The Cost of a Cloud: Research Problems in Data Center Networks

Albert Greenberg, James Hamilton, David A. Maltz, Parveen Patel
Microsoft Research, Redmond, WA, USA

This article is an editorial note submitted to CCR. It has NOT been peer reviewed.
The author takes full responsibility for this article's technical content. Comments can be posted through CCR Online.

Abstract

The data centers used to create cloud services represent a significant investment in capital outlay and ongoing costs. Accordingly, we first examine the costs of cloud service data centers today. The cost breakdown reveals the importance of optimizing work completed per dollar invested. Unfortunately, the resources inside the data centers often operate at low utilization due to resource stranding and fragmentation. To attack this first problem, we propose (1) increasing network agility, and (2) providing appropriate incentives to shape resource consumption. Second, we note that cloud service providers are building out geo-distributed networks of data centers. Geo-diversity lowers latency to users and increases reliability in the presence of an outage taking out an entire site. However, without appropriate design and management, these geo-diverse data center networks can raise the cost of providing service. Moreover, leveraging geo-diversity requires services be designed to benefit from it. To attack this problem, we propose (1) joint optimization of network and data center resources, and (2) new systems and mechanisms for geo-distributing state.

Categories and Subject Descriptors: C.2.1 Network Architecture

General Terms: Design, Economics

Keywords: Cloud-service data centers, costs, network challenges

1. INTRODUCTION

In recent years, large investments have been made in massive data centers supporting cloud services, by companies such as eBay, Facebook, Google, Microsoft, and Yahoo!. In this paper, we attempt to demystify the structure of these data centers, and to identify areas of opportunity for R&D impact in data center networks and systems. We start our investigation with the question:

Where does the cost go in today's cloud service data centers?

To quantify data center costs, we consider a data center housing on the order of 50,000 servers that would be built based on currently well-understood techniques, using good quality, highly available equipment. Table 1 provides a rough guide to associated costs. Costs are amortized, i.e., one time purchases are amortized over reasonable lifetimes, assuming a 5% cost of money. By amortizing, we obtain a common cost run rate metric that we can apply to both one time purchases (e.g., for servers) and ongoing expenses (e.g., for power). We discuss each row in detail in Section 2.

Details may vary somewhat by site or by moment in time, but these are the major costs. While networking is not the largest cost category, this paper will argue that networking and systems innovation is the key to reducing costs and getting the most out of each dollar invested.

Amortized Cost	Component	Sub-Components
∼45%	Servers	CPU, memory, storage systems
∼25%	Infrastructure	Power distribution and cooling
∼15%	Power draw	Electrical utility costs
∼15%	Network	Links, transit, equipment

Table 1: Guide to where costs go in the data center.

1.1 Cloud Service Data Centers are Different

It is natural to ask why existing solutions for the enterprise data center do not work for cloud service data centers.

First and foremost, the leading cost in the enterprise is operational staff. In the data center, such costs are so small (under 5% due to automation), that we safely omit them from Table 1. In a well-run enterprise, a typical ratio of IT staff members to servers is 1:100. Automation is partial [25], and human error is the cause of a large fraction of performance impacting problems [21]. In cloud service data centers, automation is a mandatory requirement of scale, and it is accordingly a foundational principle of design [20]. In a well run data center, a typical ratio of staff members to servers is 1:1000. Automated, recovery-oriented computing techniques cope successfully with the vast majority of problems that arise [20, 12].

There are additional differences between the enterprise and the cloud service data center environments including:

Large economies of scale. The size of cloud scale data centers (some now approaching 100,000 severs) presents an opportunity to leverage economies of scale not present in the enterprise data centers, though the up front costs are high.

Scale Out. Enterprises often optimize for physical space and number of devices, consolidating workload onto a small number of high-price "scale-up" hardware devices and servers. Cloud service data centers "scale-out" — distributing workload over large numbers of low cost servers and hardware.

That said, enterprises are also moving toward the cloud. Thus, we expect innovation in cloud service data centers to benefit the enterprise, through outsourcing of computing and storage to cloud service providers [1, 8, 3], and/or adapting and scaling down technologies and business models from cloud service providers.

1.2 Types of Cloud Service Data Centers

Many cloud service data centers today may be termed *mega data centers*, having on the order of tens of thousands or more servers drawing tens of Mega-Watts of power at peak. Massive data analysis applications (e.g., computing the web search index) are a natural fit for a mega data center, where some problems require huge amounts of fast RAM, others require massive numbers of CPU cycles, and still others require massive disk I/O bandwidth. These problems typically call for extensive communication between servers, so the speed of the computation would drop as the propagation delay between servers increases. Further, the dollar cost of communication would go up if the servers were spread out across multiple data centers separated by long distance links, as the market price for these far exceeds the cost of intra-building links. Cloud service applications often build on one another. Having large numbers of servers in the same location eases systems design and lowers the cost of efficiently supporting applications with multiple dependencies and associated communication needs.

An area of rapid innovation in the industry is the design and deployment of *micro data centers*, having on the order of thousands of servers drawing power peaking in the 100s of kilowatts. Highly interactive applications (e.g., query/response applications, or office productivity applications [5, 4]) are a natural fit for geodiverse micro data centers placed close to major population centers, as this will minimize the speed-of-light latency and network transit costs to users. Today, micro data centers are used primarily as nodes in content distribution networks and other "embarrassingly distributed" applications, such as email [13]. However, as described in Section 5, improvements in systems software would enable micro data centers to support wider classes of applications.

2. COST BREAKDOWN

In this Section, we go through the costs of the data center described in Table 1, row by row.

2.1 Server Cost

As shown in Table 1, the greatest data center costs go to servers. For example, assuming 50,000 servers, a relatively aggressive price of $3000 per server, a 5% cost of money, and a 3 year amortization, the amortized cost of servers comes to $52.5 million dollars per year. With prices this high, achieving high utilization, i.e. useful work accomplished per dollar invested, is an important goal. Unfortunately, utilization in the data center can turn out to be remarkably low; e.g., 10%. There are some structural reasons for this:

Uneven Application fit: A server integrates CPU, memory, network and (often) storage components. It is often the case that the application fit in the server does not fully utilize one or more of these components.

Uncertainty in demand forecasts: Cloud service demands can spike quickly, especially for new services, far beyond what conventional (say 95th percentile-based) forecasts would predict.

Long provisioning time scales: Purchases, whether for upgrades or new builds, tend to be large, with components bought in bulk. Infrastructure is typically meant to last very long time periods; e.g., fifteen years. Servers are meant to last as long as 3-5 years, with increments ordered quarterly or yearly.

Risk Management: If successful, a service creator might reason, demand could ramp up beyond the capacity of the resources allocated to the service (and demand, as noted, can be hard to forecast). Inability to meet demand brings failure just when success is at hand. Given long provisioning time scales, the size of the investment, the uncertainty in demand, and the negative consequences of failure, conservatism leading to over-provisioning is a natural mechanism for risk management.

Hoarding: It is easy to get buy in from a service team for provisioning new resources, and less easy for returning them, given the factors already discussed. Inefficiencies of this type multiply across service instances.

Virtualization short-falls: Ideally, all resources (compute, storage, and networking) would be pooled, with services dynamically drawing from the pools to meet demand. Virtualization techniques have succeeded in enabling processes to be moved between machines, but constraints in the data center network continue to create barriers that prevent agility (e.g., VLANs, ACLs, broadcast domains, Load Balancers, and service-specific network engineering).

2.1.1 *Reducing These Costs*

How can data center networks and systems help to raise utilization, and solve the problems listed above? A key element of the solution is agility: the ability to dynamically grow and shrink resources to meet demand and to draw those resources from the most optimal location. Today, the network stands as a barrier to agility and increases the fragmentation of resources that leads to low server utilization. Section 3 describes the problem and the properties we seek in its solution.

2.2 Infrastructure Cost

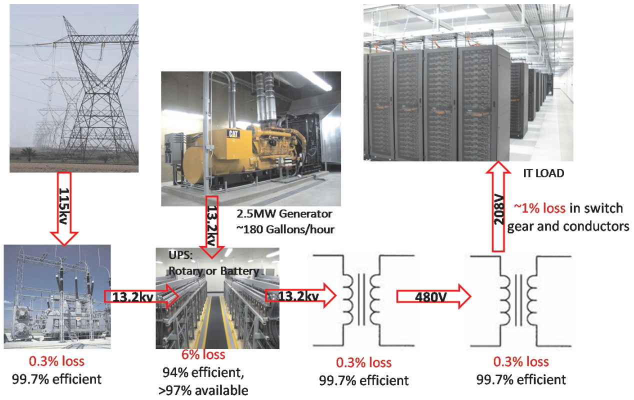


Figure 1: Infrastructure components. The utility (upper left hand corner) delivers 115KV, which transformers step down to 13.2KV, and deliver to the UPS (assumed to be battery-based here). In turn, transformers step the voltage down in stages and deliver it to the servers. In case of long term utility outages, generators (upper middle) keep the data center operational.

By infrastructure, we mean facilities dedicated to consistent power delivery and to evacuating heat. In some sense, infrastructure is the overhead of cloud services data centers. As Table 1 indicates, the aggregate cost is substantial. As depicted in Figure 1, drawing power from the utility leads to capital investments in large scale generators, transformers, and Uninterruptible Power Supply (UPS) systems. These are not commodity parts — for some the time between order and delivery is 8 months or more. With typical infrastructure cost of $200M, 5% cost of money, and 15 year amortization, the cost of infrastructure comes to $18.4 million/year.

2.2.1 *Reducing These Costs*

Driving the price of the infrastructure to these high levels is the requirement for delivering consistent power. What if we relax that requirement? Relaxing the requirement for individual server resilience led to scale-out data center designs based on very large numbers of commodity, low cost servers, with resilience *in the system* even though the components have relatively high failure rates. What if we were to deploy networks including larger numbers of smaller data centers. Among appropriate groups of these data centers, the target is 1:N resilience at data center level, that is, the failure unit becomes an entire data center. With resilience at the data center level, layers of redundancy within each data center can be stripped out (e.g., the UPS and the generators are not needed).

There is a rich problem space here, including designing strategies for balancing resilience within each data center against resilience across data centers. In Section 5 we discuss the issues around geo-diverse deployment of micro data centers, which hold the potential to provide both a relatively high degree of independence between physical data center outages (e.g., power outages), and an opportunity to economically reach data center customers with low latency.

2.3 Power

To track where the power goes, we postulate application of state-of-the art practice based on currently well understood techniques and implementation based on good quality but widely available equipment. The Green Grid [6] provides a metric to describe data center Power Usage Efficiency (PUE) as PUE = (Total Facility Power)/(IT Equipment Power). A state of the art facility will typically attain a PUE of $\sim$1.7, which is far below the average of the world's facilities but more than the best. Inefficient enterprise facilities will have a PUE of 2.0 to 3.0 [7], and very rare industry leading facilities are advertised as better than 1.2. These later reports are difficult to corroborate.

To estimate power draw for a mega data center, we assume a PUE of 1.7, a reasonable utility price of $.07 per KWH, 50,000 servers with each drawing on average 180W (servers draw as much as 65% of peak when idle), the total cost comes to $50,000 \cdot 180/1000 \cdot 1.7 \cdot \$0.07 \cdot 24 \cdot 365 = \$9.3$ million a year. Out of each watt delivered, about 59% goes to the IT equipment, 8% goes to power distribution loss, and 33% goes to cooling.

2.3.1 Reducing These Costs

Decreasing the power draw of each server is clearly has the largest impact on the power cost of a data center, and it would additionally benefit infrastructure cost by decreasing the need for infrastructure equipment. Those improvements are most likely to come from hardware innovation, including use of high efficiency power supplies and voltage regulation modules. Barroso and Hlzle introduced the term *energy proportionality* to refer to the desirable property that a server running at N% load should consume N% power [11]. Creating servers that are closer to implementing energy proportionality would improve efficiency.

One area of innovation that is impacted by networking is the idea of running the data center hotter — literally reducing the amount of cooling to save money on cooling equipment and the power it consumes. Initial experiments show that equipment failure rates increase with temperature, so the research challenge becomes determining what and how to harden. For example, the network may have to become more resilient and more mesh-like.

2.4 Network

The capital cost of networking gear for data centers is a significant fraction of the cost of networking, and is concentrated primarily in switches, routers, and load balancers. The remaining networking costs are concentrated in wide area networking: (1) peering, where traffic is handed off to the Internet Service Providers that deliver packets to end users, (2) the inter-data center links carrying traffic between geographically distributed data centers, and (3) regional facilities (backhaul, metro-area connectivity, co-location space) needed to reach wide area network interconnection sites. The value of the wide area network is shared across the data centers, and its total cost exceeds the cost of networking within any one data center. Back-of-the-envelope calculations for wide area newtwork cost are difficult, as the costs defy a simple breakdown into quantities such as fiber miles or traffic volumes. Rather, the costs vary site by site, and vary in time with industry dynamics (e.g., with tariffs, changing options for regional and wide area transport, and for peering). These costs have decreased dramatically over the past few years, but they remain signficant (e.g., wide area transport costs have decreased from approximately $100 per Mbps per month to roughly $5 per Mbps per month).

2.4.1 Reducing These Costs

Wide area networking costs are sensitive to site selection, and to industry dynamics. Accordingly, clever design of peering and transit strategies, combined with optimal placement of micro and mega data centers, all have a role to play in reducing network costs. Another approach is optimizing usage of the network through better design of the services themselves — partitioning their functionality and their state between data centers. With micro data centers built out close to users, the latency of responses can be reduced, but under the threat of undue increases in wide area network costs. Taking into account data partitioning and replication, we need better methods for design and management of traffic across the network of data centers, as well as better algorithms to map users to data centers.

2.5 Perspective

Up until now, we have identified large costs and some large opportunities to attack them. Two rules of thumb emerge:

On is Better than Off: Given the steep fixed costs for a server installed in a data center and the server's three year lifetime, it is always better for the server to be on and engaged in revenue producing activity — that is what optimizes work per investment dollar. The challenge is achieving agility, so that any server can be applied to any problem at hand. This enables the creation of large pools of free servers with statistical multiplexing benefits, and it eliminates the structural and risk management reasons for over-construction that lead to low server utilization.

Build in Resilience at Systems Level: Infrastructure costs are high largely because each data center is designed so that it will never fail. These costs can be dramatically reduced by stripping out layers of redundancy inside each data center, such as the generators and UPS, and instead using other data centers to mask a data center failure. The challenge is creating the systems software and conducting the networking research needed to support this type of redundancy between data centers.

It is worth noting that some other optimizations have less potential for impact in cloud service DCs. Consider reducing power draw in internal data center networking equipment. Well over half the power used by network equipment is consumed by the top of rack switches — while drawing less power per device than other gear, they are far greater in number. A top of rack switch draws $\sim$60W, while supporting 20 to 40 servers, each drawing $\sim$200W. The result is cumulative network power draw is a small fraction of the total data center power draw, and economizing on network power draw provides little relative impact. Similarly, improving power distribution efficiency (e.g., using a more efficient UPS than the one considered above) will have relatively low impact, as power distribution is already fairly efficient.

We next discuss areas of work that do offer major opportunities to improve data center efficiency:

- To attack low utilization, we need better mechanisms for increasing network agility (Section 3) and providing appropriate incentives to shape resource consumption (Section 4).

- To attack the problem of lowering latency to end users and increasing the reliability of the cloud in an economical way, we need better mechanisms for joint optimization of network and data center resources (Section 5.1), and new systems and mechanisms for geo-distributing state (Section 5.2).

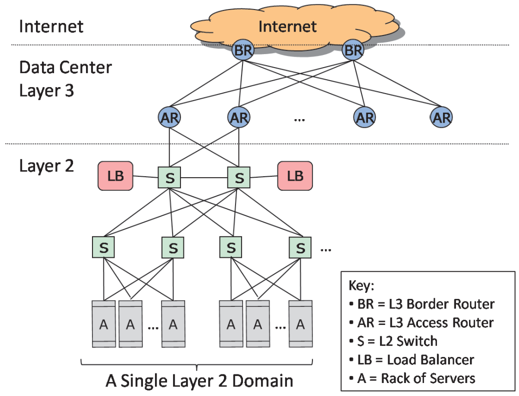
3. AGILITY

We define agility inside a single data center to mean that any server can be dynamically assigned to any service anywhere in the data center, while maintaining proper security and performance isolation between services. Unfortunately, conventional data center network designs work against agility - by their nature fragmenting both network and server capacity, and limiting the dynamic growing and shrinking of server pools. In this section, we first look at the network within the data center as it exists today and then discuss some desirable properties for a better solution.

3.1 Networking in Current Data Centers

Multiple applications run inside a single data center, typically with each application hosted on its own set of (potentially virtual) server machines. A single data center network supports two types of traffic: (a) traffic flowing between external end systems and internal servers, and (b) traffic flowing between internal servers. A given application typically involves both of these traffic types. In Search applications, for example, internal traffic dominates – building and synchronizing instances of the index. In Video download applications, external traffic dominates.

To support external requests from the Internet, an application is associated with one or more publicly visible and routable IP addresses to which clients in the Internet send their requests and from which they receive replies. Inside the data center, requests are spread among a pool of front-end servers that process the requests. This spreading is typically performed by a specialized hardware load balancer [23]. Using conventional load-balancer terminology, the IP address to which requests are sent is called a virtual IP address (VIP) and the IP addresses of the servers over which the requests are spread are known as direct IP addresses (DIPs).



and Design

Figure 2: The conventional network architecture for data centers (adapted from figure by Cisco [15]).

Figure 2 shows the conventional architecture for a data center, taken from a recommended source [15]. Requests arriving from the Internet are IP (layer 3) routed through border and access routers to a layer 2 domain based on the destination VIP address. The VIP is configured onto the two load balancers connected to the top switches, and complex mechanisms are used to ensure that if one load balancer fails, the other picks up the traffic [24]. For each VIP, the load balancers are configured with a list of DIPs, internal IP addresses over which they spread incoming requests.

As shown in the figure, all the servers that connect into a pair of access routers comprise a single layer 2 domain. With conventional network architectures and protocols, a single layer-2 domain is limited in size to about 4,000 servers in practice, driven by the need for rapid reconvergence upon failure. Since the overhead of broadcast traffic (e.g., ARP) limits the size of an IP subnet to a few hundred servers, the layer 2 domain is divided up into subnets using VLANs configured on the Layer 2 switches, one subnet per VLAN.

The conventional approach has the following problems that inhibit agility:

Static Network Assignment: To support internal traffic within the data center, individual applications are mapped to specific physical switches and routers, relying heavily on VLANs and layer-3 based VLAN spanning [19] to cover the servers dedicated to the application. While the extensive use of VLANs and direct physical mapping of services to switches and routers provides a degree of performance and security isolation, these practices lead to two problems that ossify the assignment and work against agility: (a) VLANs are often policy-overloaded, integrating traffic management, security, and performance isolation, and (b) VLAN spanning, and use of large server pools in general, concentrates traffic on links high in the tree, where links and routers are highly overbooked.

Fragmentation of resources: Popular load balancing techniques, such as destination NAT (or half-NAT) and direct server return, require that all DIPs in a VIP's pool be in the same layer 2 domain [23]. This constraint means that if an application grows and requires more front-end servers, it cannot use available servers in other layer 2 domains - ultimately resulting in fragmentation and under-utilization of resources. Load balancing via Source NAT (or full-NAT) does allow servers to be spread across layer 2 domains, but then the servers never see the client IP, which is often unacceptable because servers use the client IP for everything from data mining and response customization to regulatory compliance.

Poor server to server connectivity: The hierarchical nature of the network means that communication between servers in different layer 2 domains must go through the layer 3 portion of the network. Layer 3 ports are significantly more expensive than layer 2 ports, owing in part to the cost of supporting large buffers, and in part to marketplace factors. As a result, these links are typically oversubscribed by factors of 10:1 to 80:1 (i.e., the capacity of the links between access routers and border routers is significantly less than the sum of the output capacity of the servers connected to the access routers). The result is that the bandwidth available between servers in different parts of the DC can be quite limited. Managing the scarce bandwidth could be viewed as a global optimization problem – servers from all applications must be placed with great care to ensure the sum of their traffic does not saturate any of the network links. Unfortunately, achieving this level of coordination between (changing) applications is untenable in practice.

Proprietary hardware that scales up, not out: Conventional load balancers are used in pairs in a 1+1 resiliency configuration. When the load becomes too great for the load balancers, operators replace the existing load balancers with a new pair having more capacity, which is an unscalable and expensive strategy.

3.2 Design Objectives

In order to achieve agility within a data center, we argue the network should have the following properties:

Location-independent Addressing: Services should use location-independent addresses that decouple the server's location in the DC from its address. This enables any server to become part of any server pool while simplifying configuration management.

Uniform Bandwidth and Latency: If the available bandwidth between two servers is not dependent on where they are located, then the servers for a given service can be distributed arbitrarily in the data center without fear of running into bandwidth choke points. Uniform bandwidth, combined with uniform latency between any two servers would allow services to achieve same performance regardless of the location of their servers.

Security and Performance Isolation: If any server can become part of any service, then it is important that services are sufficiently isolated from each other that one service cannot impact the performance and availability of another. It is not uncommon for large-scale services to come under Denial-of-Service attacks or for certain services to generate disproportionate amounts of traffic due to code or configuration errors. The network must ensure that such traffic cannot impact other services co-located in the data center.

3.3 Current Approaches

A number of approaches are being explored to meet the requirements of intra-data center networks. Major commercial vendors are developing Data Center Ethernet (e.g., [14]), which uses layer 2 addresses for location independence and complex congestion control mechanisms for losslessness. Researchers have proposed designs for fast interconnects with varying degrees of location independence, uniform bandwidth, and performance isolation [16, 10]. Others have suggested using servers themselves as nodes in the interconnect [17].

4. INCENTING DESIRABLE BEHAVIOR

A different opportunity to get more work for each dollar invested in the data center stems from shaping resource consumption – a form of yield management. Designing mechanisms to implement economic incentives that encourage efficient behavior is a rich area for study and impact. Without reasonable incentives, customers (in particular, internal customers), have little to drive them to modulate their demand, leading to a vicious cycle of facilities procurement, followed by a lengthy period of highly bursty load and low utilization. Of top importance are the problems of trough filling and server allocation during times of shortage.

Trough filling: Periods of peak usage of network and power are relatively expensive to a data center – both resources are typically charged based on 95^{th} percentiles of usage, meaning that the cost is determined by the height of the peaks and not by the total area under the curve of usage across time. Thus, a large peak to valley ratio in the temporal usage pattern is inefficient, as the "troughs" in the usage curve can be filled at little additional cost. There are many "bin packing" opportunities to manage services to smooth resource consumption, at many levels of granularity. For example, ensuring leased/committed capacity with fixed minimum cost is always used is a safe way to improve efficiency. By setting prices that vary with resource availability, and by incenting service developers to differentiate demands by urgency for execution, workload can be shifted from peaks to troughs.

Server allocation: The creation of large unfragmented pools of servers will go a great distance towards improving agility and reducing the tendency of application operators to request more servers than they really need. However, eliminating the hoarding of servers depends on establishing a cost for having a server assigned to a service, so that there is a strong incentive to return unneeded servers to the free pool. Additional pricing mechanisms will be needed if seasonal peaks occasionally cause workload across many applications to peak simultaneously, resulting in server demand out stripping supply. (For example, the traffic to major retail websites all increase by a factor of 2-3 during the few weeks before Christmas.) In these situations, internal auctions may be the fairest and most efficient means to allocate servers among applications, but designing these auctions is relatively unbroken ground.

5. GEO-DISTRIBUTION

Speed and latency matter. There is substantial empirical evidence suggesting that performance directly impacts revenue [22]. For example, Google reported 20% revenue loss due to a specific experiment that increased the time to display search results by as little as 500 msecs. Amazon reported a 1% sales decrease for an additional delay of as little as 100 msecs. This creates a strong motivation for geographically distributing data centers around the world to reduce speed-of-light delays, but it also opens the door to additional opportunities and commensurate research challenges: determining where to place data centers; how big to make them; and using the geographic diversity of data centers as a source of redundancy to improve system availability.

5.1 Optimal Placement and Sizing

Placement and sizing of data centers presents a challenging optimization problem, involving several factors.

The first factor is the importance of geographic diversity. Placing data centers, whether mega or micro, in geographically separated areas has a number of benefits. First, it helps with decreasing the latency between a data center and the user (assuming users can be directed towards nearby DCs). Second, it helps with redundancy, as not all areas are likely to lose power, experience an earthquake, or suffer riots at the same time.

The second factor is the size of the data center. As described earlier, cloud services need some number of mega data centers to house large computations. The size of a mega data center is typically determined by extracting the maximum benefit from the economies of scale available at the time the data center is designed. This is an exercise in jointly optimizing server cost and power availability, and today leads to designs with 100,000s of servers spread over 100,000s of square feet drawing 10 to 20MW of power. Given the significant resource requirements, local factors, such as zoning, tax, and power concessions, play a large role in determining where to site a mega data center.

There are significantly more degrees of freedom in the sizing and siting of micro data centers. The minimum size of a micro data center is constrained by the need to have enough servers to provide statistical multiplexing gains and serve the workload generated by the local population while amortizing the fixed costs of the site and DC to acceptable levels. The maximum size of a micro DC is constrained by the desire that its physical size and power draw be small enough to place few restrictions on the placement of the DC. One emerging innovation in the industry is DCs constructed from servers housed in shipping containers, each container housing roughly a thousand servers and drawing less than 500 KW [18]. For comparison, the average power draw of the average American home is 1.2 KW. Another factor limiting the size of micro data centers is economic: given a fixed budget to spend on data centers, the desire to put a DC close to each desired population segment caps the size of each DC.

The third factor is network cost. One would like to place data centers as close to the users as possible while minimizing the cost and latency of transferring data between various data centers. One challenge is to find an optimal balance between performance and cost while placing micro data centers near (e.g., within tens of milliseconds) major population centers and fiber hotels supporting access to low cost Internet peering, and access to low cost dedicated or leased lines between data centers.

A more sophisticated optimization would also take into account the dependencies of the services offered from the data centers. For example, an email service may depend on an authentication service, an ad insertion service, and a buddy list maintenance service; these dependencies may call for intense and/or low latency communications. Services are often created in tiers of server pools. It is possible, for example, to decompose some services into a front end tier and a back end tier, where the front ends are mapped in micro data centers to minimize latency, and the back end to mega data centers to leverage greater resources. Several other non-technical factors contribute to deciding where to place data centers and how large to make them, including tax policies, and the target markets for the services being hosted.

5.2 Geo-Distributing State

As noted in Section 2, a key opportunity for reducing the cost of cloud service data centers is to eliminate expensive infrastructure, such as generators and UPS systems, by allowing entire data centers to fail. Turning geo-diversity into geo-redundancy requires that the critical state for data center applications be distributed across sites, and frameworks to support this remain a non-trivial systems and networking research problem.

The state-of-the-art is that every service implements its own solution for geo-distribution. For example, Facebook replicates data with all writes going through a single master data center [2]. Yahoo! mail partitions data across DCs based on user [9]. Many cloud services are re-inventing the wheel, or worse, not geo-distributing at all, because robust, general mechanisms and programming APIs for state distribution have not yet been designed. There is a very large design space for such mechanisms, and different solutions may be optimal for different types of data. For example, data such as a buddy list and buddy status fits naturally to a model where information is replicated between data centers with weak consistency assurances. On the other hand, email maps naturally to a model where each collection of data (a mailbox) is accessed by a single user, and the data can be partitioned across data centers by user ID's, with strong consistency assurances.

There are several tradeoffs to consider, such as managing the balance between load distribution and service performance. For example, the Facebook design has a single master coordinate replication — this speeds up lookups but concentrates load on the master for update operations [2]. There are also tradeoffs between communication costs and service performance to optimize. For example, data replication involves more inter-data center communication than data partitioning. Also, spreading services across data centers turns what had been internal messages between services into longer latency, higher cost messages over inter-DC links.

6. CONCLUSIONS

Data center costs are concentrated in servers, infrastructure, power requirements, and networking, in that order. Though costs are steep, utilization can be remarkably low. We identified several approaches to significantly improve data center efficiency. First, we need to increase internal data center network agility, to fight resource fragmentation and to get more work out of fewer servers – reducing costs across the board. Second, we need to pursue the design of algorithms and market mechanisms for resource consumption shaping that improve data center efficiency. Finally, geo-diversifying data centers can improve end to end performance and increase reliability in the event of site failures. To reap economic benefits from geo-diversity, we need to design and manage data center and network resources as a joint optimization, and we need new systems to manage the geo-distribution of state.

7. ACKNOWLEDGMENTS

We would like to thank our colleagues for their insights: Sharad Agrawal on the challenges of geo-distribution, Chuck Thacker on economizing internal data center networks and systems, Blaine Christian on wide area networking and incentives, and David Treadwell on resource consumption shaping.

8. REFERENCES

[1] Amazon Web Services. URL http://aws.amazon.com.

[2] Engineering @ Facebook's Notes: Scaling Out. URL http://www.facebook.com/notes.php?id=9445547199.

[3] Google app engine. URL http://code.google.com/appengine/.

[4] Google docs and spreadsheets. URL http://docs.google.com.

[5] Microsoft office live. http://office.live.com.

[6] The Green Grid. URL http://www.thegreengrid.org.

[7] The Uptime Institute. URL http://uptimeinstitute.org.

[8] Windows Azure. URL http://www.microsoft.com/azure/.

[9] Yahoo! Mail. URL http://mail.yahoo.com.

[10] M. Al-Fares, A. Loukissas, and A. Vahdat. A scalable, commodity data center network architecture. In *SIGCOMM*, 2008.

[11] L. A. Barroso and U. Hlzle. The case for energy-proportional computing. *IEEE Computer*, 40, 2007.

[12] A. Brown and D. A. Patterson. Embracing Failure: A Case for Recovery-Oriented Computing (ROC). In *High Performance Transaction Processing Symposium*, 2001.

[13] K. Church, J. Hamilton, and A. Greenberg. On delivering embarassingly distributed cloud services. In *Hotnets VII*, October 2008.

[14] Cisco. Data center ethernet. http://www.cisco.com/en/US/-netsol/ns783/networking_solutions_package.html.

[15] Cisco systems: Data center: Load balancing data center services, 2004.

[16] A. Greenberg, P. Lahiri, D. A. Maltz, P. Patel, and S. Sengupta. Towards a next generation data center architecture: Scalability and commoditization. In *PRESTO Workshop at SIGCOMM*, 2008.

[17] C. Guo, H. Wu, K. Tan, L. Shiy, Y. Zhang, and S. Luz. Dcell: A scalable and fault-tolerant network structure for data centers. In *SIGCOMM*, 2008.

[18] J. Hamilton. Architecture for modular data centers. In *Third Biemnial Conference on Innovative Data Systems*, 2007.

[19] IEEE802.1Q. IEEE Standard for Local and Metropolitan Area Networks: Virtual Bridged Local Area Networks, 2005.

[20] M. Isard. Autopilot: Automatic data center management. *Operating Systems Review*, 41(2), 2007.

[21] Z. Kerravala. Configuration management delivers business resiliency. The Yankee Group, Nov 2002.

[22] R. Kohavi, R. M. Henne, and D. Sommerfield. Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO. *KDD*, 2007.

[23] C. Kopparapu. *Load Balancing Servers, Firewalls, and Caches*. John Wisely & Sons Inc., 2002.

[24] E. R. Hinden. Virtual router redundancy protocol (VRRP). RFC 3768, 2004.

[25] W. Enck et al. Configuration Management at Massive Scale: System Design and Experience. *IEEE JSAC - Network Infrastructure Configuration*, 2008.