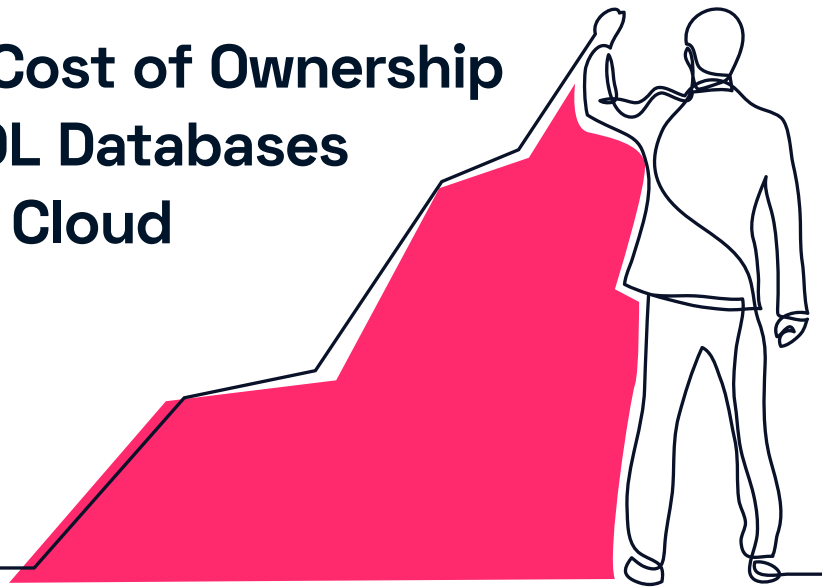


Total Cost of Ownership for SQL Databases in the Cloud



Companies are hosting their mission-critical applications in the cloud to take advantage of the cloud's on-demand resources, availability, and elasticity. But what do you need to think about when hosting SQL databases in the cloud? The price can come with sticker shock if your workloads have high I/O performance requirements and you end up overprovisioning cloud resources.

We broke down the Total Cost of Ownership (TCO) of hosting SQL databases in the cloud with David Berliner, Senior Director of Product, and Jack Vaughan, Principal Architect at Silk. They discussed the three elements of cost, the cloud's "shared nothing" architecture, and how to avoid overprovisioning.

Watch the full [video conversation here](#) and read on for a recap.

The Three Elements of Cost: Storage, Compute and Database Licenses

The three main elements of cost when hosting SQL databases in the cloud are storage, compute, and database licenses. We discuss how each element is provisioned in a cloud environment and how to mitigate overprovisioning for cost-efficient operations and scaling.

First is storage. When you think about what your costs will be for storage, you want to start not with capacity but with performance, including throughput, IOPS, and latency, as performance determines the tier of storage that you're able to use. You will provision storage up to the performance requirements of the database instance. Then, you might have additional capacity to add, but it's more likely you will have already hit or exceeded the capacity needed based on your performance requirements.

Next, will you provision for the average or for the peak? Because of the cloud's "shared nothing" architecture (more on that next), you need to make sure you are provisioning storage properly for each database. Unlike an on-premises data center, there is no SAN to share across database installs. If provisioning for the average, how are you going to scale up or down for peak? You will save money by not provisioning for the peak on a normal basis but need to think through the operational burdens to scale that up or down or curtail performance during peak workloads.

The second element of cost is compute. Start with the set up that you need based on the database cores, memory, and so forth, and then factor in the relationship that you have between the compute virtual machines and the data network. Typically, the data network in the cloud has about one gigabyte per second of throughput, with the ability to burst to two gigabytes per second for a limited duration. And to hit that, you might need to overprovision beyond what the database needs itself to maximize the potential and complement what you are provisioning on the storage side, even if you're able to hit the numbers that you need for storage.

Third is database licenses. The number of database licenses required will scale up and down with the size of or number of vCPUs of the compute virtual machine. If you are having to overprovision the compute virtual machine, you're going to overprovision your database licenses along the way. The cloud providers have a couple of ways to help mitigate this somewhat, including constrained vCPUs or perpetual licenses that you can bring to the cloud for SQL databases.

In all three areas – storage, compute and database licenses – there is inherent overprovisioning due to the cloud's architecture. And it becomes more troublesome for high I/O performance workloads.

The Cloud's Shared Nothing Architecture Leads to Overprovisioning

While the cloud is flexible and elastic, it has some quirks in its infrastructure. It has a "shared nothing" architecture. By "shared nothing," we mean its elements are associated directly and thickly provisioned. There is specific storage supporting a specific database. There are rigid instance types, VM shapes, and database options. This can lead to overprovisioning based on the performance and capacity characteristics of these fixed options. And it also makes it harder to achieve economies of scale.

Not all database implementations will fit the VM shapes in the cloud. Many customers have performance or I/O constraints. This leads to overprovisioning or in rare cases the inability to host a database using cloud native components.

As Jack explains in the webinar, performance is gated in multiple places in the cloud. We typically see I/O limitations on the size of the managed disk that's being provisioned and on the instances that you're provisioning to. On average, customers are overprovisioning by 30%, but we have seen as high as 70% to 80% to get the performance they need. The cloud has mitigated some of this by providing vCPU constrained instances, which has helped some on the licensing side, but we've seen up to 75% overprovisioning of the vCPUs when using these constrained vCPU instances.

There hasn't been a good solution to overprovisioning until now!

Silk Solves Overprovisioning and Delivers Scalable Performance

The Silk Cloud DB Virtualization Platform solves the problem of overprovisioning.

It brings shared data architecture and enterprise data services to the cloud. Customers save significantly on storage, compute, and licensing costs, while enabling databases of any performance requirements to run in the cloud.

How does Silk work?

We start with performance.

While Silk is great for enabling high I/O databases because its **architecture** enables up to 10x faster performance than cloud native alone, its architecture also allows you to start consolidating workloads and drive economies of scale. Customers provision for the average workload, combining the needs of one or more databases on a shared resource for greater efficiency. You can scale out your performance on demand when you need more performance and then scale back in – all without having to scale any of your capacity. We start to right size your cloud infrastructure and eliminate overprovisioning.

Next, we align capacity.

We eliminate overprovisioning and start to utilize **thin provisioning**. We typically see anywhere from 1.5-2x smaller footprint when starting to use Silk instead of cloud native. Then we start to apply Silk's real-time data reduction, data compression, and deduplication to further decrease that footprint. We see anywhere from 2.5-3.5x data reduction. Combining thin provisioning with data reduction, that's around 5x smaller data footprint.

And then finally, customers can use Silk's thinly provisioned snapshots.

Thin snapshots are not available natively on the cloud. On the cloud, they are full copies that take up space, are slow to create, and costly to manage. Customers can use Silk's **instantaneous, zero-footprint snapshots** to refresh non-production and reporting environments in no time.

We typically see anywhere from 5x to 10x data reduction if you're taking all three of these data features into consideration when looking at the platform, with some customers reaching even higher levels of consolidation.

In addition, Silk has a RESTful API that can connect to Terraform, Ansible Playbooks, PowerShell SDK, and other tools that can automate the provisioning of your capacity — all the way to being able to create thin snapshots and refresh your environments in minutes instead of hours.

Customers save up to half of their storage and compute costs (or more, with data services) while the need for SQL database licenses shrinks in line with compute.

Customers save up to half of their storage and compute costs (or more, with data services) while the need for SQL database licenses shrinks in line with compute.

One customer was 70% overprovisioned on their premium SSDs. They thought that reducing this cost would be the big cost saver. But the thing that really transformed their ability to cut costs was that they were using constrained vCPU instances. The customer was overprovisioning their vCPUs by 75%. They were using E64's constrained to 16 cores. But by using Silk, the customer was able to get higher I/O performance with an E16 than they were getting with an E64. They lowered their cloud TCO significantly.

Summary

There are three main elements of cost when hosting SQL databases in the cloud: storage, compute, and database licenses.

The cloud's architecture inevitably leads to overprovisioning in all three areas which drives up spend.

Silk can help you by reducing overprovisioning, enabling a shared architecture for economies of scale, and shrinking your footprint with data services and snapshots.

To learn more, [sign up for a demo or get in touch with us](#) if you want a more detailed analysis of your cloud costs.

[Watch the full webinar here.](#)

About Silk

The Silk Cloud DB Virtualization Platform gives demanding workloads 10x faster performance on the cloud compared to native cloud alone. It is a virtualization layer that sits between the underlying cloud infrastructure and customers' workloads. Without refactoring, workloads such as Oracle, Microsoft SQL Server, and industry-specific applications can move onto the GCP and Azure cloud while massively improving user experience. Industry leaders in e-commerce, software publishing, FinTech, and healthcare trust Silk with their mission-critical workloads to get the ultra-fast speeds their customers demand. Silk is headquartered in Needham, MA.

To learn more, visit <https://silk.us/>.