IEEE

CLOUD COMPUTING

CLOUD NATIVE APPLICATIONS

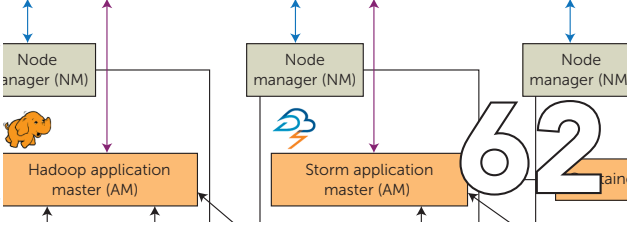
SEPTEMBER/OCTOBER 2017
VOLUME 4, NUMBER 5
www.computer.org/cloudcomputing

IEEE
IEEE computer society
IEEE ComSoc IEEE Communications Society

productize common business and technical services
similar applications and services services

anges to the application to address issues; leverage

12



62

September-October 2017
Volume 4, Issue 5
www.computer.org/cloudcomputing

COLUMNS

The Economics of Computing Workload Aggregation: Capacity, Utilization, and Cost Implications

According to the NIST definition, one key characteristic of cloud computing is "resource pooling to serve multiple consumers using a multitenant model, with different physical and virtual resources dynamically assigned and reassigned according to consumer demand."[1] In other words, instead of running in silos, workloads with time-varying demands or processing intensities are often aggregated, assigned, and provisioned into a shared pool of resources. One obvious example is cloud computing, where workloads—say, claims processing, shopping carts, and video transcoding—from multiple customers—say, insurers, ecommerce companies, and online streaming entertainment firms—are aggregated together, mapped to a shared pool of physical resources such as servers, and then execute on those resources. This is similar to the way that lodging needs of travelers are aggregated together; guests are assigned actual hotel rooms; and then are checked in and move their luggage up to their room for the duration of their stay.

There are many benefits of this type of aggregation. First, less capacity is typically needed for the pool than were individuals to build their own capacity. Second, the utilization of this capacity is greater,

EDITOR
JOE WEINMAN
joeweinman@gmail.com

 2325-6095/17/$33.00 © 2017 IEEE

because the same amount of work is being handled by fewer resources. Thirdly, therefore, the utilization-adjusted cost of a unit of sold capacity can be lower. In this issue's column, we'll quantify those benefits exactly.

Aggregation of workloads occurs in many instances in computing. One case is when siloed workloads, each running in their own dedicated capacity, are virtualized and aggregated into a private cloud of dynamically allocated, shared resources. Another case is when such private clouds from multiple customers are aggregated into a public cloud. Yet another case is when multiple public clouds are federated to share capacity.

Moving beyond these traditional and emerging cloud approaches, fog or edge computing is an emerging paradigm for computing also subject to these same laws, for better or worse.[2] The fog approach inserts one or more layers between highly centralized hyperscale cloud datacenters and endpoint devices such as user smartphones or laptops or "things", such as nannycams, connected light bulbs, or connected vehicles. Although it is typical to think of the fog as moving some cloud functionality closer to users and things, it is equally fair to view the fog as moving some "thing" functionality closer to the cloud, for example, aggregating control functionality or data management—say, in programmable logic controllers or robots or smart door locks or surveillance cameras—to a higher layer. This can have a number of benefits, such as reduced latency for time-sensitive functions—say, controlling a flexible manufacturing cell with multiple milling machines, materials handling systems, and/or interacting robots—and reducing backhaul data transport requirements—say, by compressing data or only transmitting exceptions to a centralized location. To put it another way, because fog resources are dispersed at the edge, round trip latency from a user or thing to the fog and back to that user or thing will be lower than a round trip from user or thing all the way to the cloud and back to the user or thing. On the other hand, applications in the cloud will have better response times when accessing data or services in the same cloud datacenter.

From an aggregation perspective, however, the same principles apply: reduced capacity requirements, higher utilization, and therefore lower costs. While latency and transport costs are beneficially impacted through a fog approach, cloud trumps fog for capacity, utilization, and cost. With that same focus, fog trumps devices: aggregating computing needs from lower-layer devices into an intermediate fog layer is also beneficial for capacity, utilization, and cost. As a general rule, centralizing formerly dispersed resources and functionality hurts latency and data transport costs to and from endpoints, but helps with total resource costs. There are also other factors to consider, for example, business continuity implications, remote management needs, physical security, cybersecurity and vulnerability, and storage replication impacts, which are beyond the scope of this column.

Capacity Requirements

One essential characteristic of the cloud is that available capacity is pooled and dynamically shared to support multiple time-varying workload demands. For example, tax preparers have peaks in early February and mid-April, flower sellers have peaks on Valentine's Day and Mothers' Day, retailers have peaks on cyber Monday or Single's Day, etc. The total capacity that is deployed impacts the total investment requirements of whoever is deploying that capacity, e.g., an enterprise for a private cloud or a cloud service provider for a public cloud. The capacity conundrum is that if there is insufficient capacity, enterprise workloads will either not be run or will run slowly, incurring a cost in labor productivity for internal applications and/or revenue and customer experience for external ones; if there is excess capacity it will require an excessive investment, also incurring a cost through cost of capital and/or opportunity cost of idle capital that could be more productively deployed elsewhere.[3] For cloud service providers, excess capacity means wasted investment, and insufficient capacity implies lost revenue.

Workload demand aggregation can reduce capacity requirements compared to unaggregated demands. Such aggregation can happen by aggregating workloads that would otherwise be siloed into a pool of resources dynamically shared via a private cloud; further aggregation happens when siloed private cloud workloads run by individual enterprises are brought together in an environment such as a community or public cloud shared across multiple enterprises; and maximum aggregation occurs when multiple public clouds are virtually aggregated by a federated cloud or Intercloud.[4] This sort of "east-west" aggregation can also be complemented by a "north-south" aggregation, when dispersed fog resources are aggregated into the cloud, or dispersed device functionality is aggregated into the fog.

To quantify these kinds of effects, let us model the demands of n independent compute workloads as D_1, D_2, … D_n. To keep things simple, we'll assume that they are all identically distributed with mean μ and variance σ^2. Let's further assume that they are normally distributed, although most of the results here apply regardless of distribution. If

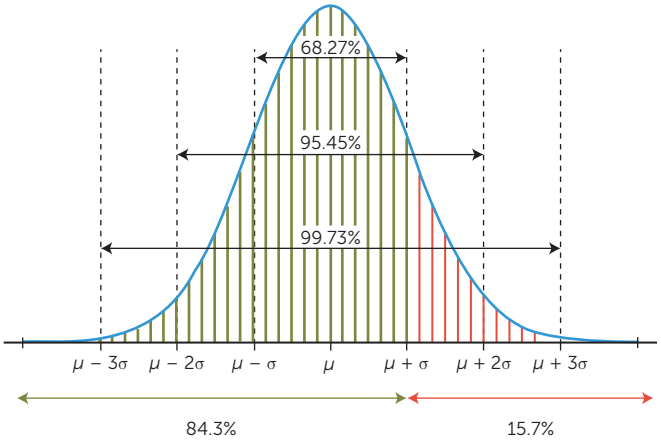


FIGURE 1. Capacity set to $\mu + 1\sigma$.

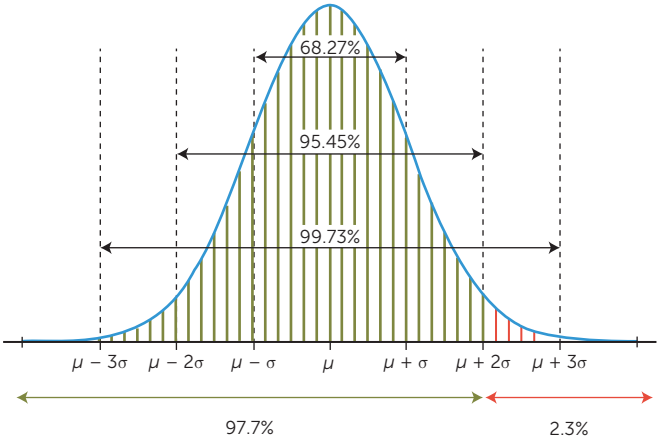


FIGURE 2. Capacity set to $\mu + 2\sigma$.

we aggregate these n demands together, the sum will be normally distributed, its mean will be $n\mu$ and its variance will be $n\sigma^2$, which means that the standard deviation of the sum will be $\sqrt{n}\sigma$. Whether for one such workload or the totality, we'd like to balance availability of compute resources with cost of compute resources, so generally will try to find a value of k standard deviations above the mean to set the capacity. For example, for a normally distributed variable with mean μ and variance σ^2, the probability that it has a value between $\mu - \sigma$ and $\mu + \sigma$ is 0.682689. Therefore, if we set capacity to one standard deviation above the mean, we know that roughly 68% of demand levels will fall within that one standard deviation, with 16% falling on either side (i.e., in each tail). This means that if we set capacity to $\mu + \sigma$, we will have sufficient capacity all but 16% of the time, as shown in Figure 1.

As shown in Figure 2, if we set capacity to $\mu + 2\sigma$, because 95.4% of the normal distribution falls within $\mu - 2\sigma$ and $\mu + 2\sigma$, only 4.6% falls outside of the range, therefore there will be sufficient capacity 97.7% of the time. Such theoretical availability numbers, i.e., likelihood of sufficient capacity, don't model other potential real-world issues, of course, such as physical "smoking hole" disasters, Distributed Denial of Service attacks, power outages, cloud data center operations issues, bankruptcies, and the like.

When we aggregate capacity, there is a smoothing effect, where a peak in one workload's demand is likely to be balanced by a trough in another. The more independent workload demands that we combine, the smoother the aggregate is. The statistic that captures this behavior is the coefficient of variation: σ/μ, in other words, the size of the variation relative to the size of the mean. After all, skyscrapers with heights varying from each other by 10 feet represents hardly any variation, whereas if people, kittens, or bacteria each varied by that amount, it would be very noticeable.

When we aggregate n independent demands with standard deviation σ and mean μ, the denominator of the coefficient of variation of the sum grows to $n\mu$, but the numerator only grows to $\sqrt{n}\sigma$, so it will be appreciated that the coefficient of variation gets smaller and smaller, i.e., the aggregate demand gets smoother/flatter, and becomes zero in the limit as n approaches infinity. At that (theoretical) point, there would be no variation; there would be perfectly flat aggregate demand, and capacity would have 100% utilization.

Suppose that we would like to ensure that capacity is sufficient some given percentage of the time. For example, if we wanted sufficient capacity 97.7% of the time for a demand with mean μ and standard deviation σ, we would set it to $\mu + 2\sigma$, in line with the earlier discussion regarding setting capacity by relating it to the standard deviation of demand. If we *don't* aggregate demands

and pool resources, we would then set capacity to a specific $\mu + k\sigma$ for each of the same siloed (i.e., unpooled) resources, for a total capacity of $n(\mu + k\sigma) = n\mu + kn\sigma$. However, if we *do* aggregate demands and serve them out of pooled resources, for that same sufficiency, we only need to set capacity to $n\mu + k\sqrt{n}\sigma$.

To illustrate, consider the demand level at various times from a workload with a mean demand of 10 units and a standard deviation of 2 units, as shown in Figure 3. This could be the number of thousand virtual machines needed each day, or, in other domains, the number of cars sold each day, or the number of emergency room visits each hour.

If we were to examine 16 such workloads in aggregate, as shown in Figure 4, we end up with an aggregate demand that has a mean of 160 and standard deviation of 8 ($\sqrt{16} \times 2$). Thus, the coefficient of variation is reduced from 0.2 = 2/10 to 0.05 = 8/160. This is easy to see by comparing Figure 3 with Figure 4; Figure 4 has a scatter plot which is "tighter".

As another example, we can calculate what would happen if we were to set capacity to be three standard deviations above the mean. In the case of partitioned demands and unpooled resources, the total capacity we would need is $n(\mu + k\sigma)$, or $256 = 16(10 + 3 \times 2)$, whereas if demands and resources were aggregated, we would only need $n\mu + k\sqrt{n}\sigma$ or $184 = 16 \times 10 + 3 \times 4 \times 2$.

Of course, the actual difference will vary based on the mean μ, the standard deviation σ, the desired capacity headroom factor k, and the number of aggregated workloads n.

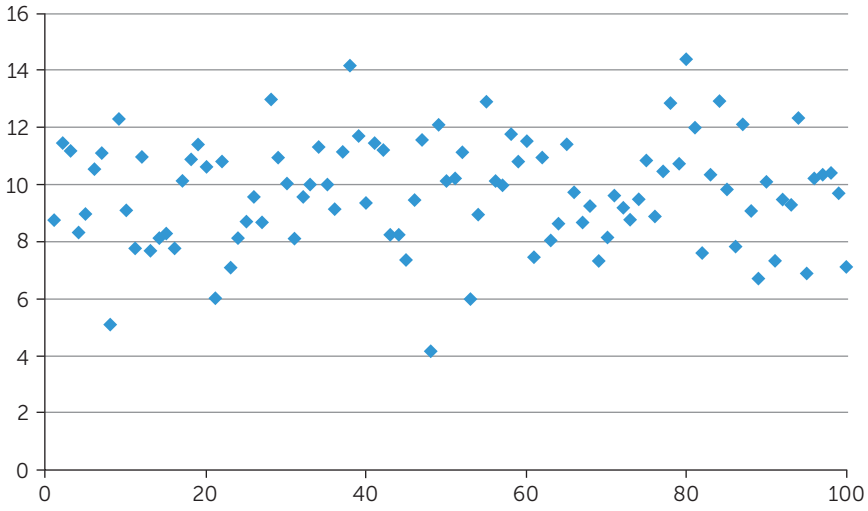


FIGURE 3. 100 samples from a normally distributed random variable X with $\mu = 10$, $\sigma = 2$.

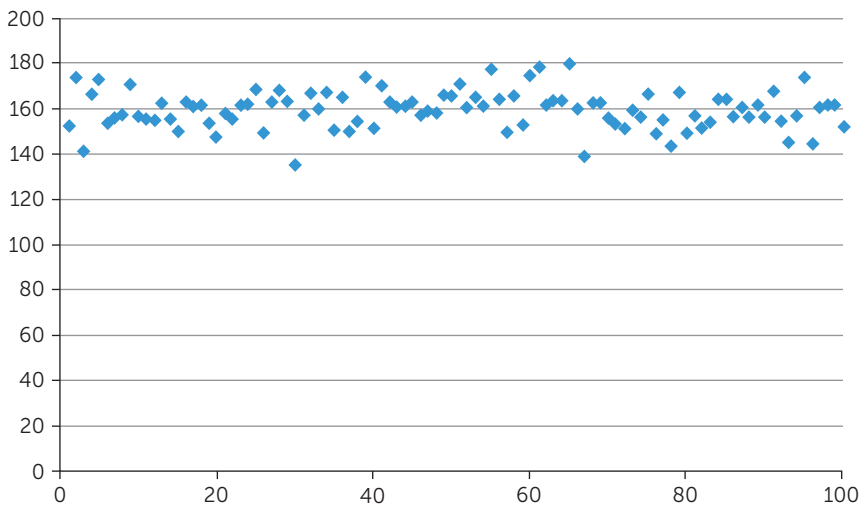


FIGURE 4. 100 samples from a random variable that is the sum of 16 independent, identically distributed normal random variables with $\mu = 10$, $\sigma = 2$.

Capacity Utilization

Another important metric is the expected value of utilization of total deployed capacity, which is just the mean demand divided by the total capacity. For a single resource, that is $\mu/(\mu + k\sigma)$, and therefore for n *unaggregated* demands served out of n such resources it's the mean total workload demand divided by the total deployed capacity, or $n\mu/[n(\mu + k\sigma)]$. In the case of aggregated workload demands, it's the same total work executing on lesser capacity, so the utilization increases to $n\mu/(n\mu + k\sqrt{n}\sigma)$. Figure 5 shows how workload aggregation monotonically increases to utilization of 100% in the limit, as a function of these four variables. In other words, utilization $U = f(n, \mu, \sigma, k)$.

To put it simply, dynamic aggregation of varying, independent workload demands into pooled capacity always generates benefits in terms of capacity

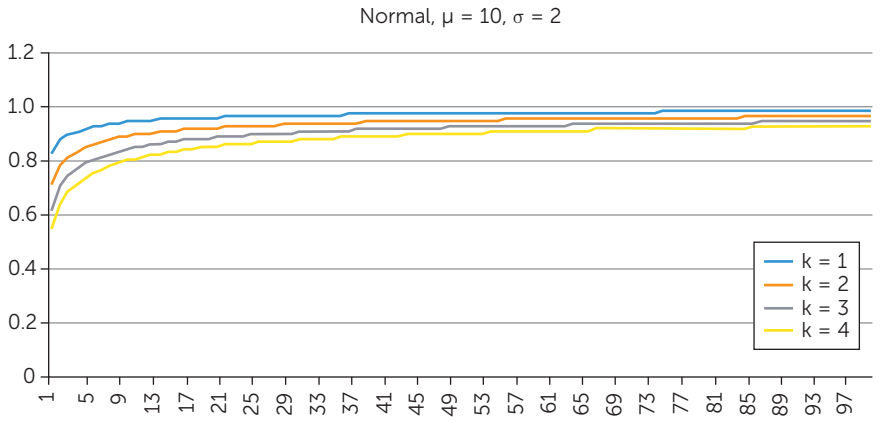


FIGURE 5. Utilization $U = f(n, \mu, \sigma, k)$ where $\mu = 10$ and $\sigma = 2$ for various k from $n = 1$ to 100.

requirements and capacity utilization—briefly stated as less capacity, that is better utilized—whether it is from dedicated application silos to a private cloud, from private clouds to a public cloud, from multiple public clouds to an intercloud or federated cloud, from dispersed fog to cloud, or from individual devices to the fog. Of course, these benefits can come at a price, such as data transfer costs and latencies, security vulnerabilities, or extra software to implement dynamic resource allocation.

Unit Resource Cost Improvements

These capacity and utilization effects come together to impact the overall economics of demand aggregation, impacting chief information officers, chief financial officers, and IT personnel as they think about implementing private clouds or migrating to a public cloud. Consider a simple analogy. If you ran a fruit stand and needed to throw away a rotting peach for every fresh peach that you sold, your economics would be disadvantaged compared to a competitor of the same size with less shrinkage.

In a similar way, if you have more resources that are less well utilized, you will have a unit cost disadvantage compared to someone with fewer resources and higher utilization levels. In other words, for the same amount of work, more resources and lower utilization are two sides of the same coin. Even without scale economies, the statistics of aggregation provide a basis for a lower delivered cost.

For the fruit stand, the unit *price* of a *sellable/consumable* peach reflects the unit acquisition cost of a peach * (the total number of sellable/consumable *plus* unsellable/unconsumable peaches acquired)/the number of sellable/consumable peaches. In other words, the sales price has to increase based on the ratio of total peaches to good peaches. As the number of bad peaches increases, the selling price has to increase to reflect that loss, assuming profit margins and peach acquisition costs remain the same. Conversely, the sales price can decrease as there are fewer bad peaches; ideally, none.

Instead of peaches, let's consider compute resources. Then, the unit *price* of a *sellable/consumable* compute resource reflects the deployment costs of a resource * (the total number of sellable/consumable *plus* unsellable/unconsumable resources deployed)/the number of sellable/consumable resources. In other words, instead of the sales price increasing based on the ratio of total peaches to good peaches, the sales price has to increase based on the ratio of total capacity to "good," i.e., consumed, capacity. Peaches are perishable, as is computing capacity, which can't be inventoried and thus rots instantly when it isn't used.

The price increase penalty is easy to determine from the formulas above. The expected value of the price increase for unaggregated resources is just:

$$\frac{\textit{total unaggregated capacity}}{\textit{used unaggregated capacity}} = \frac{n(\mu + k\sigma)}{n\mu}$$

The price increase for aggregated resources is:

$$\frac{\textit{total aggregated capacity}}{\textit{used aggregated capacity}} = \frac{n\mu + k\sqrt{n}\sigma}{n\mu}$$

By dividing by n throughout, this works out to:

$$= \frac{\mu + \frac{k\sigma}{\sqrt{n}}}{\mu}$$

Since k, μ, and σ are constants, it is pretty clear that, in the limit as $n \to \infty$, there is effectively no penalty.

To put it another way, the denominators don't change, since the application workloads remain the same. There is the same amount of work either way; and the same amount of good peaches either way. The same number of people (n) still buy the same average number of peaches each (μ) or do the same average amount of computing. Therefore, the ratio of the penalty for siloing vs. aggregation is

$$\frac{\frac{n(\mu + k\sigma)}{n\mu}}{\frac{n\mu + k\sqrt{n}\sigma}{n\mu}}$$

This of course is just:

$$\frac{n(\mu + k\sigma)}{n\mu + k\sqrt{n}\sigma}$$

Figure 6 shows the cost penalty for any given number of workloads n and given capacity level of $\mu + k\sigma$, for the case where μ is 10 and σ is 2. To use our favorite example, where n is 16 and k is 3, this works out to 256/184 = 1.39, a 40% cost penalty. If we increase n to 100, the difference is 1600/1060 = 1.51, a 50% cost penalty.

Caveats

The real world may not always match the conditions explored above. For example, workload variation may not be normally distributed, workloads may not have the same mean or standard deviation, dynamic pricing (as with spot instances) may be used to smooth capacity by promoting demand, there are network costs to demand aggregation, and so forth.

However, the underlying insights are unassailable: demand is smoothed by aggregation of independent, uncorrelated workloads and therefore capacity requirements to meet a given availability target, in the sense of sufficient capacity, can be lower, utilization of that reduced capacity can be higher, and unit costs and prices of sold resources can therefore be lower after aggregation. These benefits occur immediately—with the second workload to be aggregated—and from there exhibit decreasing marginal returns to scale, before considering any scale economy effects.

Summary

Aggregation creates unarguable economic benefits in terms of capital investment required for service resource capacity; utilization of that capacity; and the delivered unit cost. These benefits offer a series of advantages for private cloud over siloed resources; public cloud over a private cloud; cloud over fog; fog over devices/things, and so forth. These advantages need to be traded off against other factors, for example, response time, data networking costs, etc.

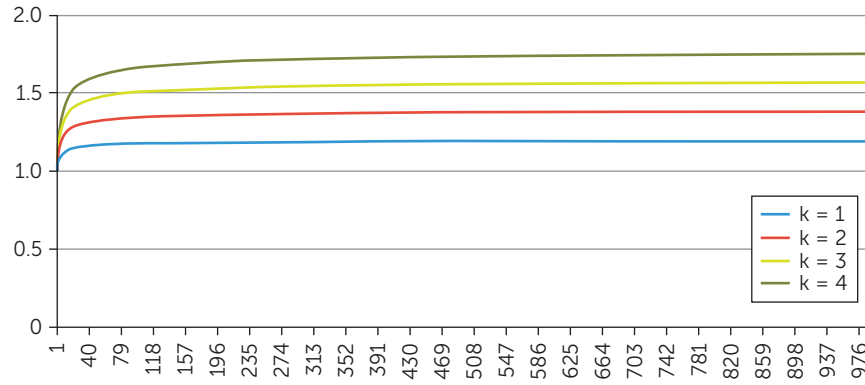


FIGURE 6. Relative cost for siloed resources vs. aggregated resources as the number of workloads increases, given four different capacity headroom settings.

References

1. P. Mell, and T. Grance, "The NIST Definition of Cloud Computing," NIST Special Publication 800-145, Sept. 2011; http://nvlpubs.nist.gov/nistpubs/Legacy/SP/nistspecialpublication800-145.pdf.
2. F. Bonomi et al., "Fog Computing and Its Role in the Internet of Things," *Proc. ACM Mobile Cloud Computing (MCC'12)*, Aug. 17, 2012; http://conferences.sigcomm.org/sigcomm/2012/paper/mcc/p13.pdf.
3. J. Weinman, *Cloudonomics: The Business Value of Cloud Computing*, Wiley, 2012.
4. J. Weinman, "Intercloudonomics: Quantifying the Value of the Intercloud," *IEEE Cloud Computing*, Sept./Oct. 2015, pp. 40–47.

JOE WEINMAN *is a frequent global keynoter and author of Cloudonomics and Digital Disciplines. He has held executive leadership positions at AT&T, HP, and Telx. He also serves on the advisory boards of several technology companies. Weinman has a BS in computer science from Cornell University and an MS in computer science from the University of Wisconsin-Madison. He has completed executive education at the International Institute for Management Development in Lausanne. Weinman has been awarded 22 patents. Contact him at joeweinman@gmail.com.*