On Delivering Embarrassingly Distributed Cloud Services

Ken Church
Microsoft
One Microsoft Way
Redmond, WA, USA
church@microsoft.com

Albert Greenberg
Microsoft
One Microsoft Way
Redmond, WA, USA
albert@microsoft.com

James Hamilton
Microsoft
One Microsoft Way
Redmond, WA, USA
jamesrh@microsoft.com

ABSTRACT

Very large data centers are very expensive (servers, power/cooling, networking, physical plant.) Newer, geo-diverse, distributed or containerized designs offer a more economical alternative. We argue that a significant portion of cloud services are embarrassingly distributed – meaning there are high performance realizations that do not require massive internal communication among large server pools. We argue further that these embarrassingly distributed applications are a good match for realization in small distributed data center designs. We consider email delivery as an illustrative example. Geo-diversity in the design not only improves costs, scale and reliability, but also realizes advantages stemming from edge processing; in applications such as spam filtering, unwanted traffic can be blocked near the source to reduce transport costs.

Categories and Subject Descriptors

K.6.4 [System Management]: Centralization/decentralization.

General Terms

Management

Keywords

Embarrassingly Distributed, Economies of Scale, Spam, POPs (Points of Presence).

1. Introduction

Large data centers are being built today with order 10,000 servers [1], to support "cloud services" – where computational resources are consolidated in the data centers. Very large (mega) data centers are emerging with order 150,000 multi-core servers, realized, for example, as 150 containers with 1000 servers per container.[1] In total, cloud service providers are on a path to supporting up to a million servers (some providers are rumored to have already crossed this point), in tens to hundreds of locations.

Imagine a family of solutions with more or less distribution, ranging from a single POP (point of presence) to a million. This paper will explore trade-offs associated with size and geo-diversity. The trade-offs vary by application. For *embarrassingly distributed* applications, i.e. applications with relatively little need for massive server to server communications, there are substantial opportunities for geo-diversification to improve cost, scale, reliability, and performance. Many applications fall somewhere in the middle with ideal performance at more than one POP, but less than a million. We will refer to mega-datacenters as the mega model, and alternatives as the micro model.

Table 1: Options for distributing a million cores across more or less locations (POPs = Points of Presence).

POPs	Cores/POP	Hardware/POP	Co-located With/Near
1	1,000,000	1000 containers	Mega-Data Center
10	100,000	100 containers	
100	10,000	10 containers	Fiber Hotel
1,000	1,000	1 container	
10,000	100	1 rack	Central Office
100,000	10	1 mini-tower	P2P
1,000,000	1	embedded	

Large cloud service providers (Amazon, Microsoft, Yahoo, Google, etc.) enjoy economies of scale. For example, large providers enjoy a wide set of buy/build options for the wide area network to support internal and external data transport to their data centers, and can create and manage dedicated networks, or buy network connectivity arguably at costs near those incurred by large network service providers. In the regional or metro area (e.g., pipes from data centers to the wide area network) and in peering (e.g., to large broadband service providers), these large cloud service providers have less choice and may incur higher costs. Nevertheless, by buying numerous and/or large pipes and delivering large volumes of traffic, the cloud service providers can obtain significant discounts for data transport. Savings in computational and networking resources can in turn be passed on to creators of cloud service applications, owned and operated by the cloud service provider or by third parties.

One might conclude that economies of scale favor mega-data centers, but it is not that simple. By analogy, large firms such as Walmart, can expect favorable terms primarily because they are large. Walmart can expect the same favorable term no matter how they configure their POPs (stores). Economies of scale depend on total sales, not sales per POP. In general, economies of scale depend on the size of the market, not mega vs. micro.

Large data centers are analogous to large conferences. A small (low budget) workshop can be held in a spare room in many universities, but costs escalate rapidly for larger meetings that require hotels and convention centers. There are thousands of places where the current infrastructure can accommodate a workshop or

Hotnets, Oct 2008, Calgary, CA.

[1] http://perspectives.mvdirona.com/2008/04/02/FirstContainerizedDataCenterAnnouncement.aspx

two, but there is no place where the current infrastructure could handle the Olympics without a significant capital investment. Meetings encounter diseconomies of scale when they outgrow the capabilities of off-the-shelf venues.

So too, costs escalate for large mega-data centers. For example, if a mega-data center consumes 20MW of power at peak from a given power grid, that grid may be unable or unwilling to sell another 20MW to the same data center operator. In general, the infrastructure for a new mega data center (building, power, and networking) calls for building/lighting up significant new components. Yet, if the data centers are smaller (under the micro moel), there is increased opportunity to exploit the overbuild in what is already there in the current power grid and networking fabric. There are thousands of places where the current infrastructure could handle the load for a container sized data center, but there is no place where the current infrastructure can handle a thousand containers without a significant capital investment. Data centers encounter various diseconomies of scale when they become so large that they require significant investment in infrastructure.

It is risky and expensive to put all our eggs in one basket. Paraphrasing Mark Twain (*The Tragedy of Pudd'nhead Wilson*), if all our eggs are in one basket, then we must watch that basket carfully. In the mega-data center this means a very high degree of redundancy at many levels – for example in power delivery and provisioning [1]. For example, as we cannot afford to lose the entire site owing to power failure or network access failure, the mega data center may incur large costs in batteries, generators, diesel fuel, and in protected networking designs (e.g., over provisioned multiple 10 GE uplinks and/or SONET ring connectivity to the WAN).

Many embarrassingly distributed applications could be designed at the application layer to survive an outage in a single location. Geo-diversity can be cheaper and more reliable than batteries and generators. The more geo-diversity the better (at least up to a point); *N*+1 redundancy becomes more attractive for large *N*. Geo-diversity not only protects against short term risks (such as blackouts), but also longer term risks such as a supplier cornering a local market in some critical resource (network, power, etc.). Unfortunately, in practice, inefficiencies of monopoly pricing can dominate other considerations. With small containerized data centers, it is more feasible to adapt and provision around such problems (or leverage the capability to do so in negotiations), if the need should arise.

On the other hand, there are limits to geo-diversification. In particular, it is much easier to manage a small set of reliable sites. There is little point to provisioning equipment in so many places that supply chain management and auditing become overwhelming problems. It may be hard to run a distributed system without on site workforce with timely physical access to the machine rooms. (Yet, new containerized designs have promise to dramatically mitigate the need for timely physical access[1].)

Though there has been some degree of reporting[1-14] on the nature of large and small data centers, much remains proprietary, and there has been little discussion or questioning of basic assumptions and design choices. In this paper, we take up this inquiry. In Section 2, to understand the magnitude of the costs entailed in mega-data center physical infrastructure, we compare their purpose built design with a gedanken alternative where the servers are distributed among order 1000 condominiums. The results suggest smaller footprint data centers are well worth pursuing. In Section 3, we consider networking issues and designs for mega and micro data centers, where the micro data centers are order 1K to 10K servers. In Section 4, we ask whether there are large cloud service applications that are well suited to micro data center footprints, specifically examining solutions that can be realized in an "embarrassingly distributed" fashion, and look at email in some depth. In Section 5, we contrast mega and micro data centers taking a more tempered view than in Section 2. We conclude in Section 6.

2. Power and Diseconomies of Scale

How do machine room costs scale with size? In a recent blog,[2] we compared infrastructure costs for a large data center with a farm of 1125 condominiums and found the condos to be cheaper. Condos might be pushing the limits of credulity a bit but whenever we see a crazy idea even within a factor of two of current practice, something is interesting, warranting further investigation.

A new 13.5 mega-watt data center costs over $200M before the upwards of 50,000 servers that fill the data center are purchased. Even if the servers are built out of commodity parts, the data centers themselves are not. The community is considering therefore moving to modular data centers. Indeed, Microsoft is deploying a modular design in Chicago[3]. Modular designs take some of the power and mechanical system design from an upfront investment with 15 year life to a design that comes with each module and is on a three year or less amortization cycle and this helps increase the speed of innovation.

Modular data centers help but they still require central power, mechanical systems, and networking systems. These systems remain expensive, non-commodity components. How can we move the entire datacenter to commodity components? Consider a radical alternative: rather than design and develop massive data centers with 15 year lifetimes, let's incrementally purchase condos (just-in-time) and place a small number of systems in each. Radical to be sure, but condos are a commodity and, if this mechanism really was cheaper, it would be a wake-up call to reexamine current industry-wide costs and what's driving them.

See Table 2 for the back of the envelope comparison showing that the condos are cheaper in both capital and expense. Both configurations are designed for 54K servers and 13.5MWs. The data center costs over $200M, considerably more than 1125 condos at $100K each. As for expense, the data center can expect favorable terms for power (66% discount over standard power rates). Deals this good are getting harder to negotiate but they still do exist. The condos don't get the discount, and so they pay more for power: $10.6M/year >> $3.5M/year. Even with the deal, the data center is behind because it isn't worth $100M in capital to save $7M/year in expense. But to avoid comparing capital with expense, we simplified the discussion by renting the condos for $8.1M/year, more than the power discount. Thus, condos are not only cheaper in terms of capital, but also in terms of expense.

In addition to saving capital and expense, condos offer the option to buy/sell just-in-time. The power bill depends more on average usage than worst-case peak forecast. These options are valuable under a number of not-implausible scenarios:

[2] http://perspectives.mvdirona.com/2008/04/06/DiseconomiesOfScale.aspx

Table 2: Condos are cheaper than data center in both capital and expense.

		Large Tier II+ Data Center	Condo Farm (1125 Condos)
Specs	Servers	54k	54k (= 48 servers/condo × 1125 Condos)
	Power (Peak)	13.5 MW (= 250 Watts/server × 54k servers)	13.5MW (= 250 Watts/server × 54k servers = 12 KW/condo × 1125 Condos)
Capital	Building	over $200M	$112.5M (= $100k/condo × 1125 Condos)
Annual Expense	Power	$3.5M/year (= $0.03 per kw/h ×24×365 hours/year ×13.5MW)	$10.6M/year (= $0.09 per kw/h×24×365 hours/year × 13.5MW)
Annual Income	Rental Income	None	$8.1M/year (= $1000/condo/month × 12 months/year × 1125 Condos − $200/condo/month condo fees. We conservatively assume 80% occupancy)

1. Long-Term demand is far from flat and certain; demand will probably increase, but anything could happen over 15 years.
2. Short-Term demand is far from flat and certain; power usage depends on many factors including time of day, day of week, seasonality, economic booms and busts. In all data centers we've looked into, average power consumption is well below worst-case peak forecast.

How could condos compete or even approach the cost of a purpose built facility built where land is cheap and power is cheaper? One factor is that condos are built in large numbers and are effectively "commodity parts." Another factor is that most data centers are over-engineered. They include redundancy such as uninterruptable power supplies that the condo solution doesn't include. The condo solution gets it's redundancy via many micro-data centers and being able to endure failures across the fabric. When some of the non-redundantly powered micro-centers are down, the others carry the load. N+1 redundancy is particularly attractive for embarrassingly distributed apps (Section 4).

It is interesting to compare wholesale power with retail power. When we buy power in bulk for a data center, it is delivered by the utility in high voltage form. These high voltage sources (usually in the 10 to 20 thousand volt range) need to be stepped down to lower working voltages which brings efficiency losses, distributed throughout the data center which again brings energy losses, and eventually delivered to the critical load at the working voltage (240VAC is common in North America with some devices using 120VAC). The power distribution system represents approximately 40% of total cost of the data center. Included in that number are the backup generators, step-down transformers, power distribution units, and Uninterruptable Power Supplies (UPS's). Ignore the UPS and generators since we're comparing non-redundant power, and two interesting factors jump out:

1. Cost of the power distribution system ignoring power redundancy is 10 to 20% of the cost of the data center.
2. Power losses through distribution run 10 to 12% of the power brought into the center.

It is somewhat ironic in that a single family dwelling gets two-phase 120VAC (240VAC between the phases or 120VAC between either phase and ground) delivered directly to the home. All the power losses experienced through step down transformers (usually in the 92 to 96% efficiency range) and all the power lost through distribution (dependent on the size and length of the conductor) is paid for by the power company. But when we buy power in quantity, the power company delivers high voltage lines to the property and we need to pay for expensive step down transformers as well as power distribution losses. Ironically, if we buy less power, then the infrastructure comes for free, but if we buy more then we pay more.

The explanation for these discrepancies may come down to market segmentation. Just as businesses pay more for telephone service and travel, they also pay more for power. An alternative explanation involves a scarce resource, capital budgets for new projects. Small requests for additional loads from the grid can often be granted without tapping into the scarce resource. Large requests would be easier to grant if they could be unbundled into smaller requests, and so the loads could be distributed to wherever there happens to be spare capacity. Unbundling requires flexibility in many places including the applications layer (embarrassingly distributed apps), as well as networking.

3. Networking

In addition to power, networking issues also need to be considered when choosing between mega data centers (DCs) and an alternative which we have been calling the micro model:

- **Mega mode**l: large DCs (e.g., 100,000 – 1,000,000 servers).
- **Micro model**: small DCs (e.g., 1000 – 10,000 servers).

The mega model is typical of the networks of some of today's large cloud service providers, and is assumed to be engineered to have the potential to support a plethora of services and business models (internal as well as hosted computations and services, cross service communication, remote storage, search, instant messaging, etc.) These applications need not be geo-diverse, and in practice many of today's applications still are not. Thus, the reliability of the application depends on the reliability of the mega-data center. In the micro model, we consider applications engineered for N+1 redundancy at micro data center level, which then (if large server pools are required) must be geo-diverse. In both models, we must support *on-net* traffic between data centers, and *off-net* traffic to the Internet. We focus the discussion here on off-net traffic; considerations of on-net traffic lead to similar conclusions. While geo-diversity can be difficult to achieve – espe-

cially for legacy applications – geo-diversity has advantages, and the trend is increasingly for geo-diverse services. For embarrassingly distributed services (Section 4), geo-diversity is relatively straightforward.

Under the mega model, a natural and economical design has the cloud service provider creating or leasing facilities for a dedicated global backbone or Wide Area Network (WAN). Off-net flows traverse: (1) mega data center to WAN via metro (or regional) links, (2) WAN to peering sites near the end points of flows, (3) ISPs and enterprises peered to the WAN. The rationale is as follows. Large cloud service providers enjoy a wide set of buy/build options across networking layers 1, 2, 3 in creating wide area and metro networks.[3] Via a global WAN, the cloud service provider can "cold potato" route to a very large set of peers, and thereby reap several benefits: (1) settlement free peering with a very large number of tier 2 ISPs (ISPs who must typically buy transit from larger ISPs), (2) lower cost settlement with tier 1 ISPs as high traffic volumes are delivered near the destination, and (3) (importantly) a high degree of unilateral control of performance and reliability of transport. In the metro segment, there will be typically some form of overbuild (SONET ring or, more likely, multiple diverse 10GE links from the data center) of capacity to protect against site loss. A strong SLA can be supported for different services, as the cloud service provider has control end to end, supporting, for example, performance assurances for database sync, and for virtualized network and computational services sold to customers who write third party applications against the platform.

In the micro model, a vastly simpler and less expensive design is natural and economical, and is typical of many content distribution networks today. First, as the computational resources are smaller with the micro data center, the networking resources are accordingly smaller and simpler, with commodity realizations possible. To provide a few 10 GE uplinks for the support of 1K to 10K servers commodity switches and routers can be used, with costs now in the $10K range[18]. In the mega data center, these network elements are needed, as well as much larger routers in the tree of traffic aggregation, with costs closer to $1M. In the micro model, the cloud service provider buys links from micro data center to network services providers, and the Internet is used for transit. Off-net traffic traverses metro links from data center to the network service providers, which deliver the traffic to the end users on the Internet, typically across multiple autonomous systems. As we assume N+1 redundancy at micro data center level, there is little or no need for network access redundancy, which (coupled with volume discounts that come from buying many tail circuits, and with the huge array of options for site selection for micro data centers) in practice should easily compensate for the increase in fiber miles needed to reach a larger number of data centers. In buying transit from network providers, all the costs of the mega model (metro, wide area, peering) are bundled into the access link costs. Though wide area networking margins are considered thin and are becoming thinner, the cost of creating dedicated capacity (mega model) rather than using already created shared capacity is still higher. That said, in the micro model, the cloud service provider has ceded control of quality to its Internet access providers, and so cannot support (or even fully monitor) SLAs on flows that cross out multiple provider networks, as the bulk of the traffic will do. However, by artfully exploiting the diversity in choice of network providers and using performance sensitive global load balancing techniques, performance may not appreciably suffer. Moreover, by exploiting geo-diversity in design, there may be attendant gains in reducing latency.

[3] While less true in the metro area, a user of large wide area networking resources can fold in metro resources into the solution.

4. Applications

By "embarrassingly distributed" applications, we mean applications whose implementations do not require intense communications within large server pools. Examples include applications:

- Currently deployed with a distributed implementation: voice mail, telephony (Skype), P2P file sharing (Napster), multicast, eBay, online games (Xbox Live),[4] grid computing;
- Obvious candidates for a distributed implementation: spam filtering & email (Hotmail), backup, grep (simple but common forms of searching through a large corpus)
- Less obvious candidates: map reduce computations (in the most general case), sort (in the most general case), social networking (Facebook).

For some applications, geo-diversity not only improves cost, scale, reliability, but also effectiveness. Consider spam filtering, which is analogous to call gapping in telephony[17]. Blocking unwanted/unsuccessful traffic near the source saves transport costs. When telephone switching systems are confronted with more calls than they can complete (because of a natural disaster such as an earthquake at the destination or for some other reason such as "American Idol" or a denial of service attack), call gapping blocks the traffic in central offices, points of presence for relatively small groups of customers (approximately 10,000), which are likely to be near the sources of the unsuccessful traffic. Spam filtering should be similar. Blocking spam and other unwanted traffic mechanisms near the source is technically feasible and efficient[14] and saves transport. Accordingly, many cleansing applications, such as spam assassin[15], can operate on both mail servers and on end user email applications.

Email is also analogous to voice mail. Voice mail has been deployed both in the core and at the edge. Customers can buy an answering machine from (for example) Staples and run the service in their home at the edge, or they can sign up with (for example) Verizon for voice mail and the telephone company will run the service for them in the core. Edge solutions tend to be cheaper. Phone companies charge a monthly recurring charge for the service that is comparable to the one-time charge for the hardware to run the service at home. Moving the voice mail application to the edge typically pays for itself in a couple of months. Similar comments hold for many embarrassingly distributed applications. Data center machine rooms are expensive, as seen in Section 2. Monthly rents are comparable to hardware replacement costs.

Let us now consider the email application in more depth.

4.1 Email on the Edge

Microsoft's Windows Live Hotmail has a large and geo-diverse user base, and provides an illustrative example. Traffic volumes are large and volatile (8x more traffic on some days than others), largely because of spam. Hotmail blocks 3.4B spam messages per day. Spam (unwanted) to ham (wanted) ratios rarely fall below 70% and can spike over 94%, especially after a virus outbreak.

[4] Online games actually use a hybrid solution. During the game, most of the computation is performed at the edge on the players' computers, but there is a physical cloud for some tasks such as match making and out-of-bandwidth signaling.

Adversaries use viruses to acquire zombies (bot farms). A few days after an outbreak, zombies are sold to spammers, and email traffic peaks soon thereafter.

Hotmail can be generally decomposed into four activities, all of which are embarrassingly distributed:

1) Incoming Email Anti-malware and Routing
2) Email Storage Management
3) Users and Administrator Service
4) Outgoing Email Service

Incoming Email Anti-malware and Routing: Mail is delivered to the service via SMTP. Load balancers distribute incoming connections to available servers. Edge blocks are applied to reject unwanted connections via IP black lists and anti-spam/virus mechanisms. Additional filters are applied after a connection is established to address Directory Harvest Attacks (en.wikipedia.org/wiki/E-mail_address_harvesting) and open relays (en.wikipedia.org/wiki/Open_mail_relay).

Email Storage Management: The store has to meet requirements on reliability, availability, throughput and latency. It is common practice to build the store on top of a file system (although propriety blob storage solutions are also popular). Header information and other metadata are maintained in a structured store for speed.

Users and Administrator Service: Requests come into the service from users in a variety of protocols including POP, IMAP, DAV, Deltasync, HTTP (web front ends). These requests are typically sent to pools of protocol servers. The protocol servers make authentication requests for each user to a separate authentication service: looking up the user's email address and finding the appropriate email storage server for that user, making internal transfer requests from the storage server, and returning the results in the appropriate format. Administrative requests are handled in the same way although with different permission and scope from normal users.

Outgoing Email Service: The outgoing email service accepts e-mail send requests from authenticated users. These messages are typically run through anti-malware facilities to avoid damaging the overall service reputation by distributing malware. And then the messages are routed as appropriate internally or externally.

4.2 Implementing Email near the Edge

Although Windows Live Hotmail and other email services are currently implemented as central in-the-core services with relatively few (10) data centers, more POPs could improve response time and service quality by distributing work geographically. Some mail services (such as Yahoo) migrate mailboxes as users move (or travel). Reliability can be achieved by trickling data from a primary server to a secondary server in another location, with small impact on overall cost. Order 100 POPs are sufficient to address latencies due to the speed of light, though more POPs enhance features such as blocking unwanted traffic near the source.

Microsoft Exchange Hosted Services[13] provides an example in the marketplace of hosted email anti-malware services.

5. Mega vs. Micro

Applications in the data center fall roughly into two classes: large analysis and service. Many large analysis applications are best run centrally in mega data centers. Mega data centers may also offer advantages in tax savings, site location and workforce centralization. Interactive applications are best run near users. Interactive and embarrassingly distributed applications can be delivered with better QoS (e.g., smaller TCP round trip times, and greater independence of physical failure modes) via micro data centers. It can also be cheaper to deliver such services via micro data centers.

With capital investment for a mega data center that run $200M to $500M before adding servers, the last point is important. Major components of the mega data center infrastructure are not commodity parts; e.g., 115 KVA to 13.2 KVA and 13.2 KVA to 408 VA transformers. Moreover, mega data centers are constructed with high levels of redundancy within and across layers[1]. In particular, power redundancy (UPS, resilient generator designs, seas of batteries, backup cooling facilities, and storage for 100K gallons of diesel) consumes at least 20% of the total infrastructure spend. In contrast, micro data center designs use commodity parts. With resilience in the network of micro data centers, there is little or no spend on generators, diesel, redundant cooling; the cost of many levels of redundancy disappears. As a result, the unit capital cost of resources in the mega data center exceeds that of the micro data center. To capture this in a simple model, we assume that resources have unit cost in the micro data center, but the same resources cost U in the mega data center, where $U \geq 1$.

While varying by application, networking and power consumption needs scale with the workload. If we split workload from a single large center into K smaller centers, then some efficiency may be lost. A compensatory measure then is to use load balancing (e.g., via DNS or HTTP level resolution and redirection). For example, an overloaded micro data center might redirect load to another micro data center (chosen in a random, or load and proximity sensitive manner). This can reclaim most of the efficiencies lost. New traffic is introduced between micro data centers can be mitigated by measures discussed earlier: edge filtering, application or network layer DDoS scrubbing (see e.g. [22]), time shifting of traffic needed to assure resilience and optimization of transport costs between fixed sites (e.g., locating near fiber hotels in metro areas). To first order, as capital costs of the data center dominate operational costs of networking and power[1], and taking into account available measures, we do not see the uplift in networking costs from internal transfers as appreciable.

To get some understanding of the worst case for networking and power capital costs, let's consider a simple model for the case of no cross data center load balancing. A parsimonious model of Internet workload[19][20], ideally suited to scenarios such as data centers that multiplex large numbers of flows, models workload as $m_t + \sqrt{a\, m_t}\, w_t$ where m_t is the time varying mean traffic rate, w_t is a stationary stochastic process with zero mean and unit variance (e.g., Fractional Gaussian Noise), and the single parameter a captures the "peakedness" or bustiness of the load. For example, this model can capture the phenomenon seen for an email provider in Figure 1. (Peakedness is sensitive to large workload spikes that are not filtered out[19] – though well run services must ultimately manage these by graceful degradation and admission control[21], with some workload turned away (spikes crossing the capacity line in Figure 1.)) If the workload is decomposed into K individual streams, with constant parameters m_i, a_i, and with independent realizations of a common Gaussian process, the model continues to hold with $m = \sum_1^K m_i$, and peakedness $a = 1/m \sum_1^K m_i\, a_i$, the weighted sum.

A service provider needs to design networking and power to accommodate most peaks. Assuming uniformity, independent

Gaussian behavior, and focusing on loads m during busy hours, the resource required for the mega center can be estimated as $m + n\sqrt{am}$, where the new parameter n captures the SLA. (Setting $n = 2$ corresponds to planning enough capacity to accommodate workload up to the 97.5th percentile.) As the unit cost of resources in the mega data centers is U, the total resource cost is then $U[m + n\sqrt{am}]$. Similarly, the total resource cost for the micro data center is K[m/K+$n\sqrt{am/K}$]. Thus, the spend to support a mega data center beyond that needed to support K micro data centers without load balancing comes to $m(U-1) - n\sqrt{ma}(\sqrt{K} - U)$. For large resource demands m, the result hinges on the unit cost penalty U for the mega data center. If U is even slightly larger than 1, then for large m the first term dominates and mega data center costs more. If unit costs are identical ($U = 1$), then in the case of no load balancing, the micro data centers cost more -- though the increment grows with $\sqrt{m}$ and so is a vanishing fraction of the total cost, which grows with m. Specifically, the increment grows with a workload peakedness term $\sqrt{a}$, a fragmentation term $\sqrt{K} - 1$, and a term n reflecting the strength of the SLA.

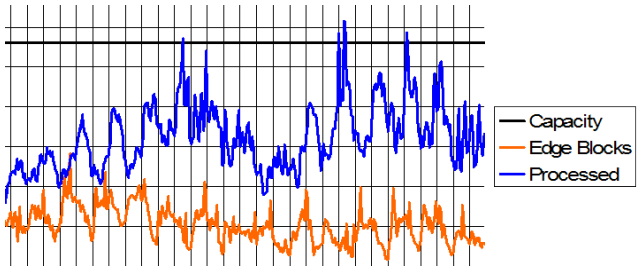

Figure 1. Processed (upper curve) and blocked (lower curve) traffic to an email provider (under spam attack).

6. Conclusions

Cloud service providers are on a path to supporting up to a million servers. Should we build a few mega datacenters under the mega model, or lots of smaller datacenters under the micro model? When applicable, the micro model is simpler and less expensive, both in terms of power (section 2) and networking (section 3); geo-diversity and N+1 redundancy eliminate complicated and expensive protection mechanisms: batteries, generators, and redundant access and transit networks. The micro model is not appropriate for all applications, but it is especially attractive for embarrassingly distributed applications, as well as applications that use small pools of servers (less than 10,000). Section 3 mentioned a number of examples, and described email in some detail. For spam filtering, geo-diversity not only simplifies the design, but the extra points of presence can block unwanted traffic near the source, a feature that would not have been possible under the mega model. Putting it all together, the micro model offers a design point with attractive performance, reliability, scale and cost. Given how much the industry is currently investing in the mega model, the industry would do well to consider the micro alternative.

REFERENCES

[1] Hamilton, J. "An Architecture for Modular Datacenters," *3rd Biennial Conference on Innovative Data Systems Research (CIDR)*, 2007.

[2] Weiss, A., "Computing in the Clouds," *ACM Networker*, Vol. 11, Issue 4, pp 18-25, December 2007.

[3] Manos, M., Data Center World, 2008, (see Mike Rath: http://datacenterlinks.blogspot.com/2008/04/miichael-manos-keynote-at-data-center.html)

[4] Arnold, S., "The Google Legacy: How Google's Internet Search is Transforming Application Software," *Infonetics*, 2005.

[5] *Caterpillar Rental Power Spec Sheets*, http://www.cat.com/cda/layout?m=43125&x=7, 2006.

[6] Cringley, R, *Google-Mart: Sam Walton Taught Google More About How to Dominate the Internet than Microsoft Ever Did*, http://www.pbs.org/cringely/pulpit/2005/pulpit_20051117_000873.html, Nov. 2005.

[7] Cummins. *Cummins Rental Power Generation*, http://www.cumminspower.com/na/services/rental/, 2006.

[8] Hoelzle, U., *The Google Linux Cluster*, http://www.uwtv.org/programs/displayevent.asp?rid=1680, Sept., 2002.

[9] IDC, *Worldwide and U.S. Software as a Service 2004-2008 Forecast and Analysis*, IDC #31016, Mar., 2004.

[10] Interport Maintenance Co, Inc., *Container Sales Pricing*, http://www.iport.com/sales_pricing.html, 2006.

[11] Kahle, B., *Large Scale Data Repository: Petabox*, http://www.archive.org/web/petabox.php, Jun., 2004.

[12] Menon, J., *IBM Storage Systems*, http://www.almaden.ibm.com/almaden/talks/cerntalksumm.pdf, IBM, 2001.

[13] Microsoft, *Microsoft Exchange Hosted Services*, http://www.microsoft.com/exchange/services/default.mspx, 2006.

[14] Microsoft, *Microsoft Compute Cluster Server 2003*, http://www.microsoft.com/windowsserver2003/ccs/default.mspx, 2006.

[15] Spam Assassin, http://spamassassin.apache.org/.

[16] Nortel Networks, *ISO 668 Series 1 Steel Freight Containers*, http://www.nortel.com, 2006.

[17] *Engineering and Operations in the Bell System*, AT&T Bell Laboratories, 1977.

[18] Greenberg, A.; Maltz, D.; Patel, P.; Sengupta, S.; Lahiri, P., "Towards a Next Generation Data Center Architecture: Scalability and Commodization," Workshop on Programmable Routers for Extensible Services of Tomorrow (*Presto*), 2008.

[19] M. Roughan and A. Greenberg and C. Kalmanek and M. Rumsewicz and J. Yates and Y. Zhang, "Experience in measuring Internet backbone traffic variability: Models, metrics, measurements and meaning," Proc. International Teletraffic Congress (*ITC-18*), 2003.

[20] I. Norros, "A storage model with self-similar input," Queueing Systems, 6:387--396, 1994.

[21] Hamilton, J., "On Designing and Deploying Internet-Scale Services," USENIX Large Installation Systems Administration Conference (*LISA*), 2007.

[22] Cisco Guard, http://www.cisco.com.