# Server Engineering Insights for Large-Scale Online Services

The rapid growth of online services in the last decade has led to the development of large data centers to host these workloads. These large-scale online, user-facing services have unique engineering and capacity provisioning design requirements. The authors explore these requirements, focusing on system balancing, the impact of technology trends, and the challenges of online service workloads.

Christos Kozyrakis
Stanford University

Aman Kansal
Sriram Sankar
Kushagra Vaid
Microsoft

In their insightful 2000 article, Gray and Shenoy offered 22 rules of thumb for designing data storage systems.[1] Focusing primarily on database systems, they evaluated how applications use processors, storage, and networking resources and used that information to derive ratios for balanced system design and project how technology trends would affect performance and price.

A decade later, there have been significant changes in the information technology landscape. Although stand-alone database systems are still an important workload, new distributed applications (such as search engines, webmail, social networks, and software-as-a-service [SaaS]) have emerged as the next frontier of computing. Because of their massive scale and high availability requirements, online services run on large scale-out server farms and require system design principles to be applied not only at the server level but also holistically at the datacenter level.[2] With large facilities including tens of thousands of servers and requiring tens of megawatts of power to host Internet services 24/7/365, infrastructure designers must consider factors such as workload performance characteristics, power and cooling efficiencies, and total cost of ownership (TCO).

In addition to the emergence of a new class of workloads, technology scaling has aggressively progressed. Over the past decade, the industry has witnessed a fundamental shift in the trends for microprocessors and storage systems technology. Scalar performance improvements for microprocessors via continuous frequency scaling have slowed, giving way to throughput performance improvements via an increasing number of cores on each processor (CPU) chip.[3] On the storage front, solid-state storage is gradually complementing or displacing magnetic storage in several application scenarios.[4] With the advent of new memory technologies such as phase-change memory, some long-term projections now indicate that all memory in computer systems could soon be nonvolatile, especially considering DRAM's scaling limitations.[5]

In this article, we use a framework similar to Gray and Shenoy's to observe and study balanced system design for servers hosting large-scale Internet services. We base our work on three key online services at



Published by the IEEE Computer Society

## Representative online services

The three representative services used in this study are Microsoft's largest in terms of the number of deployed servers. In addition, their software and hardware architectures reflect realistic constraints in available technology, development and operations costs, and user patterns. Although considering these applications under alternate assumptions might be interesting, our study focuses on the lessons we can draw from the current state of the art.

### Hotmail

Hotmail represents Microsoft's online email offering serving hundreds of millions of users. Its workload characteristics fundamentally reduce to retrieving data from stored mailboxes in response to user requests. Hotmail stores several petabytes of data spread across tens of thousands of servers.

Hotmail is not only a large-scale application, but is also representative of storage-bound three-tier services. To a certain extent, it can be considered a specialized, scaled version of a database system for online transaction processing that hosts large amounts of data and serves numerous random requests. Although we might expect some locality in a single user's requests, it is insignificant compared to the random nature of metadata access and the multiplexing of I/O requests from many users. The bulk of the request service burden falls on back-end servers storing user mailboxes. The front-end and middle tiers are essentially stateless external-facing servers that implement client connections, various mail-delivery protocols, and supplementary mail processing, such as spam filtering. The back-end servers are significantly heavier weight than the front-end and middle-tier servers and account for a much larger server and design cost. We base our analysis on this most important back-end tier, using a few representative servers. Since the servers are load balanced, the selected servers represent the group as a whole. A typical Hotmail storage server is a dual CPU socket machine with two direct attached disks and an additional storage enclosure containing up to 40 SATA drives in a RAID 1+0 configuration, with a large portion of the disk drives striped for mailbox metadata (SQL) lookup, typically the IOPS intensive portion of the back-end tier.

### Cosmos

Cosmos is a highly parallelized data storage and analysis engine similar to Dryad.[1] It is representative of large-scale data analysis applications including MapReduce[2] and Hadoop (http://lucene.apache.org/hadoop), parallel databases,[3] and other distributed and streaming programming paradigms used to process massive amounts of data.

Cosmos essentially implements a distributed data storage and computation platform on a single tier of servers. The Cosmos cluster used in production consists of several tens of thousands of servers and is under relatively high load, serving a job stream that almost never runs dry. We chose half a dozen servers from the production pool to study the performance metrics. Typical servers are dual CPU socket machines with 16 to 24 Gbytes of memory and up to four SATA disks.

### Bing

Bing is an Internet search application that performs Web crawling, index management, and query lookup. The Bing architecture consists of a front-end tier of user-facing servers with query caches, a middle tier of query distributors and aggregators, and a back-end tier of index lookup servers.

Similarly to Hotmail and Cosmos, the Bing deployment scale consists of several tens of thousands of servers. An important differentiating factor is that Bing represents an emerging class of latency-critical three-tiered services, such as social networks and many online gaming frameworks. Such services typically implement a distributed storage scheme using the main memories of thousands of servers in the data center to avoid the high latency of frequent disk accesses. This trend is evident in the popularity of distributed memory frameworks such as memcached. Note that memory-based storage for huge volumes of data is impractical in the absence of large, scale-out facilities due to hardware limitations. Traditional stand-alone databases either use disk arrays to provide high-bandwidth access to large volumes of data, or in-memory structures to provide low-latency access to small data collections.

Our study focuses on Bing's primary workhorses—the back-end tier servers used for index lookup. These servers are sized such that much of the index can be allocated in memory and disk lookups are avoided for most queries. Most of the servers used in past deployments have been commodity servers with dual CPU socket, about 2 to 3 Gbytes per core of DRAM, and two to four SATA disks.

### References


1. M. Isard et al., ''Dryad: Distributed Data-Parallel Programs from Sequential Building Blocks,'' *Proc. European Conf. Computer Systems* (EuroSys), ACM Press, 2007, pp. 59-72.
2. J. Dean and S. Ghemawat, ''MapReduce: Simplified Data Processing on Large Clusters,'' *Proc. 6th Symp. Operating Systems Design & Implementation* (OSDI 04), Usenix Assoc., 2004, pp. 137-150.
3. D. De Witt and J. Gray, ''Parallel Database Systems: The Future of High Performance Database Processing,'' *Comm. ACM,* vol. 36, no. 6, 1992, pp. 85-98.


Microsoft: Hotmail, the Cosmos framework for distributed storage and analytics,[6] and the Bing search engine (see the ''Representative online services'' sidebar). These services host millions of users, occupy tens of thousands of machines across several data centers, and process petabytes of data. More importantly, they represent the requirements and trends across three different classes of datacenter workloads.

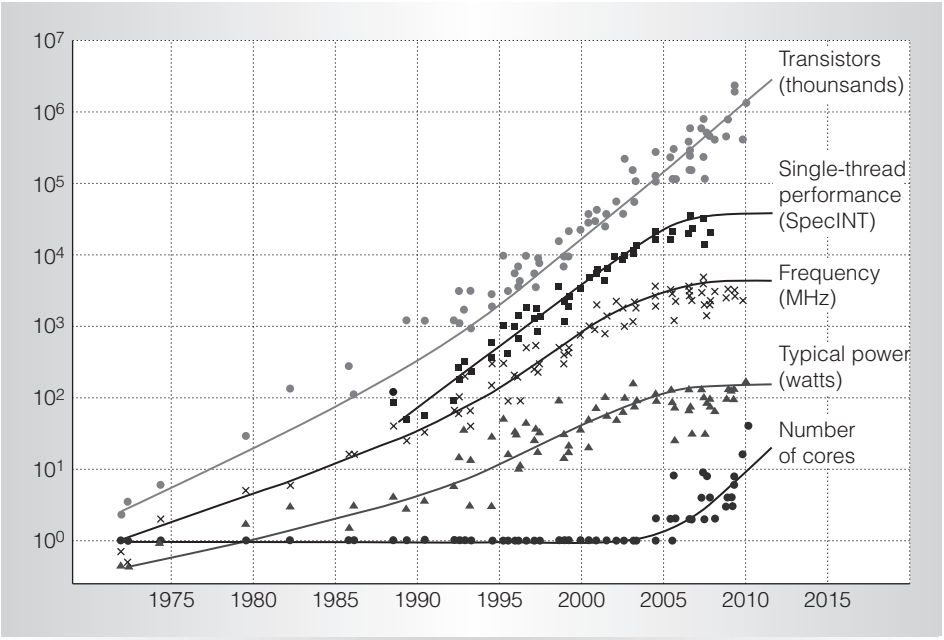


Figure 1. Scaling trends for the transistor count, clock frequency, number of cores, and single-thread performance of microprocessor chips. C. Batten created this figure based on data from M. Horowitz, F. Labonte, O. Shacham, K. Olukotun, L. Hammond, and C. Batten.

## Performance data collection

We obtained performance data from multiple production servers actively hosting the three online services subject to real user workloads. We refer to this performance data as *production data*. While we collected performance data using different mechanisms across the various online services, we abstract out these details for the purpose of presentation and reduce the data to the common, relevant metrics required for our analysis.

Additionally, we obtained the applications' source code in a form that could be executed on our lab servers. Although this code differed slightly from the production versions, it preserved the key computational characteristics and let us stress the applications with increased workloads per server. The stress tests let us study the server performance when operating at its peak capacity rather than the lower workload seen in production servers due to overprovisioning. The performance data from the lab setup is referred to as *stressed data*. We did not study network performance in the stressed tests as these are scaled down to one or a few machines.

Both the production and stressed performance data include performance counters that characterize the CPU usage, privileged CPU usage, disk space used, disk throughput, queue lengths, network data rates, output queue lengths, memory space used, and memory bandwidth when needed. We processed the raw data to extract various statistics such as temporal averages and relevant ratios. Due to space restrictions, we only present summarized data relevant to key observations.

## Server technology scaling for online services

The first part of Gray and Shenoy's study focused on basic technology trends for processors, memory, and storage. Most observations are still valid. The transistor count for processor and memory chips still quadruples every three years as predicted by Moore's law. Hard disk capacity scales at a similar rate.[7] However, the past decade has seen some significant technology shifts. Due to power constraints, processor clock frequency and single-thread performance (latency) have nearly flattened (see Figure 1). The increasing transistor counts are now used for more processor cores per chip (bandwidth). Processors now follow the same trend as memory and hard disks, for which latency improvements lag significantly behind

**Table 1. Processor, memory bandwidth, and disk bandwidth use under stress testing (in percentages).**

| Service | CPU utilization | Memory bandwidth | Disk bandwidth |
|---|---|---|---|
| Hotmail | 67 | N/A | 71 |
| Cosmos | 88 | 1.6 | 8 |
| Bing | 97 | 5.8 | 36 |

**Table 2. Average utilization of memory, storage, and networking resources during production runs over a one-week period (in percentages).**

| Service | Main memory capacity | Memory bandwidth | Disk capacity | Disk bandwidth | Network bandwidth |
|---|---|---|---|---|---|
| Hotmail | 92 | N/A | 75 | 0.91 | 27 |
| Cosmos | 39 | 1.1 | 52 | 0.68 | 9 |
| Bing | 88 | 1.8 | 30 | 1.10 | 10 |

bandwidth improvements.[7] At the same time, the lack of significant latency improvements for hard disks is motivating the use of nonvolatile memory (flash) devices for main storage.[4] Although more expensive per bit of storage, flash devices might be advantageous in terms of cost performance.

With these trends in mind, let's look at how the three online services use CPU, memory, disk storage, and networking. Table 1 shows the use of CPU, memory bandwidth, and disk bandwidth in lab tests, which saturate the servers using heavy input load scenarios. These results indicate the server component that determines the maximum performance attainable from each machine. Table 2 shows the average utilization of memory and storage components observed in production environments. These results represent the typical use of servers and their components. We calculated disk bandwidth use based on the maximum disk bandwidth that the server can achieve for the common block size used by each application. Note that the raw disk bandwidth available varies significantly across services because the corresponding servers include a different number of hard disks. Our stress tests did not exercise the network interfaces, hence we do not report network bandwidth.

For Bing and Cosmos, the maximum CPU utilization reported by the operating system during stress tests reaches almost 90 percent or higher, indicating that processors saturate first (Table 1, column 2). Since for many online services, processor utilization above 80 percent is often undesirable because it impacts quality of service (request latency), the overall service deployment is often provisioned to keep average CPU utilization at moderate levels, typically in the 40 to 60 percent range. Looking at hardware counter statistics for cycles per instruction (CPI) and cache miss rate per instruction (MPI), Bing exhibits modest CPI rates and frequent main memory accesses. Cosmos is similar to data-mining database workloads with low CPI and streaming memory accesses. Hotmail accesses most of its data from disk and hence memory bandwidth is not of interest for this service.

Bing preloads most of the index in memory to avoid slow disk accesses and thus reduce query execution latency. Consequently, Bing fully utilizes the available DRAM capacity (Table 2, column 2). Bing and Cosmos are not bound by memory bandwidth, so the multiple integrated memory controllers in modern processors are far from being a bottleneck (Table 1, column 3). This is the case because Bing is limited by the latency of main memory accesses, whereas Cosmos performs a significant amount of computation on its input data. Bing uses disk storage primarily for logging, while Cosmos uses it for storing distributed data sets. Neither application is bottlenecked by the network and disk bandwidths (Table 2, columns 5 and 6).

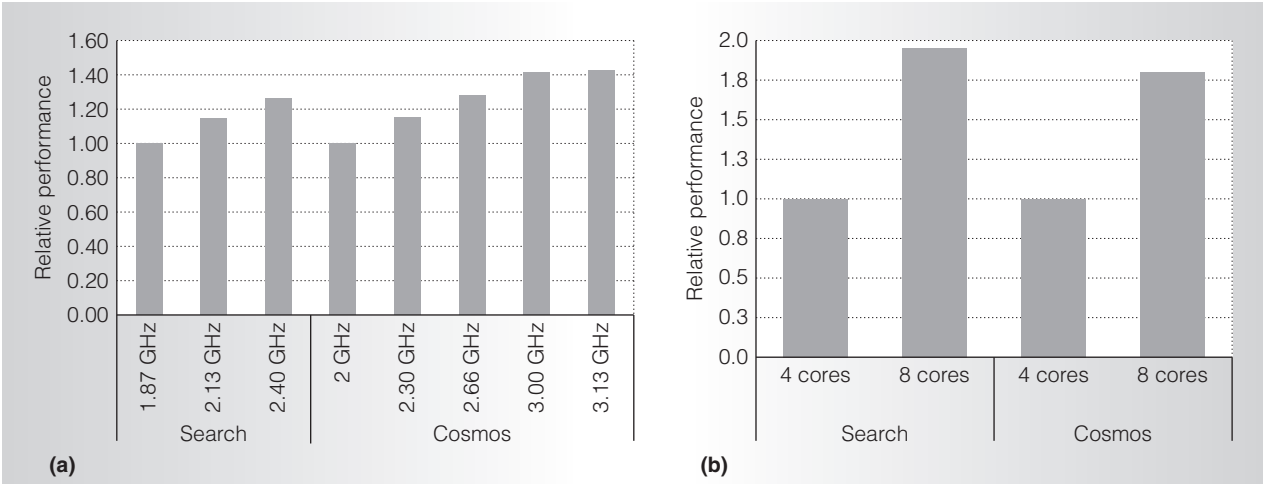


Figure 2. Performance scaling as a function of processor frequency and number of cores for Bing and Cosmos.

Hotmail places more emphasis on storage performance than Bing and Cosmos. Under stress tests (Table 1), the processors reach 67 percent utilization but performance is ultimately limited by the bandwidth utilization of the disk array holding the mailbox metadata and messages. For the data portion of the hard disks, we observed up to 71 percent utilization on a random I/O access pattern. Note that the average disk subsystem bandwidth utilization in a production environment is measured over longer time durations and hence tends to be orders of magnitude lower than peak values. For instance, the maximum observed bandwidth utilization is around 70 percent of the maximum theoretical bandwidth for Hotmail, while the average utilization is 0.91 percent. This difference is due to long periods of low user activity (such as at night). Table 4 presents the difference between average and peak utilizations. Hotmail also uses network bandwidth significantly better than the other services, but without being limited by networking. It is worth noting that even though Hotmail is storage bound during peak loads, it uses up most of its available memory capacity because Hotmail metadata is managed using an SQL database, which can leverage available memory for its caches.

We now examine how the two CPU-bound services—Bing and Cosmos—interact with processor scaling trends. Figure 2 shows that the performance, measured in terms of throughput, for both services improves with higher clock frequency at a linear factor of 90 percent. We observe flattening only at the highest clock frequencies (3 GHz). Nevertheless, clock frequency scaling is becoming increasingly difficult due to the higher power consumption and higher cost of high-frequency components. The trend is now toward scaling the number of cores per chip. Both services can benefit from this trend as they scale almost perfectly with the number of cores (linear factors of 95 and 80 percent, respectively). We also studied the impact of the size of last-level caches (multimegabyte L3 caches) and found that halving the cache sizes doesn't significantly affect either service.[8] Bing's working set extends beyond the last-level cache and memory access is required anyway, whereas Cosmos can stream data equally well using smaller caches. This behavior can also serve as an interesting input to processor architects and could help properly size the shared and dedicated caches allocated to multiple cores. Reddi et al. present additional details on frequency scaling's impact and an in-depth analysis of processor stalls with respect to Bing.[8] Thus, we observe that:

*Observation 1. Online services scale well with an increasing number of cores. Since memory bandwidth was not a bottleneck in our studies, and most online services serve multiple parallel user requests, scaling*

**Table 3. Design ratios for system balancing rules.**

| Application | MIPS/ core | Disk read MBps/ core | Disk read IOPS/ core | Instructions/ I/O byte | Memory MBps/ Core | MIPS/ disk MBps | MIPS/ memory MBps | Mbytes/ MIPS | Instructions/ disk I/O |
|---|---|---|---|---|---|---|---|---|---|
| Amdahl | 1 | | | | | 8 | | 1.0 | 50,000 |
| Hotmail | 1,059 | 0.32 | 20.88 | 3,271 | N/A | 3,271 | N/A | 1.9 | 50,734,894 |
| Cosmos | 3,698 | 0.24 | 0.84 | 15,173 | 40.9 | 15,173 | 724 | 0.2 | 4,402,872,260 |
| Bing | 1,849 | 0.17 | 1.40 | 10,643 | 145.8 | 10,643 | 101 | 1.1 | 1,323,224,579 |

*of workload throughput with an increasing number of cores should continue.*

For memory and storage systems, an emerging trend is solid-state, nonvolatile devices. Devices such as flash provide a sweet spot between the cost for storage and access latency: flash is less expensive than DRAM but orders of magnitude faster than disk drives for random access read workloads. The cost ratios for DRAM to flash to disk are roughly 100:10:1, while the latency ratios are 1:5,000:1,000,000. However, the predominant question here is how exactly can flash feature in low-cost scale-out deployments for online services.

First, it can replace high-end SAS 15K-RPM drives for the I/O-intensive portions of an application. For instance, the metadata accesses in Hotmail can be served by databases stored in solid-state devices (SSDs) to provide higher IOPS and lower power consumption for random lookups. Nevertheless, Hotmail exhibits high utilization of disk capacity. Hence, we must also consider flash's impact on the TCO for the data center. Since flash is more expensive per bit than disk, we must use a metric that quantifies performance per dollar (such as queries per second [QPS] per dollar) for the entire data center to determine if flash will significantly improve efficiency.

Flash can also act as an intermediate tier between main memory and disk. This helps latency-sensitive applications that preload data into memory from the disk, as in the case of Bing where SSDs can serve as memory extension buffers. However, flash's access latency is still much higher than that of DRAM and hence, flash cannot directly replace memory. Its introduction into high-volume, commodity server farms will require careful consideration of the service cost to handle data movement between layers to optimize performance, reliability, and TCO in a scale-out scenario. Furthermore, the services might need to explicitly manage data movement between memory, flash, and disk depending on the dataset size and access patterns. Hence, we also observe that:

*Observation 2. Flash storage will emerge as a memory extension pool for many online services as it mitigates I/O access latencies at a lower cost per bit than DRAM.*

## Server system balance for online services

We now focus on the system balance for online services using Gray and Shenoy's framework for database systems. We compute ratios similar to their rules 11, 12 and 13[1]:

- *Rule 11.* Amdahl's revised balanced system law states that a system needs 8 MIPS/Mbytes per second (MBps) I/O, but the instruction rate and I/O must be measured for the relevant workload. Sequential workloads tend to have lower CPI than random workloads.
- *Rule 12.* The memory size to instructions ratio (Mbytes/MIPS, denoted Alpha) is rising from 1 to 4 and this trend will continue.
- *Rule 13.* Random I/Os happen about once each 50,000 instructions. The instruction count is higher for sequential I/O (since more data is accessed within each sequential I/O).

Table 3 shows the ratios discussed in these three rules for the online services studied. We collected the performance data using servers that are dual-socket, quad-core machines (eight cores total) and include two to four disks for Bing and Cosmos and larger disk arrays for Hotmail. We calculated the core

**Table 4. Average and maximum resource usage ratios.**

| | | Disk MBps/core | | MIPS/disk MBps | |
|---|---|---|---|---|---|
| **Application** | **MIPS/core** | **Avg+2Sigma** | **Max** | **Avg+2Sigma** | **Max** |
| Amdahl | 1 | | | 8 | 8 |
| Hotmail | 1,059 | 0.32 | 25.22 | 3,271 | 42 |
| Cosmos | 3,698 | 0.24 | 2.73 | 15,173 | 1,357 |
| Bing | 1,849 | 0.17 | 5.73 | 10,643 | 323 |

CPI for the purposes of calculating the MIPS and memory bandwidth from the stress runs, and we believe it is a fair representation of production workloads. Since the performance counter values change with time, we use the average plus two standard deviations as a representative measure for the maximum metric for all variables considered. The quad-issue processors used in this study can execute about 10 billion instructions per second.

As column 7 in Table 3 shows, the processor instructions to disk bandwidth ratio for Bing and Cosmos is several orders of magnitude higher than the Amdahl value of 8 for database systems. This behavior is primarily due to these services' design, which has them pre-allocate their data in memory before processing it to achieve better latency and throughput than if the data were accessed from disk. This design choice is not shared with the throughput-oriented workloads studied by Gray and Shenoy, where disk latency was acceptable. Hence, the processor instructions to memory bandwidth ratio (Table 3, column 8) might be a more important metric for these online services. The MIPS to memory MBps ratio is lower, but still an order of magnitude greater than the MIPS to disk MBps ratio for databases. An obvious conclusion is that these online services are performing more processing than database systems. To some extent the ratios are affected by the fact that the application design and server configurations were optimized to match the available commodity hardware, but the difference from databases is large enough to imply that the characteristics of online services are fundamentally different from those of databases. Part of the reason that the achieved MIPS rating is high is that Bing and Cosmos use data compression on memory and disk data, respectively, causing extra processing on compression and decompression, a design choice again motivated by storage requirement growth projections.

Gray and Shenoy's note in rule 11 regarding CPI being lower for sequential workloads still applies. The Cosmos service, which performs predominantly sequential processing, has higher MIPS than Bing and Hotmail, which access data randomly. Hotmail is I/O dependent and therefore exhibits lower MIPS and a one order of magnitude lower MIPS to disk MBps ratio than Bing and Cosmos. Nevertheless, its MIPS to disk MBps ratio differs significantly from Amdahl's ratio of 8 for databases.

The system design implication then is that the correct Amdahl value for online application servers differs across workloads. For instance, according to rule 11 for databases, we would need about 1,250 MBps of disk bandwidth per core. In an online services scenario that uses commodity SATA disk drives with roughly 100 MBps sustained sequential bandwidth, this implies using about 96 disk drives for each dual-socket quad-core server. The actual number of disks used is in fact much smaller because the ratios are very different for these services, especially for those that use the main memory of thousands of servers as the primary data buffer. In a classic database system, the data set is significantly larger than the amount of memory available, hence the disk bandwidth and ratio of MIPS to disk MBps continues to be important. For example, consider the recent measurement of the TPC-E workload, which performs well on a system with 16 processors, 96 cores, and approximately 900 15K-RPM disk drives—a core-to-disk ratio of about 10 (see http://www.tpc.org/tpce/results/tpce_result_detail.asp?id=109110201). This suggests that the MIPS/MBpsI/O for conventional database workloads is still similar to previously

observed Amdahl values. You can find a detailed study of database workloads in datacenter environments elsewhere.[9]

Hence, we can state rule 11 for balanced systems for the widely variant online services type workloads as:

> *Observation 3. While the ratio of MIPS/MBpsI/O varies significantly across workloads, the ratio is at least two orders of magnitude higher for online services than for database systems. Moreover, for services that use distributed main memory structures as the primary data storage, the MBpsI/O measured should be based on memory I/O rather than disk I/O.*

The alpha ratio of 1 noted in rule 12 still holds. Gray and Shenoy predicted the ratio would rise from 1 to 4 because systems were expected to use larger memories as disk I/O continued to become more precious.[1] Indeed, disk I/O has become more precious and memory sizes have continued to grow, so much so that most of the data accessed by Bing and Cosmos is preloaded to memory for data processing. And yet, contrary to what was believed 10 years ago, the alpha ratio has not grown from 1 to 4 for online services. The key reason for this deviation is the growth of thread-level parallelism due to the increasing number of cores in processor chips. Hence, the overall available MIPS per CPU socket has increased disproportionately faster than expected. We can therefore state that:

> *Observation 4. The fast rise in available MIPS per CPU socket has effectively compensated for the rise in memory size, maintaining the alpha ratio near 1.*

Considering rule 13, we again note that because Bing and Cosmos services maintain much of their data in memory, comparing the number of instructions per disk I/O to the corresponding values for databases is not appropriate. Moreover, computing a similar ratio with respect to memory I/Os is not feasible because the main memory systems use random access devices and the concept of counting the number of I/Os or drawing a distinction between random and sequential is not relevant. Still, we report the instructions per disk I/Os for these services for completeness. Instead of the previously reported 50k instructions per I/O, the cores here can execute more than a billion instructions for every disk I/O for Bing and Cosmos. Even for Hotmail, which is disk bound, the instructions per disk I/O differ significantly from Gray and Shenoy's reported values.[1]

## Datacenter provisioning and cost considerations

More than a decade ago, single server balances were crucial for the scale-up applications that were prevalent. Given the nature of the large-scale online services of this decade, it is now important to emphasize both the overall datacenter balance and the server system balance.

### The overprovisioning dilemma

Online services have unique requirements since the workload is driven by user demand. For instance, planned and unplanned peaks can occur in Bing usage depending on major news events around the world. Hence, the datacenter designer must decide whether to provision for peak usage, allowing best-case performance for infrequent events, or for average usage, resulting in reduced performance during peak loads. Because both cost and performance are crucial, a simple provisioning guideline is unlikely to emerge and the datacenter designer must make conscious design choices to manage this dilemma. To illustrate this point, we present the MIPS/MBps ratio for the selected online services using average plus twice the standard deviation (Avg+2Sigma) and peak values (Max) in Table 4. An order of magnitude difference exists between the peak and average ratios among these services. Most notably for Bing, the peak and average usages are far apart, with a MIPS to disk MBps ratio of 10,643 at Avg+2Sigma compared to a ratio of 323 at maximum disk throughput. Note that the use of Avg+2Sigma data captures the typical peak usage and ignores rarely observed maximum values as outliers. Other studies have also reported that the CPU use of servers in data centers rarely exceeds 20 percent.[10] However, a datacenter operator must also consider the effects of the

maximum peaks that might be encountered, since such scenarios can lead to overutilization of the datacenter power budget (if all servers were to peak simultaneously), causing a service outage. In such situations, the operator must consciously decide to allocate datacenter power based on server operating points.

More extensive research is required to determine a sweet spot between the average and peak values that can act as the efficient operating point for the data center with tolerable peak performance. For instance, application throttling can help reduce performance during peak demand and server/rack-level power-capping mechanisms can counter the threat of service outage due to peak usage. Power-control policies combined with cautious overprovisioning will be crucial in that scenario. Elastic alternatives where provisioned resources can be dynamically allocated, such as using a private cloud infrastructure to cohost a large pool of services, can alleviate this problem for certain online services. Although cloud infrastructures are currently in a nascent stage, the need for dynamic provisioning felt by most online services will likely make such shared larger-scale data centers more attractive in the future.

Hence, we also observe that:

*Observation 5. Designers currently handle the large dynamic range of workload variations by considering the worst-case peak behavior observed for an online service. The future will likely show better control techniques that allow for optimal balance between server power and delivered performance, and a growth in shared cloud infrastructures that allow dynamic resource provisioning for individual services.*

## Reliability and availability using commodity data storage

Data storage designs, including both databases and online services, must address storage reliability and build in redundancy to accommodate failures. Gray and Shenoy noted that the ratio between disk capacity and disk bandwidth was increasing by 10 times per decade,[1] suggesting that mirroring will become more important than RAID5. This ratio has continued to increase over the past decade. They noted disks with 70 Gbytes capacity, 25 MBps transfer rates, and 10 ms access times.[1] Current SATA drives offer 2 Tbytes with 70 MBps bandwidth. Thus, capacity has increased by 28 times, while bandwidth has improved by 2.5 times, sustaining the expected 10 times increase in the capacity to bandwidth ratio. Disk access times have not changed much and currently range from 2 to 15 ms. It is clear that disk bandwidth continues to be more important than storage capacity, tilting the balance further in favor of mirroring over RAID5.

In our study of Microsoft's large-scale online services, including the three examined here, we found that RAID5 has completely disappeared. Although many of the array controllers in the deployed servers support RAID5, the applications did not use it due to its slower write throughput and longer disk rebuild times. Even when the application had to store a significant amount of cold data, designers often mixed the cold data with hot data to reduce the strain on I/O bandwidth rather than move it to slower RAID5 storage to save space.

Mirroring and striping, on the other hand, has become ubiquitous. It was used in several instances in the broader set of applications considered for this article, at the disk controller layer via RAID1+0 and more often at the application layer through replication. Application-layer replication methods, such as SQL server replication, protect against server hardware and software faults in addition to disk failures. The sustained increase in storage capacity has thus allowed application designers to build in increased redundancy to protect against failure modes other than those addressed by RAID alone. Furthermore, many applications use replication to increase throughput, a natural design technique that was expected to have emerged due to the slower increase in data transfer rates compared to storage capacities. Although not observed in the applications we studied, space-efficient approaches, such as RAID5 and similar techniques implemented in software, will likely remain relevant for applications storing predominantly cold data, such as long-term archives.

Online services have also begun using georeplication. Historically, companies protected against the loss of an entire enterprise

location by periodically shipping tape or disk backups; today, large online services can use near real-time georeplication. This allows, at least in theory, a hot-swap of an entire enterprise facility in case of inclement weather, war, or major infrastructure breakdown. Fundamentally, the large scale-out nature of services, which extend their deployment beyond servers in a single enterprise facility, and higher network bandwidths make this possible. Much georeplication today aims to reduce network latency for users spread across widely separated geographic locations. While sufficient redundant capacity might not be provisioned for the failure of an entire data center at this point (implying that a hot-swap might not currently work in practice), current trends suggest that such redundant capacity will likely become more feasible over time. With cloud computing infrastructures now offering georeplication (see http://www.microsoft.com/windowsazure), its use will likely become an integral part of many scale-out applications, even for smaller-scale enterprise applications that would not have found multiple facilities economically feasible when hosted individually. Georeplication can thus be seen as a potentially valuable technology transfer from massively scale-out large applications to smaller commodity online services.

We therefore observe that:

*Observation 6. The constantly increasing storage capacity to bandwidth ratio has marginalized the use of RAID5 while mirroring and replication are increasingly being used to provide data reliability for online services. The use of georeplication is likely to increase for online services, since it not only improves latency for geography-specific user content delivery but also provides fault tolerance against additional failure modes.*

### Datacenter cost balance

We also evaluate the trade-off between various datacenter platforms based on balanced servers that deliver a different absolute amount of compute power. Such platforms differ in the number of sockets, cores, and DRAM channels within a server. For each platform type, factors such as power, price, and performance play key roles in determining a given service's effectiveness. When extrapolated to large-scale datacenter deployments, we consider the ratio of scale-up (compute capacity within a server) to scale-out (compute capacity distributed across servers) that will provide the most economical outcome, while still meeting the key system balance ratios discussed earlier.

For our analysis of the Bing service, we consider four types of scale-up platforms, ranging from single-processor to eight-processor servers. We scaled-out (multiply deployed) each server type in a data center server farm. We projected performance, power, and datacenter operating costs for each scenario based on system pricing data available online from large server vendors. We used a publicly available cost model to derive the datacenter's TCO.[11] The model assumes $200 million in facility costs based on 15 megawatts of critical power with a datacenter power usage efficiency of 1.7. The model amortizes power distribution and cooling infrastructure costs over 15 years and the purchase cost of servers over three years. We sized the servers used for the analysis to satisfy the key ratios shown in Table 3, relating the memory and storage capacities and bandwidth to processor performance. Table 5 shows the performance, capital expenses, and operating expenses for these four server types, with one to eight CPU sockets. We derived the relative performance from the data used in Figure 2 for performance scaling with the number of cores. Because networking equipment costs tend to be insignificant compared to the server costs in data centers, we do not consider them here.

The above analysis shows that at this point, dual-socket platforms offer the most compelling proposition for overall throughput-per-TCO dollar. This is primarily influenced by pricing effects for CPUs targeted at different market segments and the server infrastructure (motherboard, memory, power supplies, chassis, and so on) needed to support the available compute capability. CPU chips for single- and dual-processor systems fall in the same price range, but chips for multiprocesoor machines are approximately two to four times more expensive. The primary reason for this pricing difference is that multiprocessor CPUs are much more complex to design and validate,

**Table 5. Datacenter total cost of ownership (TCO) calculations for various server types.**

| Feature | Single-processor server | Dual-processor server | Multiprocessor server | |
|---|---|---|---|---|
| | | | 4 processors | 8 processors |
| CPUs | 1 | 2 | 4 | 8 |
| Cores per server | 6 | 12 | 32 | 64 |
| Memory | 12 | 24 | 64 | 128 |
| Drives | 2 | 3 | 8 | 16 |
| CPU+Mem power (as the percentage of system power) | 42 | 55 | 62 | 62 |
| CPU+Mem price (as the percentage of system price) | 42 | 59 | 67 | 65 |
| Relative price | 1.0 | 1.5 | 7.3 | 14.8 |
| Relative power | 1.0 | 1.5 | 4.2 | 8.5 |
| Relative performance | 1.0 | 2.0 | 3.5 | 5.6 |
| Normalized throughput | 0.79 | **1.00** | 0.65 | 0.52 |
| Normalized TCO | 0.99 | **1.00** | 1.47 | 1.49 |
| Throughput/TCO (normalized to a dual processor) | 0.80 | **1.00** | 0.45 | 0.35 |

and have larger die sizes (because of the bigger caches and additional cores) and more expensive packages. Additionally, motherboards for larger systems have expandability options for several DIMM and I/O slots and hence require larger boards with multiple routing layers and appropriately sized power supplies—all of which increase complexity and system cost.

CPU and memory power dominate the system power (50 percent range for all platforms considered). In addition, the proportion of CPU price to system price significantly increases with number of sockets in the server.

The interesting observation for server scale-up is that as the compute capacity per server is doubled, the performance scales near-linearly but the power and price rise much faster. This effectively diminishes the value of adding additional compute capacity per server. For example, going from a single-processor to an eight-processor server increases performance by 5.6 times, but incurs a power penalty of 8.5 times and a price penalty of 14.8 times, making it unattractive to design scale-up systems for such scale-out friendly services.

Essentially, the TCO analysis shows that given current component pricing trends, it is most economical to scale the hardware infrastructure using a large server farm of commodity, dual-socket platforms, versus a smaller set of scale-up platforms. Although the exact configuration of a commodity server can change over time, we expect this observation to hold.

Overall, we can summarize our observations for economical systems design as:

> *Observation 7. Power and price should increase at a rate equal to or less than the rate at which performance increases as we scale up the compute capacity in each server of the scale-out system.*

A corollary of this observation is the following. In efficient server designs, the system's power and cost should be primarily influenced by the key components (CPU, memory, and storage) determining service performance and not by the remaining server infrastructure. In other word, for a service like Bing, the ratio of CPU+memory power/cost to system power/cost should tend to 1. This essentially implies that the power and cost of peripheral system infrastructure components, such as chip sets and boards, be kept minimal.

Although many of Gray and Shenoy's observations for databases hold for online workloads, significant differences exist. First, online services achieve scalable throughput using scale-out server farms, making latency a primary design concern for

the server unit. Hence, the memory system increasingly determines a server unit's balance. Second, online services shift cost and power considerations from a single server to the entire data center. Hence, balanced servers must be based on low-cost, commodity components that efficiently amortize the price and power overheads of the baseline server design.

Given the diversity and dynamic nature of user-facing services, significant work remains to understand and optimize their operation. Exploiting new solid-state storage technologies and managing the trade-offs between peak and average usage are two examples of challenges that will likely change both the services' software structure and the nature of the hardware infrastructure hosting these services. MICRO

## References


1. J. Gray and P. Shenoy, ''Rules of Thumb in Data Engineering,'' *Proc. Int'l Conf. Data Eng.* (ICDE 00), IEEE CS Press, 2000, p. 3.
2. L. Barroso and U. Hölzle, *The Datacenter as a Computer: An Introduction to the Design of Warehouse-Scale Machines,* Morgan & Claypool Publishers, 2009.
3. K. Olukotun and L. Hammond, ''The Future of Microprocessors,'' *ACM Queue,* 18 Oct. 2005, pp.26-29.
4. A. Leventhal, ''Flash Storage Memory,'' *Comm. ACM,* vol. 51, no. 7, July 2008, pp. 47-51.
5. B.C. Lee et al., ''Architecting Phase Change Memory as a Scalable DRAM Alternative,'' *Proc. Int'l Symp. Computer Architecture* (ISCA 09), ACM Press, 2009, pp. 2-13.
6. R. Chaiken et al., ''SCOPE: Easy and Efficient Parallel Processing of Massive Data Sets,'' *Proc. Very Large Databases* (VLDB 08), VLDB Endowment, vol. 1, no. 2, 2008, pp. 1265-1276.
7. D. Patterson, ''Latency Lags Bandwidth,'' *Comm. ACM,* vol. 47, no. 10, Oct. 2004.
8. V.J. Reddi et al., ''Web Search Using Mobile Cores: Quantifying and Mitigating the Price of Efficiency,'' *Proc. ACM IEEE Int'l Symp. Computer Architecture* (ISCA 10), 2010, to appear.
9. S. Kavalanekar et al., ''Measuring Database Performance in Online Services: A Trace-Based Approach,'' *Proc. TPC Technology Conf. Performance Evaluation & Benchmarking* (TPCTC 09), LNCS 5895, Springer, 2009, pp. 132-145.
10. L. Barroso and U. Hölzle, ''The Case for Energy-Proportional Computing,'' *Computer,* vol. 40, no. 12, 2007, pp. 33-37.
11. J. Hamilton, ''Datacenter TCO Model,'' http://perspectives.mvdirona.com, 2008.



**Christos Kozyrakis** is an associate professor of electrical engineering and computer science and the Willard R. and Inez Kerr Bell faculty scholar at Stanford University. His current research focuses on energy-efficient data centers, architecture and runtime environments for many-core chips, hardware and software techniques for transactional memory, and security systems. Kozyrakis has a PhD in computer science from the University of California, Berkeley. He is a senior member of IEEE and ACM.

**Aman Kansal** is a researcher in systems and networking at Microsoft Research. His research interests include energy-proportional computing, wireless and mobile systems, and sensor-actuator networks. Kansal has a PhD in electrical engineering from the University of California, Los Angeles. He is a member of IEEE and ACM.

**Sriram Sankar** is a hardware engineer in Microsoft's Online Services Division. In addition to designing server architectures for large-scale data centers, his research interests include workload characterization, energy-efficient data centers, and emerging storage architectures. Sankar has a master's degree in computer science from the University of Virginia, Charlottesville. He is a member of ACM.

**Kushagra Vaid** is a principal architect in Microsoft's Online Services Division and leads the Hardware Engineering Team for Microsoft's Datacenter Infrastructure. His current interests are in researching innovations across the entire platform stack for deploying large-scale distributed server farms, and low-power energy-efficient microprocessor designs. Vaid has an MS from the State University of New York, Binghamton. He is a member of IEEE and ACM.

Direct questions and comments about this article to Aman Kansal, 1 Microsoft Way, Redmond, WA 98052; Kansal@microsoft.com.