset.

In Figure 9 we show the distribution of PCC for CPU usage and memory usage. The mean PCC for CPU and memory usage for the `fastStorage` dataset is 0.4, which is much lower than Ren et al. [14] report, and also much lower than we found when we compared CPU and memory required or used on average.

C. Correlation of All Resource Usage

To get a better understanding of correlations between the usage of different resource types, we conduct a comprehensive analysis of all the possible pair-wise combinations (as throughout this work, the resource types considered are CPU, memory, disk read, disk write, network transmit, and network receive). We find low correlations between CPU usage and the usage of other resource types, and even lower correlation between disk and network resources. We also find that about 25% of VMs in our study exhibit strong network transmit and network receive correlation, but either strongly positive or strongly *negative*; the remaining VMs exhibit the low correlation trend we have observed for other resources.

The correlation between CPU usage and network receive and network transmit is very low. The majority of the pair-wise correlations between CPU usage and network usage are between 0.0 and 0.5. These values are much lower than, for example, the correlation values between 0.8 and 0.9 found for the values for requested CPU and memory. The correlations between disk read and network transmit are even lower than what we observe for CPU usage, and the usage of disk and network resources. This observation holds for all other pair-wise correlations of disk and network usage.

Figure 10 shows the correlation between network transmit and network receive. We observe that for the majority of VMs the correlation between sending and receiving network traffic is very low. However, about 16% of VMs have a strong *positive* correlation between sending and receiving network traffic, and about 8% of VMs have a strong *negative* correlation between sending and receiving network traffic. We conclude that network receive and network transmit have more diverse pattern of correlations than other resources.

VI. Limitations and Implications

In this section we list limitations of this work and the measures we take to mitigate them in Section VI-A, and then we discuss how can datacenter researchers and practitioners make use the findings of this works in Section VI-B.

A. Limitations

Dataset size: Unrepresentative datasets can lead to misleading characterization. Compared to other workload traces surveyed in this work (see Table I and Section VII), but not necessarily public, our traces are of medium size, in both the period and the number of nodes they cover. Our traces are also of medium size in comparison with the public traces collected from parallel [19] and grid [18] environments. Thus, our results suffer from this as much as results of studies derived from other traces in the field. Because this information is not publicly available, we can only argue that the datacenter size we considered in this work is more common in the industry as a whole than the Google, Facebook, and Microsoft datacenters.

Data collection tools: The data collection tool can cast doubts on the validity of the dataset. We rely on the tools provided by VMware, which are currently used by thousands of medium and large businesses, and thus can be considered a de-facto industry standard.

Trustworthy analysis: Mistakes in analysis occur often, in many fields of applied statistics. To alleviate this problem, in lack of a validation study conducted by a third-party laboratory, our statistical analysis is conducted by two of the authors, independently; the results have matched. We also release the data for public audit and open-access use.

Collaboration with an industry partner: Analysis in which a participant has a vested interest could lead to biased results. To alleviate this problem, in lack of a multi-party industry consortium, we have collected and analyzed two traces. We note that the studies presented in Section VII have the same limitation, but most rely on a single trace.

B. Implications

Capacity planning of distributed systems rely heavily on the usage of representative workloads [25], [26]. By studying the evolution of the resource demands of VMs at each datacenter, datacenter operators can plan physical hosts in advance of hosting VMs. In this work, we study the ACF of resources, which can serve as a basic to build time-series models (e.g., auto-regressive) to predict future resource demands. Researchers can use our findings and the traces we released to develop and verify their workload-prediction models.

VM consolidation and migration. In this work, we study resource requests and demands for VMs, which can serve as a basic to study VM consolidation [27], [28] and migration techniques [29], [30]. We study the dependency across resources by analyzing correlations between resource usage, where a negative correlation implies opportunities for consolidation. The information provided in this work can be used to guide the experimental setup of simulation studies regarding multi-resource provisioning. Moreover, researchers can leverage our traces to verify their own consolidation techniques. We are building a scheduler that migrates VMs to remote physical hosts, based on the resource usage of the VM and the load of each physical hosts.

VII. Related work

In this section we present a comprehensive comparison between our work and related work, along three axes: contributions related to benchmarks, to public datasets, and to workload characterization in datacenters.

Benchmarks. Our work collects performance logs from a production datacenter. The applications that are hosted in the datacenter are "blackboxes"; that is, we cannot reveal and cannot change the exact behaviors of these applications. This is different from benchmarking research, which selects representative applications and input datasets, and executes the applications in controlled environments. Examples of benchmarks in the datacenter space are C-meter [31] and CloudBench [32] for general cloud workloads, BigBench [33] for big data processing, GRAPHALYTICS [34] for big data graph-processing, and BigDataBench [35] for a variety of workloads.

Dataset release. Our data release complements well the few datasets that are publicly available. Many previous datacenter studies have used the workloads of distributed systems, from parallel [19] and grid [18] environments. The seminal Google workload dataset [2], released in 2011, includes only CPU, memory, and disk characteristics, and only *normalized*, rather than actual, values. The public SWIM workloads repository includes 5 workload traces, possibly extracted from publicly characterized MapReduce traces at Facebook [13], but they are very short (only 1 day) and include no information about memory, network, or number of CPUs. We are not aware of other public datasets; notably, from the important studies [15]. Our main contribution here is the release of a dataset representative for a new type of workload, that is, business-critical jobs in cloud datacenters.

Workload characterization. Table I summarizes the comparison of our work with previous studies. Overall, our study is derived from an average-sized dataset, but focuses on a different workload, and includes a more comprehensive resource view (four types of resources, including the rarely studied disk and network I/O). Our study also conducts a detailed study of both requested and used resources, something that most public datacenter-studies are lacking. We have already compared the results obtained in this work with results from previous studies, when possible.

Closest to our work, Reiss et al. [2] analyze the workloads of Google. They use a relatively limited dataset in comparison to ours, and do not cover disk and network I/O. Because the Google workloads do not match the profile of business-critical applications, we observe significantly different results. For example, in the Google trace, the actual workload is relatively stable, whereas our results indicate that CPU and memory workloads change frequently for business-critical applications. We have indicated other differences throughout this work. Birke et al. [15] analyze the resource usage of VMs in several datacenters. The authors characterize the CPU, memory, disk, and file system usage per VM, and investigate the correlations between usage of resources. However, their target workload is different from ours, in that industrial workloads seem to have very different characteristics from business-critical workloads. They do not investigate the important network resource. Moreover, in the Birke et al. study [15], all data are normalized and not publicly available, and most results are monthly averages, which contrasts to our characterization goals and achievements.

Also related to our work: Di et al. [8] analyze the workloads of Google (with the same dataset limitations as Reiss et al. [2]) and compare them with Grid/HPC systems regarding job length and host load, Mishra et al. [6] propose a workload classification approach and apply it to a four-days trace from a Google datacenter, Gmach et al. [5] analyze workload demands in terms of number of CPUs from an HP datacenter, Kondo et al. [36] characterize failure of desktop grids, and Jia et al. [37] characterize OS-level characteristics of data analysis workloads.

Other types of datacenter workloads are complemented by our study: Chen et al. [3] analyze MapReduce traces from Yahoo and Facebook regarding the input/output ratio, job count, job submission frequencies, etc.; Guenter et al. [4] analyze the workload traces from Microsoft's Live Messenger, Azure, and a shared computing cluster; Chen et al. [16] analyze the workloads of login rates and connection counts in Microsoft's Live Messenger cluster; and Benson et al. [38] study network-level traffic characteristics of datacenters.

VIII. Conclusion and Ongoing Work

Understanding the workloads of cloud datacenters is important for many datacenter operations, from efficient capacity planning to resource management. In this work, we collect

2 large-scale and long-term workload traces from 1,750 virtual machines from a distributed datacenter hosting business-critical workloads. We analyze both requested and actual resource usage in these traces, in terms of CPU, memory, and disk I/O and network I/O. We also compare these findings with previous studies of workloads from search datacenters, parallel and grid environments, etc. Our main findings from the workloads we collected, as reported in this article and detailed in a technical report [20], are:

1) More than 60% of VMs use less than 4 cores and 8 GB of memory. There is a strong positive correlation between requested CPU and memory.
2) Resource usage is low, under 10% of the requested resources, and the correlation between requested and used resources is also low.
3) Peak workloads can be 10–10,000 times higher than mean workloads, depending on resource type.
4) The CPU and memory resource usage are often predictable over the short-term. Disk and network I/O follow daily patterns.

We are currently extending this work with more in-depth statistical and time-series analysis, and further comparison with other workload studies. We plan to use the findings to improve the datacenter-wide scheduler at Bitbrains.

Acknowledgements We thank our reviewers and article shepherds. This work is generously supported by BitBrains. Our work is also supported by the National Basic Research Program of China under grant No. 2011CB302603 and No. 2014CB340303, by the Dutch STW/NWO Veni personal grant @large (#11881), by the Dutch national program COMMIT and its funded project COMMissioner, and by the Dutch KIEM project KIESA. The authors thank Hassan Chafi and the Oracle Research Labs for their generous support.

References

[1] U. Schwiegelshohn, R. M. Badia, M. Bubak, et al., "Perspectives on grid computing," *FGCS 2010*, vol. 26, no. 8, Oct. 2010.

[2] C. Reiss, A. Tumanov, G. R. Ganger, R. H. Katz, and M. A. Kozuch, "Heterogeneity and dynamicity of clouds at scale: Google trace analysis," in *SoCC*, 2012, p. 7.

[3] Y. Chen, A. Ganapathi, R. Griffith, and R. H. Katz, "The case for evaluating MapReduce performance using workload suites," in *MASCOTS*, 2011.

[4] B. Guenter, N. Jain, and C. Williams, "Managing cost, performance, and reliability tradeoffs for energy-aware server provisioning," in *INFOCOM*, 2011, pp. 1332–1340.

[5] D. Gmach, J. Rolia, L. Cherkasova, and A. Kemper, "Workload analysis and demand prediction of enterprise data center applications," in *IISWC*, 2007.

[6] A. K. Mishra, J. L. Hellerstein, W. Cirne, and C. R. Das, "Towards characterizing cloud backend workloads: insights from Google compute clusters," *SIGMETRICS Performance Evaluation Review*, vol. 37, no. 4, pp. 34–41, 2010.

[7] A. Iosup and D. H. J. Epema, "Grid computing workloads," *IEEE Internet Computing*, vol. 15, no. 2, pp. 19–26, 2011.

[8] S. Di, D. Kondo, and W. Cirne, "Characterization and comparison of cloud versus grid workloads," in *CLUSTER*, 2012, pp. 230–238.

[9] K. Deng, J. Song, K. Ren, and A. Iosup, "Exploring portfolio scheduling for long-term execution of scientific workloads in IaaS clouds," in *SC*, 2013.

[10] O. Beaumont, L. Eyraud-Dubois, and H. Larchevêque, "Reliable service allocation in clouds," in *IPDPS*, 2013, pp. 55–66.

[11] L. A. Bautista-Gomez, B. Nicolae, N. Maruyama, F. Cappello, and S. Matsuoka, "Scalable Reed-Solomon-based reliable local storage for HPC applications on IaaS clouds," in *Euro-Par*, 2012, pp. 313–324.

[12] O. Agmon Ben-Yehuda, E. Posener, M. Ben-Yehuda, A. Schuster, and A. Mu'alem, "Ginseng: market-driven memory allocation," in *VEE*, 2014, pp. 41–52.

[13] Y. Chen, S. Alspaugh, and R. H. Katz, "Interactive analytical processing in big data systems: A cross-industry study of MapReduce workloads," *PVLDB*, vol. 5, no. 12, 2012.

[14] Z. Ren, X. Xu, J. Wan, W. Shi, and M. Zhou, "Workload characterization on a production Hadoop cluster: A case study on Taobao," in *IISWC*. IEEE Computer Society, 2012.

[15] R. Birke, L. Y. Chen, and E. Smirni, "Multi-resource characterization and their (in)dependencies in production datacenters," in *NOMS*, 2014.

[16] G. Chen, W. He, J. Liu, S. Nath, L. Rigas, L. Xiao, and F. Zhao, "Energy-aware server provisioning and load dispatching for connection-intensive internet services," in *NSDI*, 2008, pp. 337–350.

[17] P. Bodík, A. Fox, M. J. Franklin, M. I. Jordan, and D. A. Patterson, "Characterizing, modeling, and generating workload spikes for stateful services," in *SoCC*, 2010, pp. 241–252.

[18] A. Iosup, H. Li, M. Jan, S. Anoep, C. Dumitrescu, L. Wolters, and D. H. J. Epema, "The Grid Workloads Archive," *FGCS*, vol. 24, no. 7, pp. 672–686, Jul. 2008.

[19] D. Feitelson, "Parallel Workloads Archive," http://www.cs.huji.ac.il/labs/parallel/workload/.

[20] S. Shen, V. van Beek, and A. Iosup, "Workload characterization of cloud datacenter of BitBrains," TU Delft, Tech. Rep. PDS-2014-001, Feb. 2014. [Online]. Available: http://www.pds.ewi.tudelft.nl/fileadmin/pds/reports/2014/PDS-2014-001.pdf

[21] D. Feitelson, "Packing schemes for gang scheduling," in *JSSPP*, 1996.

[22] A. Iosup, T. Tannenbaum, M. Farrellee, D. H. J. Epema, and M. Livny, "Inter-operating grids through delegated matchmaking," *Scientific Programming*, vol. 16, no. 2-3, pp. 233–253, 2008.

[23] H. Yanagisawa, T. Osogami, and R. Raymond, "Dependable virtual machine allocation," in *INFOCOM*, 2013.

[24] A. Verma and G. Dasgupta, "Server workload analysis for power minimization using consolidation." in *USENIX ATC*, 2009.

[25] A. Gandhi, M. Harchol-Balter, R. Raghunathan, and M. A. Kozuch, "Autoscale: Dynamic, robust capacity management for multi-tier data centers," *ACM Trans. Comput. Syst.*, vol. 30, no. 4, 2012.

[26] S. Delamare, G. Fedak, D. Kondo, and O. Lodygensky, "SpeQuloS: a QoS service for bot applications using best effort distributed computing infrastructures," in *HPDC*, 2012.

[27] A. Gupta, L. V. Kale, D. Milojicic, P. Faraboschi, and S. M. Balle, "HPC-aware VM placement in infrastructure clouds," in *IC2E*, 2013.

[28] L. Chen and H. Shen, "Consolidating complementary VMs with spatial/temporal-awareness in cloud datacenters," in *INFOCOM*, 2014.

[29] J. Ahn, C. Kim, J. Han, Y. Choi, and J. Huh, "Dynamic virtual machine scheduling in clouds for architectural shared resources," in *HotCloud*, 2012.

[30] U. Deshpande, B. Schlinker, E. Adler, and K. Gopalan, "Gang migration of virtual machines using cluster-wide deduplication," in *CCGrid*, 2013.

[31] N. Yigitbasi, A. Iosup, D. H. J. Epema, and S. Ostermann, "C-meter: A framework for performance analysis of computing clouds," in *CCGRID*, 2009, pp. 472–477.

[32] M. Silva, M. R. Hines, D. S. Gallo, Q. Liu, K. D. Ryu, and D. D. Silva, "CloudBench: Experiment automation for cloud environments," in *IC2E*, 2013, pp. 302–311.

[33] T. Rabl, A. Ghazal, M. Hu, A. Crolotte, F. Raab, M. Poess, and H. Jacobsen, "BigBench specification V0.1 - BigBench: An industry standard benchmark for big data analytics," in *WBDB'12*, 2012.

[34] A. Iosup, A. L. Varbanescu, M. Capota, T. Hegeman, Y. Guo, W. L. Ngai, and M. Verstraaten, "Towards benchmarking IaaS and PaaS clouds for graph analytics," in *WBDB'14*, 2014.

[35] L. Wang, J. Zhan, C. Luo, Y. Zhu, Q. Yang, Y. He, W. Gao, Z. Jia, Y. Shi, S. Zhang, C. Zheng, G. Lu, K. Zhan, X. Li, and B. Qiu, "BigDataBench: A big data benchmark suite from internet services," in *HPCA*, 2014.

[36] D. Kondo, F. Araujo, P. Malecot, P. Domingues, L. M. Silva, G. Fedak, and F. Cappello, "Characterizing result errors in internet desktop grids," in *Euro-Par*, 2007.

[37] Z. Jia, L. Wang, J. Zhan, L. Zhang, and C. Luo, "Characterizing data analysis workloads in data centers," in *IISWC*, 2013, pp. 66–76.

[38] T. Benson, A. Akella, and D. A. Maltz, "Network traffic characteristics of data centers in the wild," in *IMC*, 2010.