Price-Performance in Modern Cloud Database Management Systems

McKnight Consulting Group

December 2019

www.mcknightcg.com

Executive Summary

The pace of relational analytical databases deploying in the cloud are at an all-time high. And industry trends indicate that they are poised to expand dramatically in the next few years. The cloud is a disruptive technology, offering elastic scalability vis-à-vis on-premises deployments, enabling faster server deployment and application development, and allowing less costly storage. The cloud enables enterprises to differentiate and innovate with these database systems at a much more rapid pace than was ever possible before. For these reasons and others, many companies have leveraged the cloud to maintain or gain momentum as a company.

The cost profile options for these cloud databases are straightforward if you accept the defaults for simple workload or POC environments. However, it can be enormously expensive and confusing if you seek the best price-performance for more robust, enterprise workloads and configurations.

Initial entry costs and inadequately scoped POC environments can artificially lower the true costs of jumping into a cloud data warehouse environment. Cost predictability and certainty only happens when the entire picture of a production data warehouse environment is considered; all workloads, a true concurrency profile, an accurate assessment of users and a consideration of the durations of process execution.

Architects and data warehouse owners must do their homework to make the right decision. With data warehouses, it is a matter of understanding the ways they scale, handle performance issues and concurrency. DW's may look great at the low end during vendor-crafted POCs, but in reality implementations can get very expensive if deployed at production scale. Some organizations have seen upwards of 3-5X higher costs than originally projected. This has forced some deployments to different solutions - either on-premises or a different cloud platform. Time wasted. Budgets impacted. Migrations disrupted. It doesn't have to be this way.

Data professionals who used to be valued for tuning queries are now valued for tuning costs. Information is power in this new reality.

Entry costs and POCs artificially lower the true TCO. Wise database buyers today should evaluate all of the pricing options, with a keen eye on performance. Major cloud vendors also offer a number of cost savings and options such as reserved instances, spot instances, bring-your-own-licenses, containers, and serverless computing.

Reserved instance pricing can usually only be procured with annual commitments and is most cost-effective when paid in full upfront. However, long commitments may not fit the trial scenario.

The goal of this paper is to provide some of that information, looking at factors in cloud data platform pricing – such as scope, scale, deployment, etc. - and how to determine the ultimate success metric when it comes to making a decision on a cloud data warehouse deployment – price-performance.

Table of Contents

Components of a Cloud Data Warehouse Platform

Moving an analytic workload to the cloud or deploying your new workload in the cloud can be a large undertaking. The cloud is ready. To get your team ready, you need to understand a few things and consider how the cloud changes your approach to the following:

Understanding How Data Size Impacts Nodes

A node is not a node is not a node, especially when you cross cloud platforms and seek equivalency. The following are how different cloud analytics platforms define "node":

- Azure SQL Data Warehouse is scaled by Data Warehouse Units (DWUs) which are bundled combinations of CPU, memory, and I/O. According to Microsoft, DWUs are "abstract, normalized measures of compute resources and performance.[1]"
- Amazon Redshift uses EC2-like instances with tightly-coupled compute and storage nodes which is a "node" in a more conventional sense
- Snowflake "nodes" are loosely defined as a measure of virtual compute resources. Their architecture is described as "a hybrid of traditional shared-disk database architectures and shared-nothing database architectures.[2]" Thus, it is difficult to infer what a "node" actually is.
- Google BigQuery does not use the concept of a node at all, but instead refers to "slots" as "a unit of computational capacity required to execute SQL queries,[3]" which is also a vague and abstract concept to the average user.

The inevitable configuration differences when comparing platforms will be counterbalanced by price-performance calculations.

1 https://docs.microsoft.com/en-us/azure/sql-data-warehouse/what-is-a-data-warehouse-unit-dwu-cdwu
2 https://docs.snowflake.net/manuals/user-guide/intro-key-concepts.html
3 https://cloud.google.com/bigquery/docs/slots

Data size affects the number of nodes in the configuration, especially the storage nodes. Data size is booming due to a variety of factors, but mainly because success begets success with data. As the initial data generates profitable use and the platform proves able to handle the workload, it is a matter of time until new uses leverage the data and add their different data requirements to the warehouse. And using detailed data in conjunction with summary data is important for effective decision making, further contributing to data overload.

Seldom is it feasible to delete, or otherwise render inaccessible, older data. Plan on data simply accumulating ad infinitum in the warehouse. Plan on loading all the historical data you have to seed the platform with as well.

Companies are also realizing the usefulness of data that is generated outside of their confines – so called third-party data. It is no longer difficult or untested to “subscribe” to external data feeds to augment internally generated data. Marketing departments in particular have grown in their sophistication to deal with all kinds of data and the more, the better.

> The cost profile options for these cloud databases are straightforward if you accept the defaults for simple workload or POC environments. However, it can be enormously expensive and confusing if you seek the best price-performance for more robust, enterprise workloads and configurations.

Separating Storage and Compute

Although not a pricing component, the ability of the platform to separate storage and compute contributes to node specification and ultimately spend very directly.

Taking a cue from Hadoop clusters, many leading cloud analytic databases have the ability to independently scale compute and storage through specification of nodes with emphasis on one aspect over the other. This alleviates compromise in resource allocation and minimizes costly unused resources.

When compute and storage are separated, you truly only pay for what you use when you use it. Compute and Storage are independently priced and made available. Legacy on-premises high performance compute software assumes those are all tightly coupled. In this approach, you are forced to pay for compute even when you are not using it just to get space to store data, and vice-versa.

Many separate storage from the compute unit. This enables the platform to scale storage capacity and compute resources independently. This capability adjusts to various workload demands, offering potential cost savings when demand is low. It may also be interesting that you are able to pause and resume compute billing, where only storage is billed during the paused time.

For many, you pay for compute resources as a function of time. The hourly rate can vary slightly by region. Choose the lowest found hourly rate per node for the service level needed. Also, you need to add the separate storage charge to store the terabytes of data (compressed). This is also expressed as per hour, although it is substantially less than compute. Alternatively, some cloud vendors have consumption-based pricing models, where instead of paying by the hour, you pay by the byte processed.

> Maximum concurrency can be difficult to calculate because it is a function of the number of queries, the submitted queries' execution plan, the size of the warehouse, and the maximum number of multi-cluster setting. In our experience some platforms only run 6 concurrent simple scan queries (SELECT with a single column filter WHERE clause) before starting to queue, at least until another level of enterprise features are purchased.

Impact of Concurrency Demand on Cloud Deployment

Concurrency is affected by a number of factors and takes on many forms. **The most common operation that impacts database operations is when queueing occurs due to a number of users performing ad-hoc analysis at the same time.**

Concurrency issues can manifest as slow performance, uneven performance or even query timeout. If the database has concurrency limitations, designing around them is difficult at best, and limiting to effective data usage. This could be a dilemma for the DBA. Does the DBA make the compute clusters larger to improve performance, or is it a matter of allowing the clusters to scale out for concurrency?

Cost and Commitment

“Pay for what you use”, the motto of the cloud, is great for incubating ideas and getting started. Once you think about production workloads, savings are generally offered for longer-term commitments, which give the vendor some revenue assurance. Some models, like reserved pricing can be procured with 1 or 3-year commitments and is cheapest when paid in full upfront. Know the potential of the commitment discount up front during the price-performance testing and assessment.

Total Cost of Ownership is More Than Just Cloud Costs

Of course, one of the promises of cloud is the ability to focus on core competencies and your business rather than dealing with machines, software patching, hardware failure, elongated procurement cycles, etc. As companies adopt the cloud, the responsibility of certain functions shifts to the cloud vendor. These necessary items are now being rented instead of owned, but it does not mean they do not need your oversight, nor does it mean your staffing for hardware and software management is zero.

In the transition to cloud for the analytic database, a comfort level will need to be attained and monitoring will be needed for the previously in-house managed infrastructure, but furthermore the database, with certain features, can help minimize staffing bloat as a result of moving to the cloud.

Cheaper prices for roughly equivalent hardware can easily be counterbalanced when the platform is lacking features. For example, you may get less storage in one solution and another may have additional cost for I/O plus backup.

Autonomous Administration

Ease of configurability and overall simplicity, that is both easy to administer and flexible in handling almost any usage pattern is an impact on overall costs as well as storage and compute resources.

The platform should be really pushing forward with automating repetitive tasks. Leveraging automation, an enterprise relinquishes granular control while enabling resources to focus on other things. However, for organizations that have the expertise, granular control may still bear optimization fruit.

While no one wants an over-engineered system, there is only upside to having indexes and tuning at your disposal. Free performance features are the sign of a mature database. They enable service level commitments and workloads that cannot not be achieved otherwise. No tuning is often great for marketing, but impractical in the real world and not in the company's best interest.

Set up should be simple, asking basic information such as selecting the storage and CPU configuration.

However, at the same time, the platform should provide tuning capabilities such as indexes, should you need them. The only alternative to having tuning capabilities is living with the performance you receive without them or increasing the specification of or number of nodes – and cost.

Lack of Platform Features Leads to Increased Configuration and Management

Naturally, the lack of features that leads to coding workarounds, or non-intuitive implementations that create grist in understanding also contributes to long-term administration cost. The list of features and functions that will need to be implemented one way or another in an enterprise data platform is long.

A platform lacking stored procedures, referential integrity and uniqueness capabilities imposes tremendous cost on staffing a project. Having these available is only upside.

Also, implementations are de-risked with implementations of mission critical options for backup and disaster recovery, which typically includes a standby database. Being fully ANSI-SQL compliant allows users familiar with SQL (which are many) to be very comfortable using the environment.

Cloud Analytics Platform Pricing

Data analytics platforms load, store, and analyze volumes of data at high speed, providing timely insights to businesses. Data-driven organizations leverage this data, for example, for advanced analysis to market new promotions, operational analytics to drive efficiency, or for predictive analytics to evaluate credit risk and detect fraud. Customers are leveraging a mix of relational analytical databases and data warehouses to gain analytic insights.

Despite apparent similarities in analytical relational database in the cloud offerings, there are some distinct differences in how the platforms are priced, and ultimately their total cost of ownership.

There is a misconception that cloud data warehouse is always cheaper than on-premises. It's only cheaper if you have the right database and handle your database business effectively. You must understand the differences between consumption (pay as you go) and reserve pricing (all you can eat) and test performance to know the extent of resources which will be consumed.

Performance

It is primarily the performance of the data access that constitutes the success of a workload. Performance can be engineered (and it always must be to some degree), but primarily we give performance a huge step forward with correct workload-platform allocation.

Price-Performance: The Ultimate Metric

The inevitable configuration differences when comparing platforms will be counterbalanced by price-performance calculations.

The price-performance metric is dollars per query-hour ($/query-hour). This is defined as the normalized cost of running the workload on the cloud platform. It is calculated by multiplying the rate (expressed in dollars) offered by the cloud platform vendor times the number of computation nodes used in the cluster and by dividing this amount by the aggregate total of the best execution times for each query (expressed in hours).

To determine pricing, each platform has different options. Buyers should be aware of all their pricing options, not just the evident ones.

Some also have reserved instance pricing, which can be substantially cheaper than on-demand pricing. However, reserved instance pricing can only be procured with 1- or 3-year commitments and is cheapest when paid in full upfront. Long commitments go counter to the agile nature of today's projects.

For some, you pay for compute resources as a function of time, but you also choose the hourly rate based on certain enterprise features you need. For the lowest level of support, there is a rate per node per hour. If you want, for example, "multi-cluster" (automatic scale out of additional clusters), enterprise-level security and more support, there is a different rate per node per hour for that. It goes up from there.

With another, you pay for bytes processed and the underlying architecture is unknown, and the environment is scaled automatically. You would multiply the terabytes of data by the on-demand dollars per terabyte pricing. There is also a cost-per-hour flat rate where you would need to calculate how long it would take to run your queries to completion.

Some nodes have alternative names. One "unit", for example, is 64 GB of memory, 8vCPU, and 1 TB of disk.

A "slot" is a logical unit of compute measure (CPU, etc.). This could be an evolving unit of measure where hardware is incremented in the cloud. A 2019 one of these can be underpowered compared to a 2022 one of these, and this makes equivalency with competitive clouds and platforms, and establishing price equivalency over time, difficult.

Entry costs and POCs can artificially lower the true TCO. Wise database buyers today should evaluate all pricing options, with a keen eye on performance. Customers need to analyze current workloads, performance and concurrency and project those into realistic pricing in alternative platforms. Major cloud vendors also offer a number of cost savings and options such as reserved instances, spot instances, bring-your-own-licenses, containers, and serverless computing. Reserved instance pricing can usually only be procured with annual commitments and is most cost-effective when paid in full upfront. However, long commitments may not fit the trial scenario.

"Gotcha" Scenarios

Some cloud analytics platforms have features that appear beneficial and desirable. However, there are some "gotchas" or hidden downsides to those features many organizations do not realize until they have deployed these platforms into production and experienced them firsthand.

Scale-Out Impacts Cost

Easy scale-out is heralded as a win-win feature in cloud analytics. Performance-based scale out occurs when a database cluster does not have enough resources to handle a workload with multi-user concurrency. In that event, a DBA or system administrator (for manual scale-out) or the system itself (automatic) deploys an additional cluster or additional compute nodes to help alleviate the pressure brought on by high user concurrency. The extra compute resources will run for as long as they are needed to finish the heavy workload period, and then be spun down by an administrator manually or automatically by the system. In theory, this seems like a great feature to have. There is, however, a glaring downside—cost.

The additional compute resources come, of course, at a cost. If an additional identical cluster is deployed to handle the additional user queries, the cost doubles for the time period the additional cluster is up and running. If additional compute nodes are added, the cost of the individual nodes gets added to the cloud fees. “Only pay for what you need” seems attractive, but we wanted to shed light on how scale-out can significantly impact, or even breaks, a Capex budget.

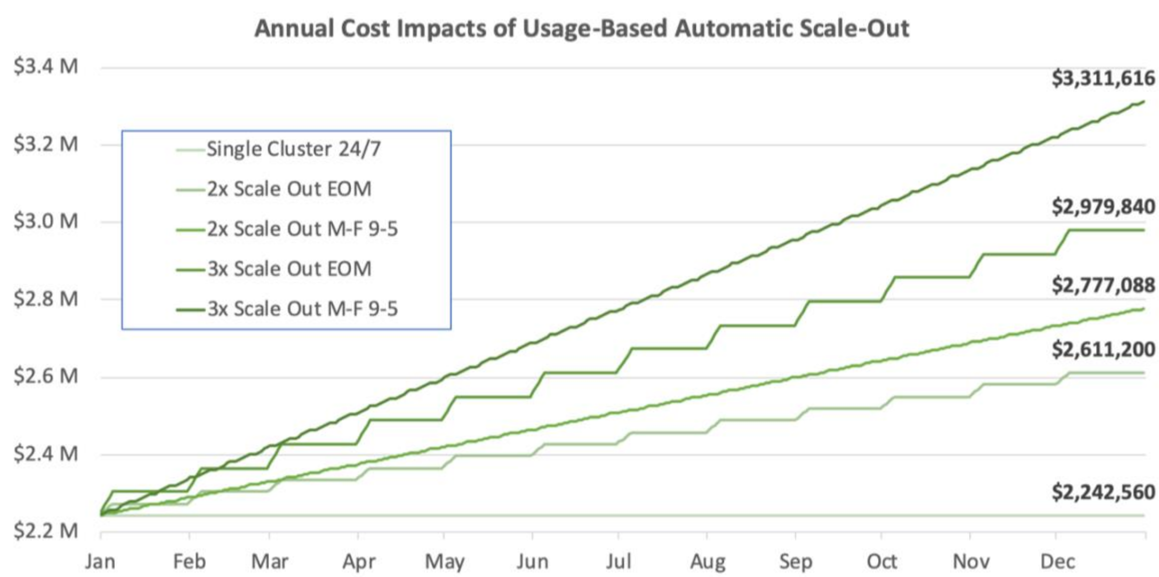
The chart below illustrates a number of scenarios. Let’s say you budget $2.2 million for a single cluster to run 24/7 for an entire year (64 compute units at $4.00 each per hour). The flat line on the bottom of the chart represents this “annual budget” based on this single cluster running 24/7/365.

However, after you go into production, you realize this single cluster is not enough compute power to get through periods of heavy usage where more concurrent users are online submitting queries, so you enable multiple cluster scale-out. Let’s examine a few scenarios.

In one scenario, your Finance department performs some heavy end-of-month processing. They are balancing the general ledger and run some heavy analytical queries against the data warehouse to get financial projections and forecasts. These additional concurrent users and their queries are in addition to the normal day-in-dayout query loads. This scenario is called "EOM" and represents end of month intensive processes that trigger a scale out event every month.

In the second scenario, you discover the analytics load you anticipated and specified during the proof-of-concept phase of your cloud data warehouse development is woefully inadequate—particularly during normal business hours (Monday-Friday 9AM-5PM). This higher concurrency scenario is represented by "M-F 9-5" on the chart and again represents heavy usage triggering the need for scale out.

Notice what happens when you scale out additional clusters for end-of-month or Monday-Friday 9-5. If you have to scale out one additional cluster (2x) for EOM, your annual data warehouse expense jumps to $2.6 million. For M-F 9-5 scale out, it jumps to nearly $2.8 million. If you have to scale two additional clusters (3x), your annual expense explodes to $3 million to cover the EOM processing and $3.3 million to cover heavy Monday-Friday 9-5 usage! This could result in going 167% over budget for the year.



Memory Pressure on Scale-Out Compute

Another "gotcha" with cloud data warehouses, particularly those with scale-out clusters is the performance cost of memory pressure and disk spilling. The performance improvements touted by scale-out platforms can actually backfire in certain scenarios. As we will discuss in the next section, the greatest performance killer to analytical workloads is disk I/O. Thus, whenever a data warehouse does not have enough memory to build a join hash table and keep it in memory, it has to spill it to disk. This is costly in terms of performance, because the DBMS has to do double work writing, sorting, and reading the hash table information all on disk—rather than in memory. Get enough concurrent users issuing such queries, and you will see near gridlock conditions.

With an automatic scale-out warehouse, one might be tempted to "save" on costs by saying, "if we need a large-sized cluster during periods of heavy concurrent usage, but only a medium-sized cluster during light usage, then we can just provision a medium-sized cluster and let it scale up to two medium clusters during the busy hours to handle the higher concurrency." Let's see what this means.

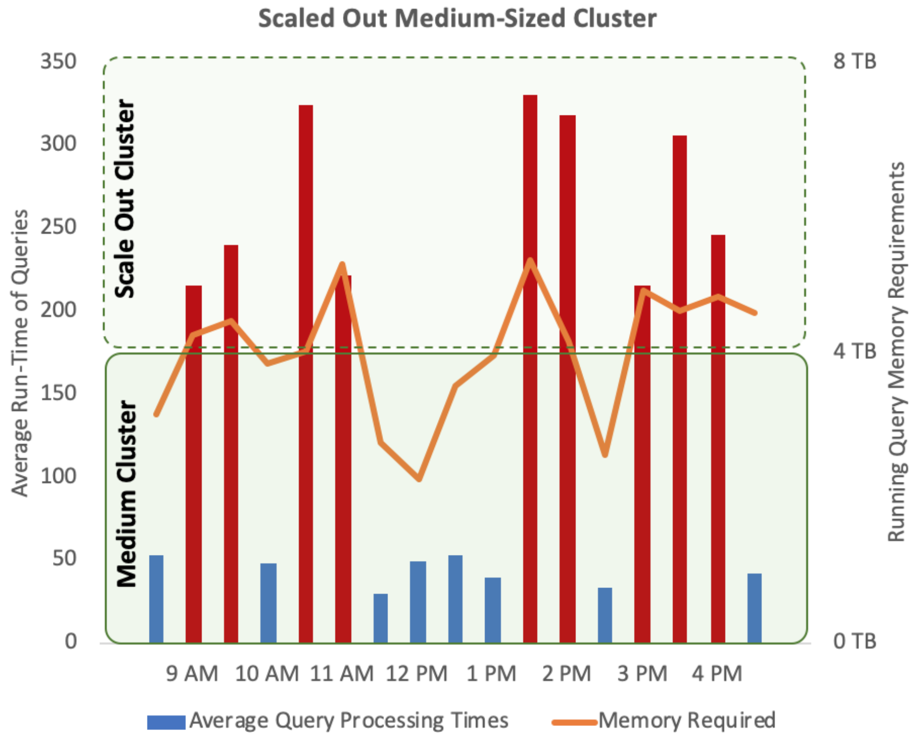
Let's say for the sake of discussion that a medium-sized cluster has 32 CPUs and 256GB of RAM. A large-sized cluster has 64 CPUs and 512GB of RAM. On paper, it looks like two (2) medium-sized clusters running in parallel equals one (1) large-sized standalone cluster. The problem is this is not the case. A medium-sized cluster only has 256GB of RAM.

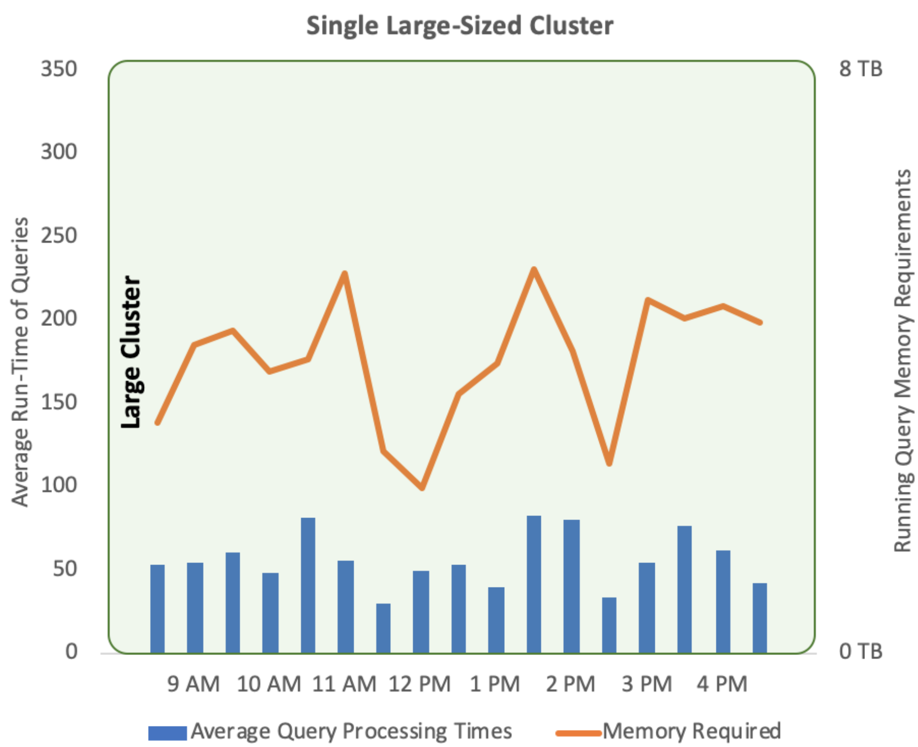
Say a single query involving a large multi-terabyte table with a JOIN needs 300GB of RAM for its hash table. The two medium-sized clusters are still separated by workloads. You cannot split a single query job over two clusters—only concurrent user workloads can be offloaded onto the second cluster. Thus, the query with the large JOIN would spill to disk on the medium-sized cluster, but run all in memory on the standalone large-sized cluster.

In the charts below, this scenario is played out. This depicts the impacts of scaling out a medium-sized cluster (with 4TB of memory) versus running a standalone large-sized cluster (with 8TB of memory). The chart shows the query activity over a given workday from 9:00AM to 5:00PM. The orange line shows the memory required by the active workloads. Whenever the orange line exceeds the 4TB memory allocated by the medium-sized cluster, there is disk spilling, even though a second scaled-out medium cluster is up and running and carrying some of the users' workloads. In our benchmarking experience, disk spilling can cause a 4x or more slowdown. In our example, the average query response times go from around 50 seconds to sometimes over 5 minutes. In the second chart, we show a large-sized cluster with 8TB of memory and everything runs smoothly with no disk spilling and great performance.

The bottom line is that scaling out does not affect performance. Scaling out only helps with higher concurrency, and it does not matter the size of the cluster. Scaling up and making the clusters larger is the only way to affect the performance of individual queries.

Memory Pressure and Disk Spilling Impact on Scaled Out Clusters





Faster I/O is Expensive

Any database administrator will tell you one of the top performance killers in analytical engines is disk input/output (I/O). Disk reads are some of the most expensive resource constraints and technical bottlenecks in analytical workloads. That's why in-memory analytics came into vogue as systems were able to achieve terabytes of RAM. Still, disk I/O is important, because with data warehouse data volumes at 100TB, or even 1 petabyte (PB), persistent disk storage is still a requirement and I/O-bound workloads are still very prevalent as much as we attempt to optimize memory utilization.

Cloud vendors offer varying levels of storage I/O performance. Conventional, on-premise disk performance is typically measured in MB or GB per second of throughput. However, in the cloud, performance is usually measured in input/output operations per second (IOPS). IOPS measures the actual read and write operations that can happen in one second, while throughput is the data transfer rate to and from the storage device. While they measure different things, they typically correlate. Throughput is an older measure, while IOPS is considered a more contemporary performance metric.

In the cloud, storage is offered by media type—solid state (SSD) or spinning hard drive (HDD). SSD is also offered in tiers. Azure has Standard and Premium. AWS has general purpose (gp2) and provisioned (io1) SSD tiers. With AWS, general purpose drives have a set IOPS based on their size in bytes. However, with provisioned storage, one can actually provision a certain amount of IOPS (up to a limit).

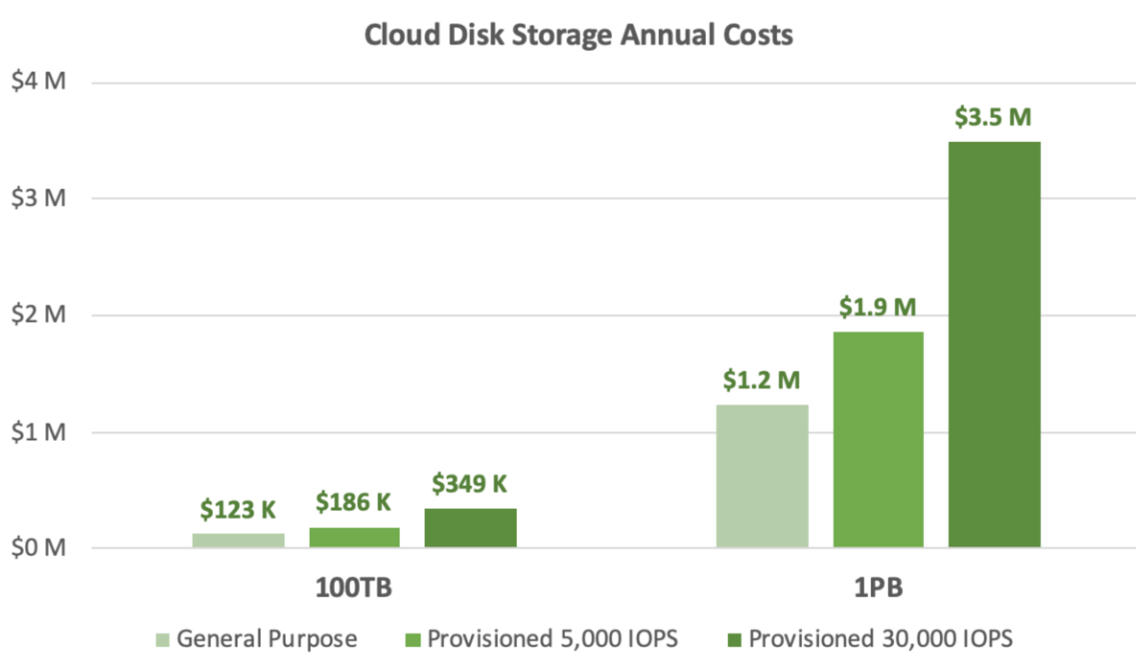
For most data warehouse and analytical uses, you would probably never use general purpose or standard tier storage—except for development or testing instances. The constrained storage bandwidth would choke out performance and slow queries with disk reads to a grind. It would be like rush hour traffic with two lanes closed.

Therefore, in most cases, we would recommend 5,000 IOPS per disk volume at a minimum for the lighter end of warehouse use. For heavy use, 30,000 IOPS would be necessary. This is costly.

In the chart below, we show the annual cost differences in three tiers and IOPS-levels of cloud storage. On the left is the annual cost (using current AWS rates) of 100TB of storage with general purpose, 5,000 provisioned IOPS, and 30,000 provisioned IOPS storage. On the right is the same storage configurations with 1PB of total storage. As you can see, provisioned IOPS can nearly triple storage costs over general purpose or standard storage tiers.

Obviously, one can mitigate these costs by utilizing "hot" and "cold" storage—that is, off-loading seldom used data to a cheaper storage medium (such as AWS S3, Azure Blob, or Google Cloud Storage). In many cases, this cold data can be queried directly from these cloud storage mediums by referencing them as external tables in the data warehouse DBMS.

Thus, storage volume and IOPS can factor greatly into the price-performance equation.



Testing for Price-Performance

Conducting tests for price-performance is an essential step in the platform selection process. It can, however, be extremely difficult to do well. Consider the following aspects of creating a fair, apples-to-apples, budget-informative test:

- Query performance
- Load performance
- Query performance with concurrency
- Ease of use
- Who are the competitors/release numbers
- Queries, Schema, Data
- Scale
- Query Cut-Off
- Number of runs/cache
- Number of nodes
- Tuning allowed
- Vendor Involvement
- Any free third party, SaaS, or on-demand software
- Any not-free third party, SaaS, or on-demand software
- Instance type of nodes
- Measure Price/Performance!
    - Scale-out Cost
    - Memory Pressure Cost
    - Storage Tier Cost

Conclusion

Without a doubt, the cloud offers organizations many benefits to analytical workloads and their performance with scalability and elasticity. However, ***buyer beware***. The total cost of ownership of cloud analytics platforms scales up too. Demand for analytics at your company will only increase in the coming years.

Hardware (CPU, memory, and input/output) is often the biggest performance bottleneck of a database management system. Most cloud analytical products scale hardware in powers of 2. However, costs for these scale-ups and scale-outs usually double as well. In more conventionally-priced systems, you can add more memory here or more CPU there at a more fractional cost. As demand for analytics grows, more and more concurrent users will bring increasing demands with more and more complex workloads. With a cloud on-demand pricing model, greater demand will drive costs higher and faster than a comparable conventionally-priced platform. Remember, the cloud marketing message “only pay for what you use” is a two-sided coin. There’s certainly an upside if demand is lower than predicted, but the downside is when demand exceeds expectation. And the cost ceiling can be quite high.

Instead, your analytics platform should deliver ***value*** to your organization. All told, the true gauge of value is price-performance. Thus, we recommend that you demand reliable performance at a predictable price from your analytical platform—whether in the cloud or not.

Sponsored by

teradata.

About Teradata

Teradata is the connected multi-cloud data platform for enterprise analytics company. Our enterprise analytics solve business challenges from start to scale. Only Teradata gives you the flexibility to handle the massive and mixed data workloads of the future, today.

The Teradata Vantage architecture is cloud native, delivered as-a-service, and built on an open ecosystem. These design features make Vantage the ideal platform to optimize price performance in a multi-cloud environment.

Our experience working with thousands of customers and partners around the world, across a wide range of verticals and industries, makes us the most effective platform for delivering business outcomes and unlocking unlimited value by turning data into your greatest asset. Learn more at Teradata.com.