Addressing the Stranded Power Problem in Datacenters using Storage Workload Characterization

Sriram Sankar
Global Foundation Services
Microsoft Corporation
srsankar@microsoft.com

Kushagra Vaid
Global Foundation Services
Microsoft Corporation
kvaid@microsoft.com

ABSTRACT

Datacenter operators face unique challenges to optimally provision power among deployed servers. Allocated server power is frequently over-provisioned and this results in *stranding* of available datacenter power capacity. Standardized power efficiency benchmarks like SPECpower_ssj2008 can be used for determining power allocation, in conjunction with methodologies to estimate the contribution from the disk subsystem. In this paper, we explore a trace-driven methodology for determining power contribution of the storage components. We show the benefits of this methodology as opposed to typical power provisioning used in the industry.

Categories and Subject Descriptors

C.4 PERFORMANCE OF SYSTEMS

General Terms

Measurement, Performance

Keywords

Stranded Power, Storage Characterization, Datacenter Power Provisioning

1. Introduction

Datacenters hosting large scale web services have unique challenges to ensure optimal provisioning of available power capacity amongst the deployed servers. Given the significant cost incurred in building a large-scale datacenter (often in the range of $200M-$300M) [1], it is essential to have methodologies in place to ensure that every watt of provisioned power is put to work for the application base. One of the widely used metrics for measuring datacenter efficiency is PUE (Power Utilization Effectiveness) [2] which is the ratio of total power supplied to the facility divided by the power consumed by the IT load (computing equipment). PUE numbers are used to determine how much of the provisioned power is distributed to the IT load versus the support infrastructure. As an example, a PUE of 2 indicates that for every watt delivered to the IT load, another watt is used by the infrastructure for cooling the servers. Even though PUE is beneficial as a key metric to track datacenter efficiency, it doesn't give us any insight into how much of the provisioned IT load power is actually being used during normal datacenter operation. If a server consumes less power than it is allocated, the remaining power is called *stranded power* and this results in unused capacity that has already been paid for.

Typical industry estimates for building large-scale datacenters range between $10 to $20 per watt for supporting the peak IT load [17]. Every megawatt not utilized can result in an over spend of $10M-$20M which could have been potentially used more effectively. Additionally, using up the build capacity in an existing datacenter can delay the need to build a new datacenter facility, which is a huge expense that can impact the financial performance of an enterprise. This highlights the need to minimize *stranded power* within a datacenter facility, by ensuring intelligent allocation of available power capacity for deploying as many servers as possible without exceeding the capacity limits.

For typical IT enterprises, there are a variety of workloads and server types that are housed in the datacenter, and it is challenging to find the optimal allocation of server power that strands minimal amount of power. In many cases, it is not possible to recreate the production workload characteristics in lab scenarios, which makes it difficult to estimate the actual power draw seen in the datacenter environment. In such cases, datacenter operators usually find it safe to overprovision the allocated power per server. Since this has to be done in the context of minimizing the amount of stranded power, it is important to perform this power provisioning activity based on sound engineering principles and a detailed understanding of the workload characteristics. In the absence of representative workloads, datacenter operators may choose to use a standardized power efficiency benchmark to evaluate the peak power consumption characteristics of the server and then use those results for datacenter capacity planning.

The SPECpower_ssj2008 benchmark [3] (we will refer to this as *SPECpower* through the remainder of the paper) was the first industry effort to put forward an uniform power efficiency metric that could be applied across various server types. *SPECpower* is primarily a CPU/memory intensive benchmark and doesn't stress the disk subsystem. One way to get around this is limitation is to run a disk stress tool (such as IOMeter [4]) alongside *SPECpower* to include the power consumption values from the storage component. However there are concerns with this approach as well, the primary one being that without knowledge of the workload the disk stress tool may give results that overestimate the power that needs to be allocated (especially for servers with large disk subsystems) as compared to what will actually be

WOSP/SIPEW'10, January 28–30, 2010, San Jose, California, USA.

observed in production environments, leading to potential stranding of power when applied across thousands of servers in the datacenter.

In this paper we evaluate approaches for addressing this issue of datacenter power provisioning by using *SPECpower* alongside a trace-driven methodology for determining the actual power contribution of storage subsystem activity.

2. Background

2.1 Hard Disk Drive Power Consumption

A typical hard disk drive consists of multiple rotating surfaces, which are held together by a central spindle. The magnetic media on top of the surfaces are written to and read from by magnetic read/write heads. The head is moved to a specific position on top of the magnetic media by arms (Actuators). These arms are powered by a Voice Coil Motor (VCM). The data is brought underneath the head by the rotating action of the spindle, which is driven by a Spindle Power Motor (SPM). A request is serviced at the disk by first moving the seek head to the desired position (Seek), waiting till the rotating platter brings the required sector underneath the head (Rotational Latency) and transferring the stored bits / writing bits to the sector (transfer).

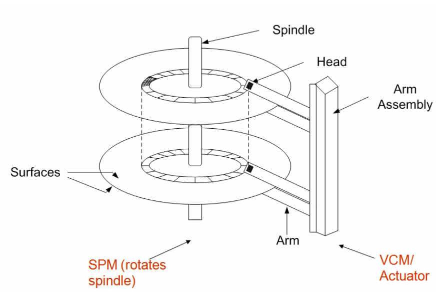

Figure 1: Conventional Hard Disk Drive

For each request serviced at the disk drive, power consumed can be subdivided into these four individual components [5,6]: a) Seek power (power consumed by the actuators / voice coil motors driving the actuators), b) Rotational Power (power consumed by the spindle motor to rotate the disk at a constant RPM), c) Transfer power (power consumed to transfer the bits across the channel) and d) electronic circuitry power losses (power consumed by the amplifiers, controller, cache lines and other electronic circuitry present in the enclosure).

The transfer power and the power consumed by the electronic circuitry can be considered to be less significant [6] compared to power consumed by the mechanical components of the disk drive for most workloads. The major determinant that varies power consumption in a disk drive is the seek action. The voice coil motor that drives the seek activity by moving the actuator (refer Figure 1) accelerates to a particular velocity and then decelerates to position the read/write head on top of the desired track and sector. Queue depth is another factor that affects power consumed by the VCM, since requests could be rearranged so that the seek distance could be reduced. Thus, the disk could be considered to be in two major power states 1. Idle (with just spindle motor power rotating the spindle) and 2. Active (with seek activity consuming additional power). The active state power varies based on the seek characteristics of the workload.

2.2 SPECpower_ssj2008 limitations

The *SPECpower* benchmark has been very popular as the first standardized benchmark for evaluating power efficiency of different server systems. The load values reported by the benchmark can plausibly be used as a starting point for datacenter capacity planning as long as there is good correlation between the benchmark and the workloads deployed. As shown in Figure 2, we found that for one of our most widely deployed workload (Internet Search) which is primarily CPU and memory bound, there is indeed good correlation between *SPECpower* and the application for different throughput load levels. Other workloads, especially scientific computation, may not correlate as well with SPECpower.

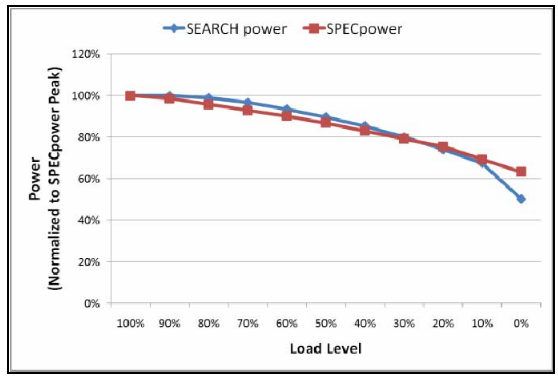

Figure 2: Correlating SPECpower load curve to actual workload

This data point indicates that *SPECpower* can be used as a reasonable proxy for determining power values contributed by the CPU and memory subsystems for peak through idle load levels for workloads like Search. However, one of the major limitations of *SPECpower* is the inability to comprehend the power contribution from the storage subsystem. This is because *SPECpower* is primarily CPU and memory bound and doesn't exercise the disks. To get around this limitation, one could run *SPECpower* (in research mode) alongside a disk stress tool such as IOMeter to simulate activity at the system level. The chart below shows such an experiment for a 16-disk setup, with curves showing basic *SPECpower* (all disks idle), *SPECpower* with 100% random activity on all 16 drives, and with 100% random activity on half the drives, i.e. 8 of the 16 drives. The disk stress is maintained consistent across the benchmark throughput load levels.

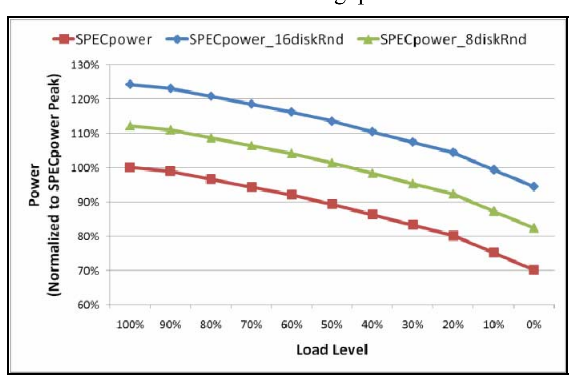

Figure 3: Effect of disk activity on system power

Note that there is a 24% difference in power between the base *SPECpower* run and the 16-disk activity variant. Assuming 100%

random activity simultaneously on all disks to calculate maximum consumed disk power can potentially help with the datacenter capacity planning issue, but there are still some limitations. Actual application workloads may not stress all disks at the same time and hence the maximum power draw may not correspond to fully parallel disk activity. Secondly, the actual disk activity patterns may be not 100% random, and may in fact be a combination of both sequential and random activity.

An example is for SQL server databases, where disk volumes are typically partitioned into DUMP, LOG, TEMPDB and DATA. The TEMPDB and DATA partitions exhibit random I/O characteristics but the LOG and DUMP partitions are accessed in sequential fashion. For getting a reliable estimate for such setups, we need to understand the nature of the disk activity resulting from the workload and the effect it has on power. In Figure 3, the middle curve is representative of a hypothetical workload that exercises only half the disks simultaneously, leaving the other half with minimal activity. Note that this scenario has 12% lower power as compared to the fully parallel 16-disk scenario. When aggregating power values across several thousand servers in the datacenter, over-provisioning of power can lead to less economical usage of the datacenter power delivery infrastructure.

In this paper, we focus on the power provisioning issue as applied to the disk subsystem. To implement a methodology where we can derive storage subsystem power values based on workload activity profiles, we need an infrastructure to capture production traces, analyze these for disk activity levels, and generate synthetic profiles for reproducing the disk activity in a lab environment. The rest of the paper addresses this topic.

2.3 Related Work

Storage power measurement has been researched for implementing power management techniques at the disk subsystem. An analysis of different power consumption at the hard disk drive is presented in [7]. Disk utilization is used to estimate power consumption in [8]. Dempsey [9] is a simulator that can perform per request power estimation along with performance simulation. Power management techniques like spin-down techniques [10, 11] and multi-RPM techniques [5, 6] have been explored in a datacenter scenario. Correlating application behavior to power estimation has been researched in [12]. CPU utilization was correlated to system power in [17] and the notion that power provisioning in datacenters should not be designed for nameplate power ratings or peak power was highlighted. In this paper, we highlight the unavailability of a standardized experimental infrastructure to determine storage power consumption in a real datacenter scenario and use real production traces and storage characterization to estimate storage subsystem power consumption for datacenter power provisioning.

3. Experiment Infrastructure

3.1 Tracing Infrastructure

We use the ETW (Event Tracing for Windows) [13] functionality, provided with Windows operating systems. ETW is a general-purpose, high-speed and scalable tracing facility that can provide Disk and File I/O traces for profiling storage subsystem activity. Kernel-provided buffering and logging mechanisms are leveraged to provide an event based tracing mechanism for events raised by both user-mode applications and kernel-mode device drivers. Windows Server 2003 and 2008 releases allow tracing to be enabled and disabled dynamically without requiring system reboots or application restarts. We capture the following information from production servers for storage events: Event (Disk Read/Write Start, Completion), Timestamp of request, Process issuing the request, Thread id, Virtual address of kernel data structure corresponding to specific IO, Request Offset, Request Size in bytes, Time elapsed, Disk number as viewed by the OS, Flags, Disk service time, Priority, File I/O details like Filename, Object ID etc. With the above level of detail at the storage subsystem, we are able to obtain information about the access profiles for the workloads and analyze them effectively.

3.2 Disk Power Characterization

From Section 2.1 (Hard Disk Power Consumption), we know that seek distribution affects power consumption at the disk drive. We design an experiment to measure the impact of different ranges of seeks, random/sequential patterns, inter-arrival time and queue length on disk power. Since this experiment would be conducted on real disks and not on simulators, we simulate the seek distance range for purposes of estimation, by populating data in fractions of the available disk space (from the outer tracks of the disk drive). We use IOMeter profiles of different workloads to generate the stream of requests for this study. The profiles vary in randomness in fixed percentages (0% - 100% in increments of 25%) and queue lengths from 1 to 256. We present representative graphs here to illustrate the empirical study undertaken on an enterprise class Seagate drive [15]. We used a Fluke power meter [16] to observe real time power variations for each of the profiles along four varying dimensions: %Randomness, Block size, Inter-arrival time and Queue depth.

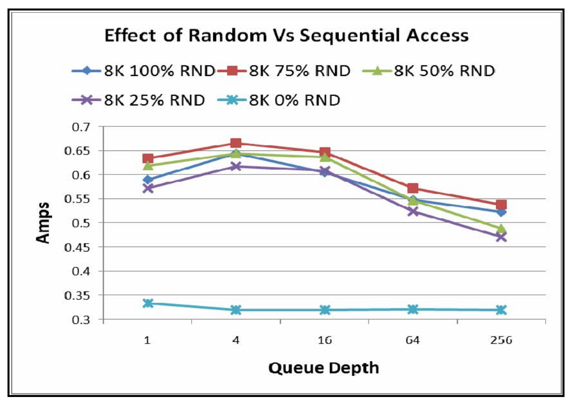

Figure 4: Effect of Random Access

As can be observed from Figure 4, there is a significant difference in current (measured in Amps) consumed by the disk drive for the pure sequential access and other accesses. An interesting trend in this figure is that a 100% random access pattern doesn't necessarily consume more power than a mixed pattern, especially at small queue lengths. At higher queue length the power consumed by the different profiles reduces marginally due to some amount of optimization of seek distances in the presence of a disk queue at the physical disk. We conducted a similar experiment for block sizes and observed that 8K block size consumes more power when compared to larger block sizes that we considered. As block sizes become larger, seeks do not happen as frequently within a given time window and hence the power consumption reduces proportionately.

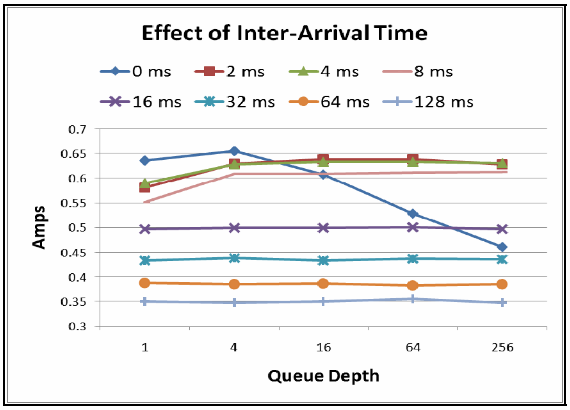

Figure 5: Effect of Inter-Arrival times

Figure 5 shows the effect of inter-arrival time on disk drive power consumption. At 0ms inter-arrival time (a setting that allows the tool to keep issuing requests), we see that larger queue depths allow for more optimization of seek distances and hence it consumes lower power. Increasing inter-arrival time to 2ms, 4ms and 8 ms show marginal power variation since seek activity is still predominant for these inter-arrival times. However, increasing beyond 8ms inter-arrival time, there is a significant drop in amps measured at the disk drive, since idle time increases with increase in inter-arrival time. We use the results of this empirical analysis to calculate the actual disk drive power consumption for our power provisioning estimations.

4. Methodology

The disk power characterization analysis discussed above provides us with per disk power consumption data with respect to particular block sizes, % randomness, queue depth and inter-arrival times. However, at the server level, there are multiple disk drives which could be active at the same time. In this section, we present a methodology to estimate the peak and average power consumption for a given workload. We also discuss how this methodology could be used for efficient power provisioning at datacenter scale.

There are three main steps in the workload characterization based methodology that we propose:

1. Collect traces of representative workloads
2. Analyze traces for workload patterns and disk I/O access
3. Create workload specific power profiles

Collect Traces

We collect production traces from several online services applications. Online services typically have a three tiered hierarchy – front-end web servers, middle-tier application logic and back-end storage and processing layer. In this paper, we present data from the back-end storage layer for two kinds of applications: 1. **BLOB-DB,** which is a SQL database for storing user metadata for generic file content store and 2. **MAPS**, which is a back-end tile storage for a large scale geo-mapping application. **BLOB-DB** is measured on a server with 37 hard disk drives configured as multiple RAID5 and RAID10 volumes, represented by DATA, LOG, TEMPDB and DUMP partitions. There are 20 disks in multiple RAID10 for DATA, 4 disks in RAID10 for LOG and a total of 9 disks in multiple RAID5 volumes for DUMP. For a SQL server setup not all these partitions are accessed simultaneously. **MAPS** is measured on a server with 12 disk drives in a simple RAID0 configuration.

Analyze Traces

We analyze the traces collected through a library of post processing scripts that summarize information about the workloads. We categorize the information into 1. Block sizes and Random access statistics, 2. Performance metrics (IOPS, MBs/sec and Latency) and 3. Power characteristics. We detect the I/O activity to each disk volume and correlate the LBN accesses at the RAID controller level to specific disk drives. We then use our disk power characterization study to correlate the block size, % randomness and queue depth for this particular workload to the power consumed by the disk drives that are active at that instant. We can then compute the power consumed by the storage subsystem for each specific workload.

Figure 6 and Figure 7 show the power characteristics for BLOB-DB and MAPS workloads respectively. For the study presented in this paper, BLOB-DB was measured for a 1 hr time interval, whereas MAPS trace was taken for 30 mins. The theoretical maximum (**Th-Maximum**) power is calculated by multiplying the active power for a single disk drive with the total number of disk drives in the system. This is the theoretical maximum power that is consumed by the entire disk subsystem with the assumption that all disks are simultaneously servicing random access requests. We then analyze the trace to ascertain whether there are indeed any data points when the disk subsystem is utilized to the maximum.

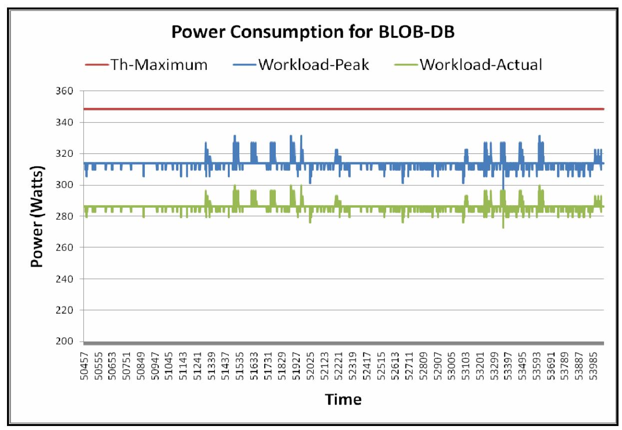

Figure 6: Disk Power Characteristics for BLOB-DB

We find that for BLOB-DB not all 37 disk drives are used simultaneously. The "Workload Peak Power" (**Workload-Peak**) line in the graph corresponds to the measured power for the workload with the peak power assumed for 8K block accesses. However, for BLOB-DB, 80% of the total accesses are 8K accesses with sequential LOG accesses (10% of total) and an average inter-arrival time of 3ms. Also, 90% of the total accesses are random in nature. Scaling the power according to the workload access patterns, % randomness and inter-arrival times, we observe power consumption that is still lower than our estimation for peak power. This "Workload Actual Power" (**Workload-Actual**) is the power that the storage subsystem actually consumes for this particular workload pattern. Note that there is a difference of ~**48** watts between the peak values observed for the *Workload-Actual* power versus the maximum theoretical peak value (*Th-Maximum*). This gap represents the amount of *stranded power* that would be unused if the server power was provisioned without any knowledge of the workload.

Figure 7 shows a similar graph for MAPS. However, there is a case where the *Workload-Peak* power estimation shows that all the disk drives are at Theoretical Maximum (*Th-Maximum*) Power level at some points during the trace. When we estimate the workload power (MAPS is largely sequential, 64K accesses), we do not see *Workload-Actual* power reach close to maximum power (*Th-Maximum*) at any point during the trace. This suggests that even for the MAPS workload which can simultaneously access all 12 disks in the server, we can provision the system at a lesser power budget than the theoretical maximum power.

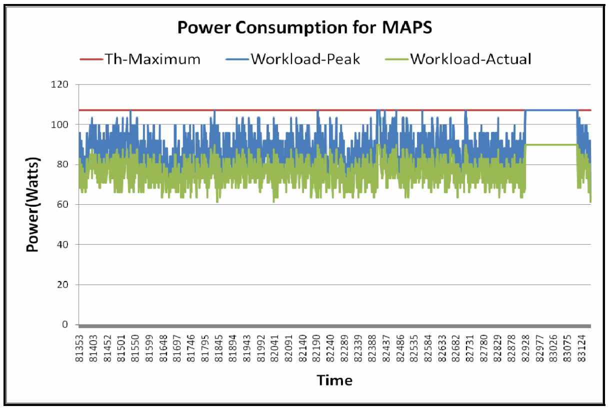

Figure 7: Disk Power Consumption for MAPS

Workload Specific Power Profiles

Once we have the trace, we can create workload specific power profiles from the trace information. The ETW trace captures block size accesses and the post processing scripts calculate random access percentages and inter-arrival times for the workload. We show the current draw for a single disk drive under various inter-arrival times in Figure 8 for the BLOB-DB workload profile. We reproduce the block access characteristics and inter-arrival times in IOMeter [4], a publicly available load generation tool. We vary the transfer delay and burst length parameters in IOMeter to reproduce the temporal characteristics of the workload.

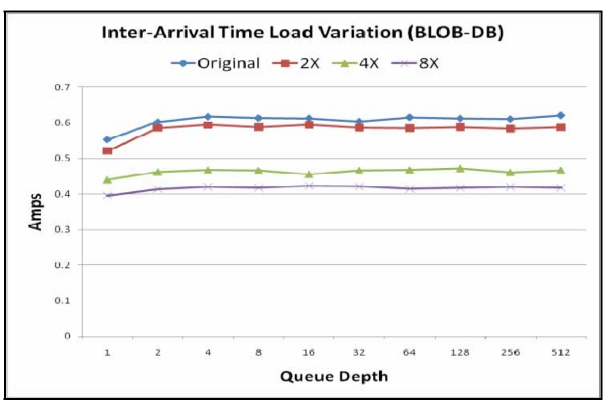

Figure 8: Varying Inter-Arrival Time on BLOB-DB

We observe from the chart in Figure 8 that with increase in inter-arrival time, the current consumed at the disk drive decreases. Since Inter-arrival time represents input load to the disk, we can now generate a load vs power consumption graph similar to Figure 3. We use *SPECpower* and the power estimation from IOMeter runs (Figure 8) to calculate system power estimation for BLOB-DB workload. Inter-arrival time is increased exponentially and the resulting power consumption is estimated.

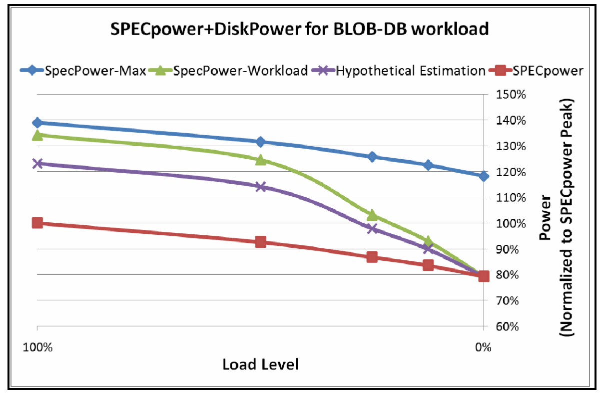

Figure 9: System Power Estimation for BLOB-DB

The "*SPECPower*" curve in Figure 9 is the default power value measured for the 37 disk system. If we use peak disk power estimates, we get the "*SpecPower-Max*" curve at the top of the chart. A better estimation than using SpecPower-Max curve for power provisioning is the "*SpecPower-Workload*" curve, which reflects BLOB-DB specific inter-arrival times, random percentages and block sizes. We can further tune the *SpecPower-Workload* curve by also estimating the number of active disks at any point in time from the trace and by applying the volume specific workload profiles. The resulting curve is shown as the "*Hypothetical Estimation*" curve in Figure 9 (29 disks were active on an average out of the total 37 disks in a BLOB-DB server, out of which 4 disks were LOG partitions with sequential disk activity that results in power consumption close to idle power). The *Hypothetical Estimation* curve gives an estimate of the optimal power efficiency based on an understanding of the workload characteristics.

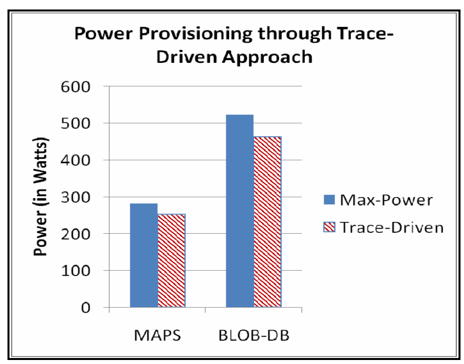

Figure 10: Advantages of Trace-Driven Power Provisioning

Figure 10 compares the power provisioning for a single server from a *Max-Power* based approach (where most datacenter operators use peak power estimates) and a *Trace-Driven* approach (where workload profiles from trace analysis is used for power estimation). We see that we would have over-provisioned power by **12%** for MAPS and by **13%** for BLOB-DB workloads if we use the Max-Power approach. When aggregated across several thousands of servers in a datacenter, this over-provisioning of power can result in significant *stranded power*.

5. Discussion

5.1 Accurate Workload Representation

In this paper we utilized currently available load generation tools and analyzed traces for performance and power characterization of a server. Currently, available tools have certain limitations for accurate power provisioning. To faithfully reproduce a workload that operates at the application layer in a real datacenter, we need to have a model that can capture storage patterns at a finer granularity. For instance, if we observe Figure 8, the power consumption of the 2X inter-arrival time curve is closer to the original inter-arrival curve, whereas the 4X curve is further apart. Inter-arrival time based load variation can provide a good approximation for performance modeling, but power consumption requires control of per request accesses since that would determine activity at each physical disk. IOMeter does not expose this level of granularity to the user. Probabilistic state transition models with LBN ranges [14] could be utilized to develop better workload representation that can accurately capture per disk IO activity and also reproduce IO activity across multiple server disks.

5.2 Cost Analysis of Efficient Power Provisioning

We use a publicly available cost model to estimate datacenter operational costs [1]. In Table 1, we show an illustration of the benefits of accurate power provisioning. From our *Trace-Driven* approach we have a better way of determining power allocation per server, and hence can accommodate more servers in the same critical power budget, resulting in the ability to service 0.9 Million more users in the same datacenter. By increasing the number of servers that can be provisioned in a datacenter, we can effectively accommodate growth in service demand and thereby positively impact cost efficiency at the datacenter level.

Table 1: Datacenter Provisioning for BLOB-DB

Variables	Max-Power	Trace-Driven
Cost of Facility ($):	$200,000,000.00	$200,000,000.00
Cost/Server ($)	$5,000.00	$5,000.00
Size of Facility (Critical Load W):	15000000	15000000
Power/Server (W)	522	463
Number of Servers:	28735	32397
Scale units	1368	**1542**
Users on 1 unit	5000	**5000**
Total Users hosted	6.8M	**7.7M**

Note 1: We assume 17 server units in 1 Scale unit

5.3 Limitations

A trace-based approach is essentially deployment based. A different storage configuration or application version might change the storage access behavior. The trace should be taken when the application exhibits steady state behavior. Rigorous testing is needed to validate and verify application profiles represented in load generator. Given these limitations, we still find that in a real production environment non-obtrusive trace logging is one of the best methods to observe system and application behavior.

6. Conclusion

In this paper we have presented a methodology for estimating disk power consumption through workload storage characterization. We have shown how this could be used in conjunction with a standardized power efficiency tool like *SPECpower* to efficiently provision power for servers in a datacenter. We also present a cost analysis that exposes the benefit of a trace-driven methodology for storage power provisioning as opposed to peak power values typically used otherwise. We believe that this is one of the important mechanisms that should be deployed for addressing the *stranded power* problem in datacenters.

7. REFERENCES

[1] James R. Hamilton, "Cost of Power in Large-Scale Data Centers" (http://perspectives.mvdirona.com, November 2008)

[2] Christopher Malone, Christian Belady, "Metrics to Characterize Data Center & IT Equipment Energy Use", Digital Power Forum, Richardson, TX (September 2006)

[3] SPECpower_ssj2008 http://www.spec.org/specpower/

[4] IOMeter, performance analysis tool. http://www.iometer.org/.

[5] S. Gurumurthi, A. Sivasubramaniam, M. Kandemir and H. Franke. 2003. DRPM: dynamic speed control for power management in server class disks. *SIGARCH Comput. Archit. News* 31, 2 (May. 2003), 169-181.

[6] S. Sankar, S. Gurumurthi, and M. R. Stan. 2008. Intra-disk Parallelism: An Idea Whose Time Has Come. In *Proceedings of the 35th international Symposium on Computer Architecture* (June 21 - 25, 2008)

[7] A. Hylick, R. Sohan, A. Rice, and B. Jones. An analysis of hard drive energy consumption. In *MASCOTS 2008*, pages 103-112.

[8] J. Stoess, C. Lang, and F. Bellosa. Energy management for hypervisor-based virtual machines. USENIX, 2007.

[9] J. Zedlewski, S. Sobti, N. Garg, F. Zheng, A. Krishnamurthy, and R. Y. Wang. Modeling hard-disk power consumption. In *FAST*. USENIX, 2003.

[10] E. Pinheiro and R. Bianchini. Energy conservation techniques for disk array-based servers. In *ICS*, 2004.

[11] S. Gurumurthi, J. Zhang, A. Sivasubramaniam, M. Kandemir, H. Franke, N. Vijaykrishnan, and M. Irwin. Interplay of Energy and Performance for Disk Arrays Running Transaction ProcessingWorkloads *(ISPASS 2003)*, March 2003.

[12] Allalouf, M., Arbitman, Y., Factor, M., Kat, R. I., Meth, K., and Naor, D. 2009. Storage modeling for power estimation. In *Proceedings of SYSTOR 2009: the Israeli Experimental Systems Conference* (Haifa, Israel). SYSTOR '09.

[13] I. Park, R. Buch, "Improve Debugging and Performance Tuning with ETW", Microsoft Corporation, April 2007 http://msdn.microsoft.com/en-us/magazine/cc163437.aspx

[14] S. Sankar and K. Vaid, "Storage Characterization for Unstructured Data in Online Services Applications" (IISWC 2009).

[15] Seagate Barracuda® ES.2 (Server & Enterprise Storage)

[16] Fluke 43B Power Quality Analyzer (http://us.fluke.com/usen/products/Fluke+43B.htm)

[17] Fan, X., Weber, W., and Barroso, L. A. 2007. Power provisioning for a warehouse-sized computer. *ISCA* 2007 (San Diego, California, USA, June 09 - 13, 2007)