WIKIBON

Tremendous Cost-Performance Advantages of OCI's Flexible Instances

January 2024

Written by: Marc Staimer, Sr. Analyst, Wikibon

Premises

Public cloud compute promised to eliminate or at least mitigate manual tasks while reducing costs for IT infrastructure. For organizations migrating from on-premises IT, the public cloud takes over those tasks via personnel or automation. However, the public cloud does not offer the ability to correctly and accurately right size CPU cores and memory to workloads at a fine-grain level with non-disruptive burstable elasticity.

Instead, public cloud customers generally have to select a hardware shape for each instance workload. Most providers offer various instance categories with dozens of instance families, each with specific instance sizes. Take for example, Amazon Web Services (AWS), as shown below.

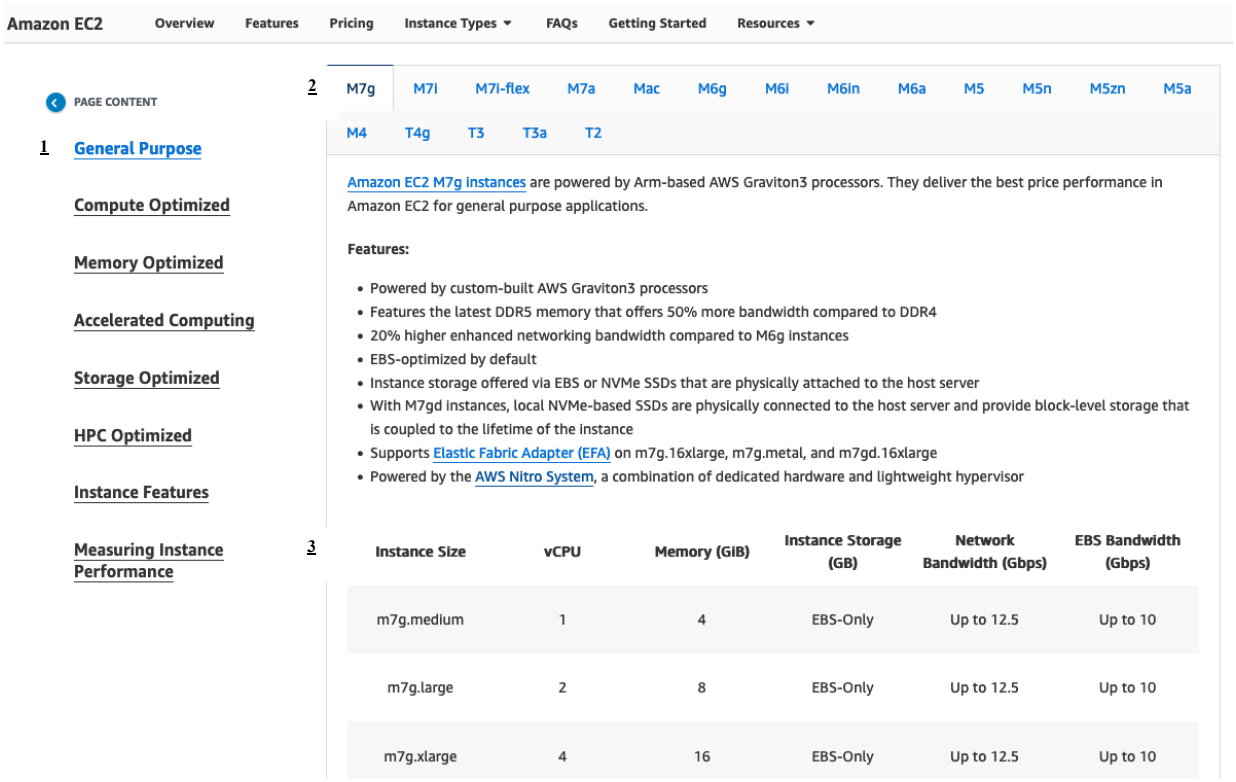


Figure 1. Compute Offerings on AWS Showing (1) Categories, (2) Shape/Instance Families and (3) Shape/Instance Sizes. Source: AWS.

Each shape comes preconfigured with a specific number of vCPUs (each vCPU is half a core) and memory. Processing and memory – in gigabytes (GB) – cannot be sized independently, i.e., the shapes are predetermined and not flexible.

This means that a workload that is heavy on compute and light on memory will commonly have a lot of unused idle memory. Vice versa, workloads that are memory intensive and not compute intensive, will have a lot of idle cores. Put differently, customers are paying for infrastructure they're not using.

It gets worse. Each fixed shape is a coarse grain jump in processing and memory. Therefore, when a workload needs 1 or 2 more vCPUs but the next fixed shape is an additional 4, 8, 16, or more vCPUs, then the unnecessary vCPUs will be idle as will be underutilized memory. Idle cores and idle memory are not free and will raise unnecessary cost.

Instance Size	vCPU	Memory (GiB)	Instance Storage (GB)	Network Bandwidth (Gbps)	EBS Bandwidth (Gbps)
m7g.medium	1	4	EBS-Only	Up to 12.5	Up to 10
m7g.large	2	8	EBS-Only	Up to 12.5	Up to 10
m7g.xlarge	4	16	EBS-Only	Up to 12.5	Up to 10
m7g.2xlarge	8	32	EBS-Only	Up to 15	Up to 10
m7g.4xlarge	16	64	EBS-Only	Up to 15	Up to 10
m7g.8xlarge	32	128	EBS-Only	15	10
m7g.12xlarge	48	192	EBS-Only	22.5	15
m7g.16xlarge	64	256	EBS-Only	30	20

Figure 2. Instance Sizes with vCPU and Memory Capacity. Source: AWS.

The plethora of fixed shapes cannot be altered and moving from one size to another, or one shape family to another, is a disruptive process. It requires a manual change to select another shape, which requires a reboot. Reboots are application and workload disruptive. They are typically scheduled for times that are the least busy, especially for mission-critical and business-critical workloads. In addition, these shapes are not elastic. At least not the way customers assume it to be.

What makes the situation worse for several Cloud Service Provider (CSP) customers is their use of reserved instances. Reserved instances tie them to specific regions, shape families, and even operating systems. The customers typically have to get advanced permissions to reallocate the initial reserved instance purchase. This process complicates and slows down any needed changes.

The change process for fixed shapes is similar to that of changing VM cores and memory for on-premises virtual servers. In either case, the customer can change a VM's shape as long as the VM is offline and they are okay with the application and workload disruption. The cost being in time, money, and risk. This will be covered more in-depth in the next section.

Another big unfulfilled public cloud promise is granular scaling that more closely matches the workload vCPU and memory requirments. In reality, fixed shape instances scale in large chunks of both vCPUs and memory. New public cloud users expect to only pay for what they use. Price-performance much higher than expectations and cost-

performance is worse. Customers are frequently frustrated and unhappy when they realize what they're paying is much more than what they were expecting.

Burstable shapes are AWS' and Azure's answer. However, burstable shapes on these two CSPs do not take advantage of more powerful CPUs, such as 4th Gen AMD EPYC processors. Therefore, they are really meant for lightweight applications where performance is not an issue. More detail about burstable shapes will be covered later in this research document.

Oracle Cloud Infrastructure (OCI) flexible instances are architected to solve these issues of fine granularity, elasticiy, burstability, and cost performance. This research paper looks deeper into problems with public cloud fixed shapes and how OCI flexible instances solve those problems. It then compares all four public cloud vendors in granularity, time and cost efficiencies, burstable elasticity, and cost/performance.

Problems with Public Cloud Fixed Shapes/Instances

Problems with public cloud fixed shape instances include coarse granularity, disruptive elasticity, burstable limitations, and excessive price and cost-performance.

Fixed Shape Instance Coarse Granularity

Fixed shapes have a set number of vCPUs (½ core per vCPU) and memory in gigabytes (GB) per shape. The common rules for these shapes are 4 GB memory per vCPU or 8GB of memory per core. Shape vCPU sizes grow exponetially with 2^n – so 2, 4, 8, 16, 32, 64, and 128. There are some public cloud shapes with 96 and 192 vCPUs.

Scalability of fixed shapes are not fine-grained. This means when a customer just needs a couple of additional vCPUs then the shape allows, they have to go the next larger shape. That will be a considerable jump in cost especially at the higher end of the shapes.

Instance name	On-Demand hourly rate	% Increase from previous instance	vCPU	Memory (GiB)
m7g.medium	$0.0408	NA	1	4
m7g.large	$0.0816	100%	2	8
m7g.xlarge	$0.1632	100%	4	16
m7g.2xlarge	$0.3264	100%	8	32
m7g.4xlarge	$0.6528	100%	16	64
m7g.8xlarge	$1.3056	100%	32	128
m7g.12xlarge	$1.9584	50%	48	192
m7g.16xlarge	$2.6112	33%	64	256

Table 1. Instance sizes, vCPU, Memory, On-demand Hourly Rate, and % Increase from One Shape to the Next. Source: AWS.

Take the example of a customer instance utilizing a 32 vCPU shape but needs 34 vCPUs. They would have to upgrade the shape tp 64 vCPUs leaving them to pay for 30 vCPUs sitting idle. That shape also doubles the memory which they are unlikely to need or use.

Another problem with fixed shapes is that they're typically on shared hardware that is oversubscribed. Oversubscription means there are times where the customer does not get what they are paying for. AWS does offer

dedicated fixed shape instances, but at a 50% premium – 36% in some of the much larger fixed shape instance use cases.

The takeaway: fixed shape instances are coarsely granular, wasteful, disruptive, and quite costly.

Disruptive elasticity

Think of the rubber band analogy. Compute elasticity is supposed to mean expanding and contracting automatically in response to the application workload's changing needs. But that's not what it means for many CSPs. Moving from one fixed shape to another requires a disruptive, short-term outage – instance reboot. Few IT organizations will or can tolerate an application outage during business hours. In the 7x24x365 economy with remote workers all over the world, all hours are business hours. Disruptions create problems, especially for mission-critical and business-critical application instances. Disruptions need to be scheduled, which delays implementation. That delay can be hours, days, weeks, or even months when the application instance can't tolerate downtime.

Until that fixed-shape instance is upgraded, application response times will suffer. When application response times increase beyond 2 seconds, productivity plummets. Time-to-market, time-to-actionable-insights, and time-to-revenue slows. External-facing applications lose potential customers.

The IT workaround is to oversubscribe – i.e., pay for the next largest shape upfront to prevent having an application disruption. This is similar to how server and storage administrators have been avoiding infrastructure disruptions on-premises for decades. But this workaround is very expensive, and counterintuitive to the cloud's promise of "paying only for what you need." The customer can pay as much as 2x the instance infrastructure they actually require, a solution that's not sustainable in the long-term.

Burstable Limitations

Burstable fixed shape instances have the potential to solve some of the granularity and disruptive elasticity issues. But, using them on many CSPs can be complicated. Customers have to decide how much of the total vCPU processing they will commit to financially. For example, a customer anticipates using approximately 20% on average of the vCPUs in the shape over a full year. That's what they commit to. They will receive credits for all the time below that 20% level up to a maximum number of credits. The credits are then used against those times when they burst above the 20% level.

Although this might sound reasonable, it turns out that most of the burstable services are severely limited. Too many of the CSPs primarily use older, low performance processors for their burstable fixed shapes. AWS for example uses its Graviton2 ARM, 1st Gen AMD EPYC processors, and Intel Xeon Scalable (Skylake, Cascade Lake) processors. To put that in perspective, 4th Gen AMD EPYC processors are up to **8x faster** than 1st Gen AMD EPYC processors, up to **12x cores**, and up to **6x memory**. Burstable shapes on the other processors are similarly handicapped. That's a problem for mission-critical and business-critical applications that absolutely require performance for low application response times and better end-user experiences.

Extensive IBM research discovered sub second response times leads to much greater user productivity, higher morale, lower employee turnover, lower training costs, faster time-to-market, faster time-to-actionable-insights, faster time-to-revenues and unique revenues, lower headcount, and higher profits. IBM research demonstrated this unequivocally in 1982 with their Red Book called, The Economic Value of Rapid Response Time.

> "When an application and its users interact at a pace that ensures that neither has to wait on the other, productivity soars, the cost of the work done on the application's computer infrastructure tumbles, users get more satisfaction from their work, and their quality improves."

On the flip side, response times greater than 2 seconds led users minds to wander, slow down their effectiveness and productivity, leading to poor morale, and more missed deadlines. A more detailed summary of this IBM research can

be found in Appendix A. Make no mistake, subsecond response times postivie impact on productivity, time-to-actionable-insights, time-to-market, and time-to-revenues will completely eclipse any savings from slower infrastructure.

The takeaway here is that most burstable instance services are too constrained for application workloads that are critical to the organization. They're really only effective for secondary or mostly idle applications where processing and memory requirements are pretty low. Unless the burstable instances are using the latest processors, they can't really be used for mission-critical and business-critical applications.

Excessive Price and Cost-Performance

Jumps in fixed shape instance sizes for vCPUs and memory enable idle cloud infrastructure. However, customers will continually pay for cloud infrastructure they do not use or need. This occurs every time they move up from one fixed shape instance to another. That means the price the customer pays is too frequently much higher than it should be.

That's not the only costs these fixed shape instances incur. Other costs include disruption costs, slow response time costs from oversubscribed cloud infrastructure or undersized fixed shape instances. Avoiding oversubscribed unpredictable slowdowns means paying for dedicated fixed shape instances at a significant – approximately 50% - premium on CSPs such as AWS. If the customer uses a region outside the USA, they will also have to pay an 'uplift'. Uplifts increase costs from 10% to 60%, depending on the country the public CSP region is located. Adding these costs to the list price makes the total cost-performance excessively high.

OCI Flexible Instances on 4th Gen AMD EPYC Processors

OCI flexible instances on 4th Gen AMD EPYC processors are specifically crafted to solve each of these problems. It starts with the fact that flexible instances are ***not*** oversubscribed. Next it delivers very fine grain granularity as their shapes are not fixed. Each OCI flexible instance allows right number of OCPUs – 2 vCPUs or 1 full core per OCPU – and memory to be be scaled to workload, 1 OCPU and GB at a time. Scaling of OCPUs and memory are independent of one another.

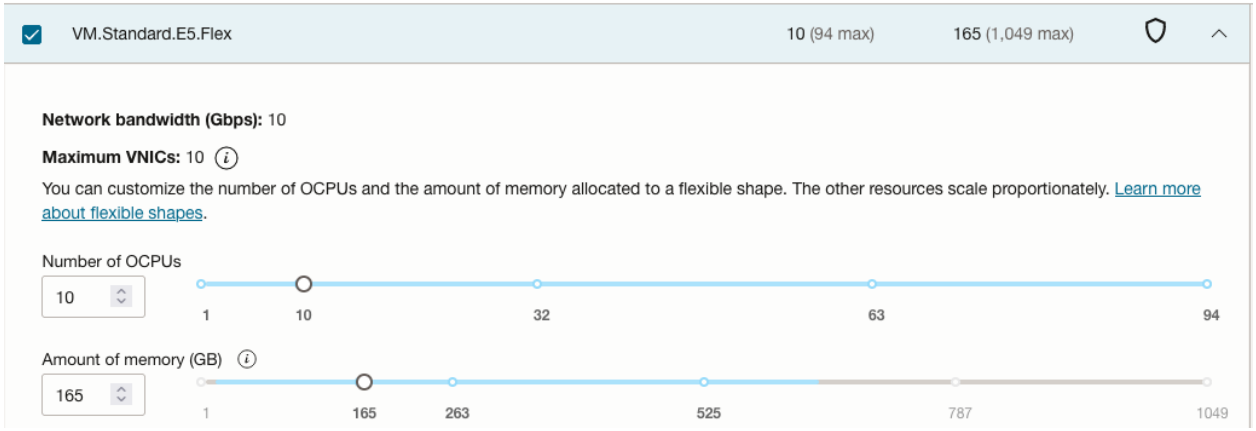


Figure 3. Screenshot of Flexible Instances Slider on OCI Console

Therefore, if an application workload is processing intensive and not memory intensive, the OCI flexible instance is configurable with the necessary OCPUs – cores – and no unnecessary memory. Vice versa, if the application workload is memory intensive but not processing intensive. In that case, the OCI flexible instance is configurable with the necessary memory and no underutilized OCPUs. There is no idle, wasted instance infrastructure and no unnecessary fees for non-utilized cloud infrastructure. Keep in mind these flexible instances require a reboot for any changes to the OCPUs or memory.

Note also that the flexible instance greatly simplifies instance selection for the cloud user. It's no longer needed to choose between various categories such as "General Purpose," "Compute Optimized," "Memory Optimized," and so on. It also eliminates the need for shape families such as M7g, M7l, etc. (see figure 1)

For those application workloads that are processing variable, OCI provides burstable flexible instances. Whereas the standard flexible instance is neither oversubscribed or automatically elastic, burstable flexible instances are both. Because the burstable flexible instance is oversubscribed, there is no guaratee the flexible instance will be able to burst when it is required.

The way OCI burstable flexible instances work starts by specifying the total number of OCPUs and baseline CPU utilization. That baseline utilization can be 12.5% or 50%, which is the minimum CPU commitment. When the application workload needs more it can use up to 100% of the cores provisioned automatically with a maximum of 1 hour of contunous bursting.

Bursting on OCI is straightforward and is tied to the flexible instances CPU resource usage. The flexible instance is credited for the time spent below its commitment and can burst an equal amount of time above the commitment, for a maximum of one hour continuously[1].

These caveats make burstable flexible instances most effective for microservices, development/test environments, continuous integration/continuous delivery (CI/CD) tools, monitoring systems, and static websites. An essential advantage of the OCI burstable instances versus other those on other CSPS is that they are available on all hardware types including the latest, high performance 4th Gen AMD EPYC processors.

There are two other unique OCI flexible instance aspects to keep in mind. The first is that the OCI price per OCPU and GB of memory is the same in every OCI region worldwide. There is no uplift for regions outside of North America. The second is availability. OCI flexible instances capabilities are available and the same for every OCI region.

Summary for OCI flexible instances:

- Fine grain granularity.
- OCPUs and memory are separately configured in units of "1".
- There are no fixed shapes.
- Makes it much simpler to match application workload requirements to cloud infrastructure.
- No wasted, idle, or unused infrastructure.
- No wasted expense.
- Performance.
- Both standard and burstable flexible instances are available on AMD EPYC processors, Intel Processors, and Ampere ARM processors.

[1] OCI burstable flexible instances are somewhat similar but more performant than burstable instances from AWS, Azure, and GCP. These CSPs have commitment levels at the 5%, 10%, 20%, 30%, or 40% of the instance vCPUs on older and slower processors. Credits are earned when usage falls below the commitment. Credits cannot exceed 1 hour of continous bursting. Those credits pay for bursts above the commitment. Application workload needs more than commitment it can use up to 100% of the vCPUs provisioned automatically with a maximum of 1 hour of contunous bursting. Burstable instances from all CSPs are oversubscribed.

Comparing MSRP for OCI flexible instances to AWS, Azure, and Google Cloud Platform (GCP)

Neither AWS nor Azure offer flexible instances like OCI, but GCP does. So to make sure pricing comparisons are fair, all the assumptions will be articulated so anyone can duplicate the work. These assumptions place AWS, Azure, and GCP in the most favorable light possible.

Assumptions:

1	All pricing is based on published MSRP before any discounts.
2	Pricing is for the fixed instances, not burstables • OCI is the only CSP providing burstables on the latest, most performant hardware. • Burstable pricing comparisons would require the lowest common denominator on hardware. • Each cloud provider uses different baseline percentages, credits, and debits. • Makes burstable comparisons extremely difficult to normalize.
3	All pricing is based on the lowest cost regions for AWS, Azure, and GCP. • OCI pricing is the same in all regions. • AWS, Azure, and GCP have significant uplifts for different regions especially those outside of North America.
4	For AWS, Azure, and GCP, the lower price for “on-demand” or “1 yr savings plan” is used. • OCI flexible instance pricing based only on the “on-demand” pricing.
5	Comparisons go to 64 OCPUs/cores. • The curve between the CSPs and OCI stays the same above 64 OCPUs.
6	Memory based on AWS, Azure, & GCP fixed shapes memory rules of ~ 4GB[2] per vCPU – 8GB per OCPU. To put that in perspective with the fixed shapes: • 2 vCPUs = 8GB, 4 vCPUs = 16GB, 8 vCPUs = 32GB, 16vCPUs = 64GB, ... 128vCPUs = 512GB, etc.
7	Shape designations compared for 3rd Gen AMD EPYC processors • AWS – m6a.large through m6a.16xlarge • Azure - Standard_D2as_v5 through Standard_D64as_v5 • GCP – N2D CPU (custom) and N2D Memory (custom) • OCI – E4 CPU and E4 Memory
8	Shape designations compared for 4th Gen AMD EPYC processors • AWS – m7a.large through m7a.16xlarge • Azure - Standard_D2as_v6 through Standard_D64as_v6 • GCP – C3D CPU (custom) and C3D Memory (custom) • OCI – E5 CPU and E5 Memory
9	95% each week’s hours.

[2] Depending on the CSP, memory is either in GBs or GiBs. For the purposes of this research, the differences are not significant

	• Usage running all 5 weekdays and 20 hours per each weekend day.
	  ○ The model can be altered to a variety of different percentages.
	  ○ All percentages apply to every CSP

10	Pricing is based over 12 months (1 year).

Graphing the pricing differences for all four Cloud Service Providers (CSP) utilizing 3rd Gen AMD EPYC processors and 4th Gen AMD EPYC processors clearly demonstrates OCI's flexible instances meaningful cost savings.

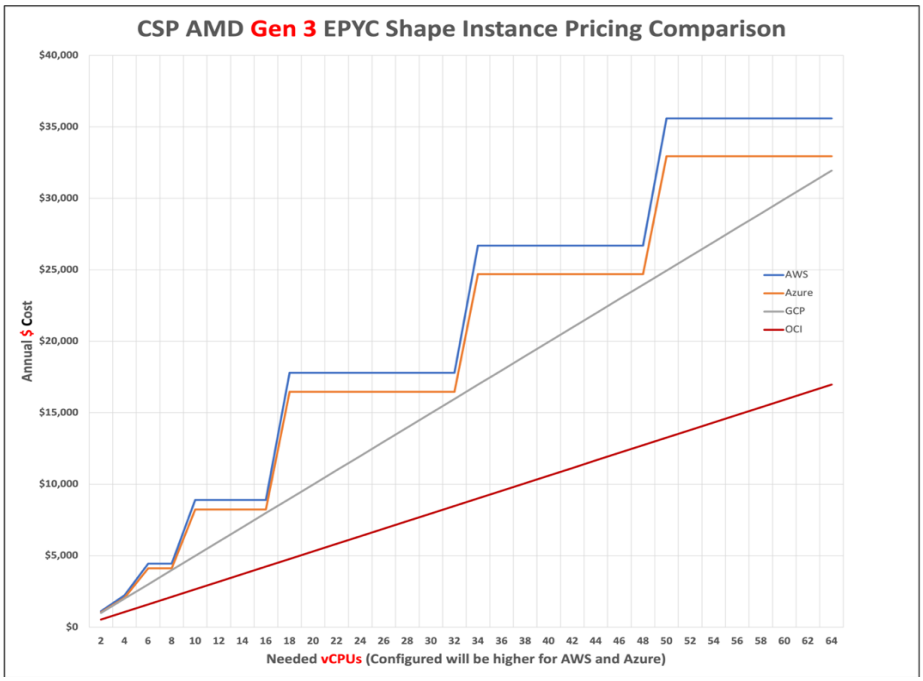


Figure 4: Annual 3rd Gen AMD EPYC Processors Instance Pricing Comparison Between AWS, Azure, GCP, & OCI All Aspects Being Equal

The chart makes obvious to the lay observer that OCI flexible instance pricing with 3rd Gen AMD EPYC is significantly less than AWS fixed shape instances, Azure fixed shape instances, and GCP flexible instances in every configuration. This is pattern holds all the way up to 128 vCPUs. For readability, the scale was limited to 64 vCPUs above.

These instance pricing differences are considerable. Just as importantly, the pricing curve is pretty similar regardless of the amount of weekly hours that are consumed. Running the numbers in all cases reveals OCI flexible instances always come in substantially less.

Keep in mind these are only the costs for a single instance. Most organizations have far more than a single instance. When multiplied by dozens, hundreds, or thousands of application workload instances, the amount of money saved is in a word…huge.

Those savings grow substantially when those instances are outside of CSP North American regions. Remember OCI does not charge an uplift for those regions. AWS, Azure, and GCP all charge significant uplifts that ranges from 1.1x to 1.6x depending on the regional data center location.

Annual savings and their percentages are revealed in the chart below for OCI flexible instances with 3rd Gen AMD EPYC.

OCI Flexible Instance Savings						
vCPUs	AWS		Azure		GCP	
2	$ 297	53%	$ 256	50%	$ 240	48%
4	$ 595	53%	$ 512	50%	$ 481	48%
6	$ 1,448	65%	$ 1,283	62%	$ 721	48%
8	$ 1,190	53%	$ 1,024	50%	$ 961	48%
10	$ 3,155	71%	$ 2,824	69%	$ 1,202	48%
12	$ 2,897	65%	$ 2,565	62%	$ 1,442	48%
14	$ 2,638	59%	$ 2,307	56%	$ 1,682	48%
16	$ 2,380	53%	$ 2,048	50%	$ 1,923	48%
18	$ 6,569	74%	$ 5,907	72%	$ 2,163	48%
20	$ 6,311	71%	$ 5,648	69%	$ 2,403	48%
22	$ 6,052	68%	$ 5,390	65%	$ 2,644	48%
24	$ 5,793	65%	$ 5,131	62%	$ 2,884	48%
26	$ 5,535	62%	$ 4,872	59%	$ 3,124	48%
28	$ 5,276	59%	$ 4,614	56%	$ 3,365	48%
30	$ 5,018	56%	$ 4,355	53%	$ 3,605	48%
32	$ 4,759	53%	$ 4,096	50%	$ 3,845	48%
34	$ 8,949	67%	$ 7,954	64%	$ 4,086	48%
36	$ 8,690	65%	$ 7,696	62%	$ 4,326	48%
38	$ 8,432	63%	$ 7,437	60%	$ 4,566	48%
40	$ 8,173	61%	$ 7,178	58%	$ 4,807	48%
42	$ 7,914	59%	$ 6,920	56%	$ 5,047	48%
44	$ 7,656	57%	$ 6,661	54%	$ 5,287	48%
46	$ 7,397	55%	$ 6,402	52%	$ 5,528	48%
48	$ 7,138	53%	$ 6,144	50%	$ 5,768	48%
50	$ 11,328	64%	$ 10,002	61%	$ 6,008	48%
52	$ 11,070	62%	$ 9,744	59%	$ 6,249	48%
54	$ 10,811	61%	$ 9,485	58%	$ 6,489	48%
56	$ 10,552	59%	$ 9,226	56%	$ 6,729	48%
58	$ 10,294	58%	$ 8,968	54%	$ 6,970	48%
60	$ 10,035	56%	$ 8,709	53%	$ 7,210	48%
62	$ 9,777	55%	$ 8,451	51%	$ 7,450	48%
64	$ 9,518	53%	$ 8,192	50%	$ 7,691	48%
	Mean %	**60%**	Mean %	**57%**	Mean %	**48%**

Table 2: OCI Annual 3rd Gen AMD EPYC Processor Instance Savings vs. AWS, Azure, and GCP

Comparing 4th Gen AMD EPYC processor instances is a bit more difficult. AWS and GCP currently only offer those processors in a couple of regions in the USA. Azure currently only has it in preview in the USA and does not provide any pricing yet. However, assuming Azure uses the same pricing philosophy they've used in the past, it's possible to fairly accurately project their 4th Gen AMD EPYC instance pricing. Based on the published Eastern region pricing of AWS and GCP plus the projected pricing of Azure, OCI flexible instance savings deliver similar savings as illustrated below.

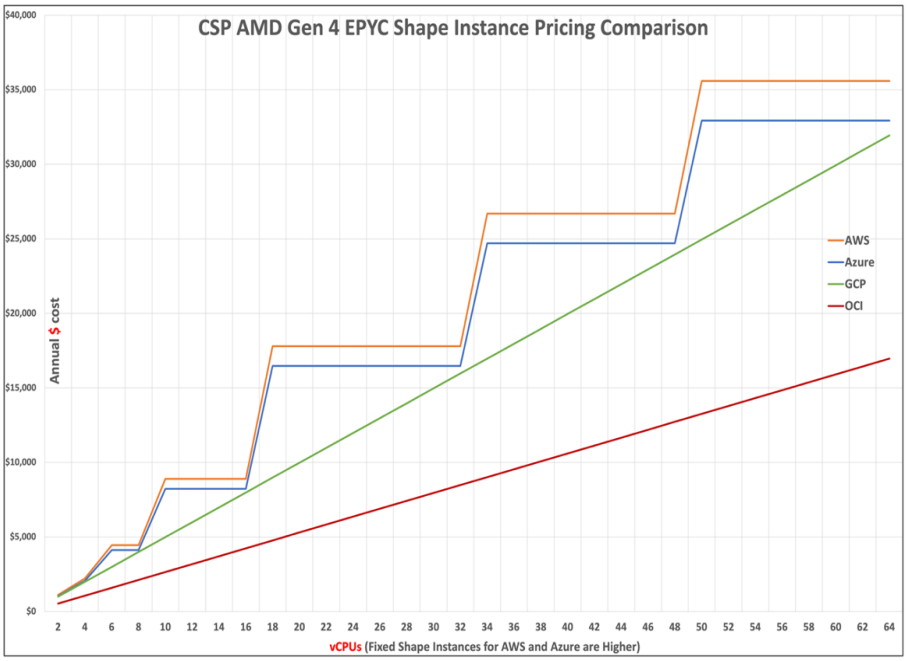


Figure 5: Annual 4th Gen AMD EPYC Instance Pricing Comparison Between AWS, Azure, GCP, & OCI All Aspects Being Equal

Just as with the 3rd Gen AMD EPYC, savings per instance are substantial. Annual savings and their percentages are revealed in the chart below for OCI flexible instances with 4th Gen AMD EPYC.

OCI Flexible Instance Savings						
vCPUs	AWS		Azure		GCP	
2	$ 362	49%	$ 307	44%	$ 191	33%
4	$ 725	49%	$ 613	44%	$ 381	33%
6	$ 1,833	61%	$ 1,611	58%	$ 572	33%
8	$ 1,449	49%	$ 1,227	44%	$ 762	33%
10	$ 4,050	68%	$ 3,605	65%	$ 953	33%
12	$ 3,666	61%	$ 3,221	58%	$ 1,143	33%
14	$ 3,282	55%	$ 2,838	51%	$ 1,334	33%
16	$ 2,898	49%	$ 2,454	44%	$ 1,524	33%
18	$ 8,483	71%	$ 7,594	69%	$ 1,715	33%
20	$ 8,099	68%	$ 7,210	65%	$ 1,905	33%
22	$ 7,715	65%	$ 6,826	62%	$ 2,096	33%
24	$ 7,331	61%	$ 6,443	58%	$ 2,286	33%
26	$ 6,948	58%	$ 6,059	55%	$ 2,477	33%
28	$ 6,564	55%	$ 5,675	51%	$ 2,667	33%
30	$ 6,180	52%	$ 5,291	48%	$ 2,858	33%
32	$ 5,796	49%	$ 4,908	44%	$ 3,048	33%
34	$ 11,381	64%	$ 10,047	61%	$ 3,239	33%
36	$ 10,997	61%	$ 9,663	58%	$ 3,429	33%
38	$ 10,613	59%	$ 9,279	56%	$ 3,620	33%
40	$ 10,230	57%	$ 8,895	54%	$ 3,810	33%
42	$ 9,846	55%	$ 8,512	51%	$ 4,001	33%
44	$ 9,462	53%	$ 8,128	49%	$ 4,192	33%
46	$ 9,078	51%	$ 7,744	47%	$ 4,382	33%
48	$ 8,695	49%	$ 7,360	44%	$ 4,573	33%
50	$ 14,279	60%	$ 12,500	57%	$ 4,763	33%
52	$ 13,896	58%	$ 12,117	55%	$ 4,954	33%
54	$ 13,512	57%	$ 11,733	53%	$ 5,144	33%
56	$ 13,128	55%	$ 11,349	51%	$ 5,335	33%
58	$ 12,744	53%	$ 10,965	50%	$ 5,525	33%
60	$ 12,360	52%	$ 10,582	48%	$ 5,716	33%
62	$ 11,977	50%	$ 10,198	46%	$ 5,906	33%
64	$ 11,593	49%	$ 9,814	44%	$ 6,097	33%
	Mean %	56%	Mean %	53%	Mean %	33%

Table 3: OCI Annual 4th Gen AMD EPYC Instance Savings vs. AWS, Azure, and GCP

Once again these are just the direct cost savings when AWS, Azure, and GCP have their ***most favorable region pricing***. Looking at the savings from real customers who are actual OCI customers shows very substantial savings over what their equivalent usage in other clouds would cost them in order to maintain the same performance, based on shape size.

The names of the OCI customers have been omitted. Starting with a collaboration platform customer saved millions of dollars (USD) based on the data below.

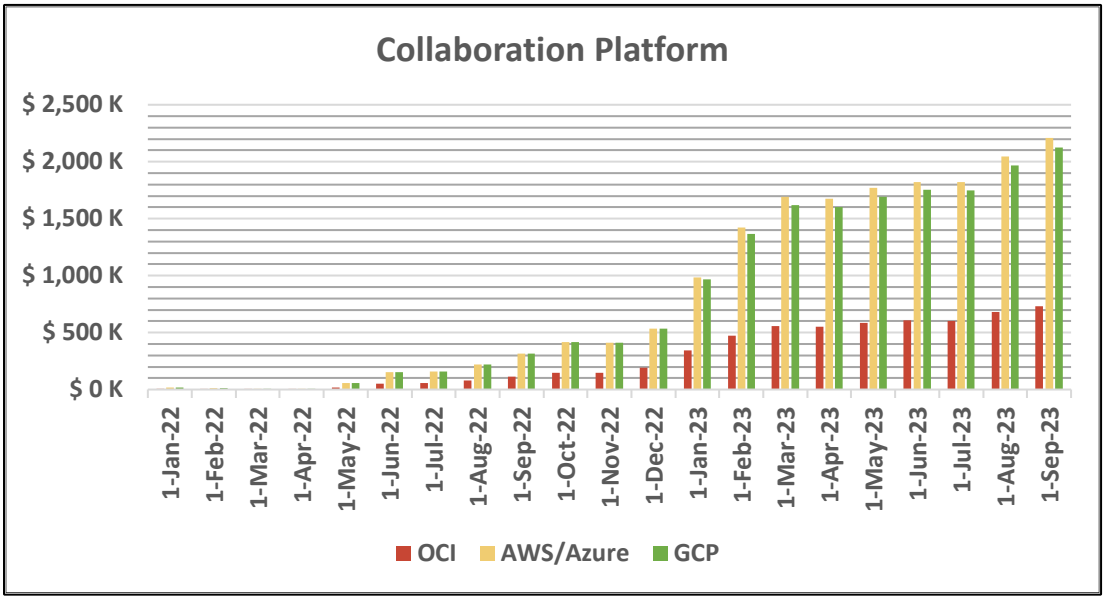


Figure 6: OCI Flexible Instance Savings for Collaboration Platform

An entertainment platform customer saved tens of millions of dollars based on the data below.

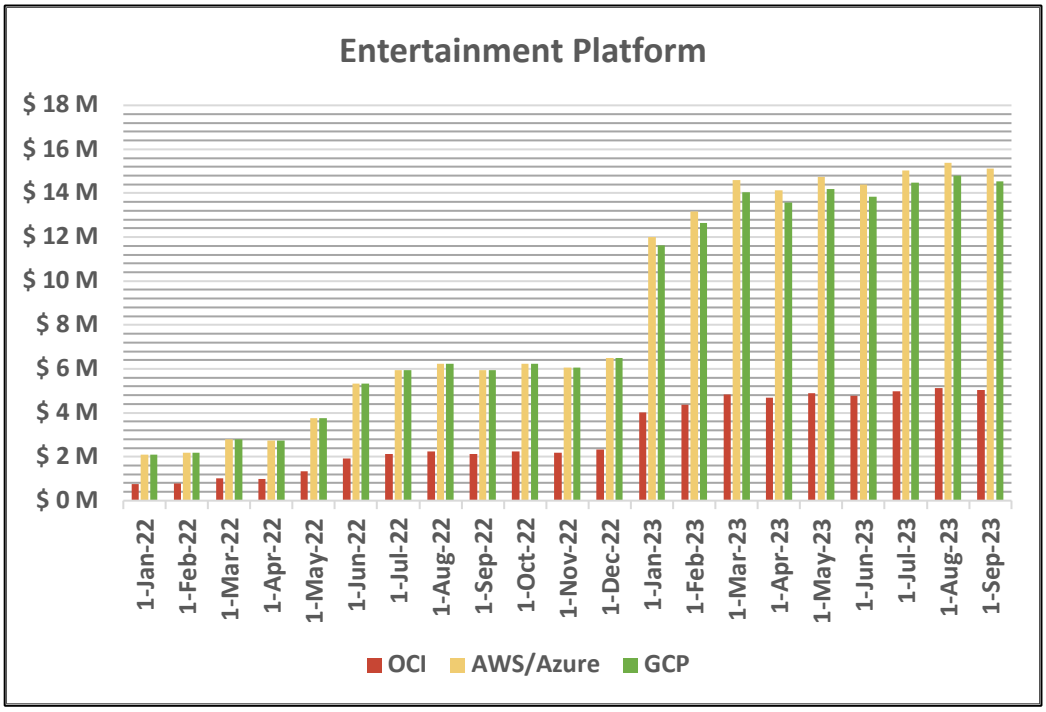


Figure 7: OCI Flexible Instance Savings for Entertainment Platform

A healthcare IT provider customer saved hundreds of thousands of dollars based on the data below.

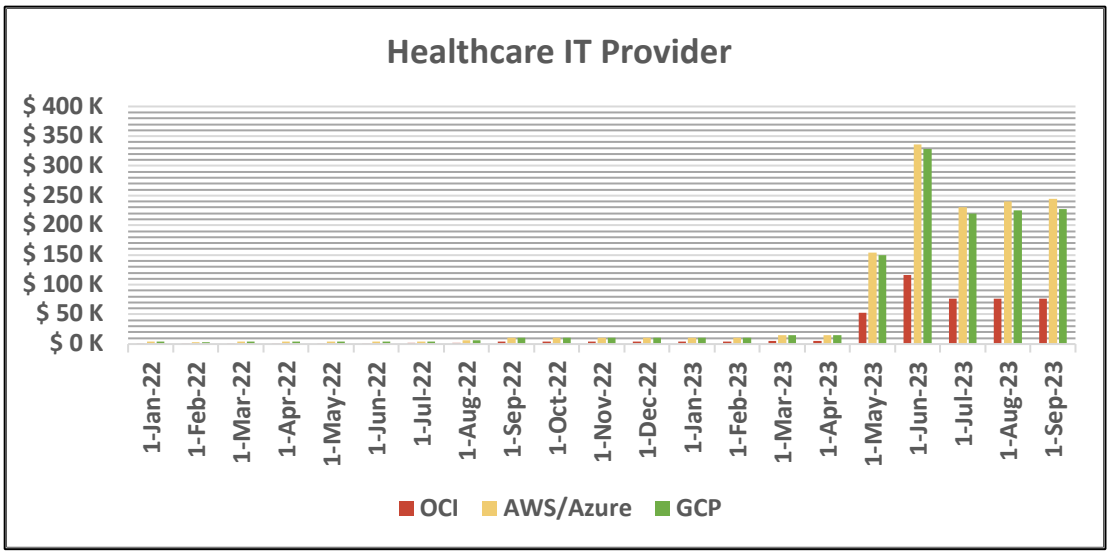


Figure 8: OCI Flexible Instance Savings for Healthcare IT Platform

Even a US state-wide educational system customer saved hundreds of thousands of dollars based on the data below.

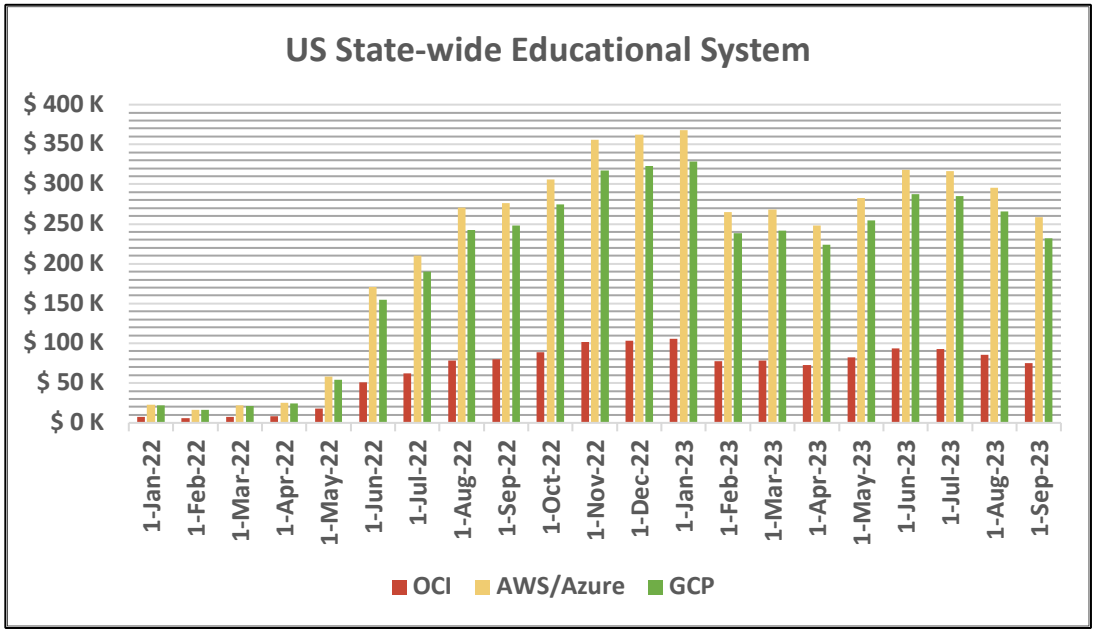


Figure 9: OCI Flexible Instance Savings for US State-wide Educational System

These OCI customers show the magnitude of possible savings from OCI flexible instances.

Conclusion

OCI flexible instances on 3rd and 4th Gen AMD EPYC processors are much more cost effective than AWS and Azure fixed shape instances as well as GCP flexible instances. OCI flexible instances deliver much lower out-of-pocket costs due to lower pricing and much better cost-performance due to right sizing.

OCI flexible instances have better fine-grained granularity than AWS and Azure with individually configured OCPUs and memory. This eliminates the customer's wasted infrastructure cost. As mentioned earlier, annual savings on OCI can be upto millions of dollars (USD) depending on workload characteristics and scale of deployment.

And if the customer wants save even more money and have automated elasticity, OCI offers burstability on both the 3rd and 4th Gen AMD EPYC processor instances. The only tradeoff is that OCI burstables are somewhat oversubscribed. However, the other CSPs do not even offer a burstable service on those modern AMD processors. They use burstable services to extend the life of their older slower processors. And they overscribe all of their instance services.

All of this transforms OCI flexible instances on 3rd and 4th Gen AMD EPYC processors into the best cost-performance for public cloud instances available today.

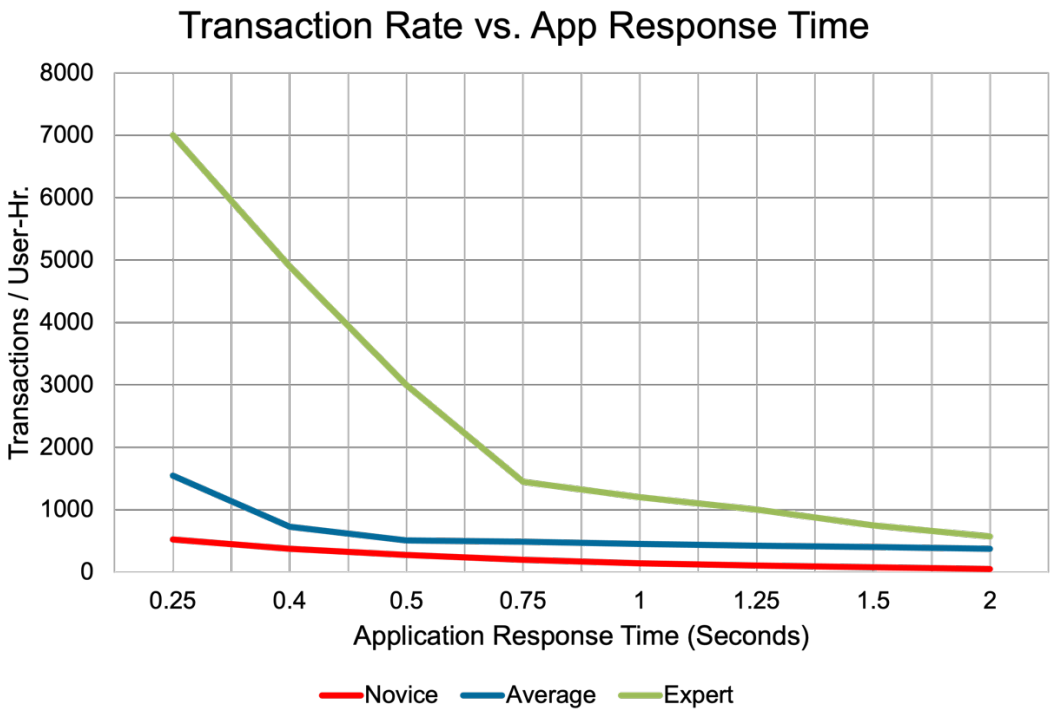
For more information about OCI flexible instances on 3rd and 4th Gen AMD EPYC processors, go to:

- Flexible Instances
- Burstable Instances

Appendix A: Rapid Response Time Economic Value

Calculating performance productivity impact:

- Performance has a substantial and measurable impact on productivity.
  - Response time has a direct correlation on user productivity, quality-of-work, and time-to-market.
  - It was determined that the maximum application response time before user productivity declines precipitously is 3 seconds. Anything over 2 second response times caused user attention to wander.
  - Application response times that are less than 3 seconds promptly increase user productivity, quality-of-work, and time-to-market.
  - Reducing response time to ~ .3 seconds more than doubles productivity versus 2 seconds. Productivity gains are substantially greater depending on the user's level of expertise.



  - Faster response times mean shortened project schedules and higher work quality.
  - ≤ .4 seconds equates into what is called the Doherty Threshold. The Doherty Threshold is when response time becomes addictive whereas > .4 seconds users' attentions begin to stray and productivity begins to decrease rapidly.

App Response Time (Sec)	Transactions per Hr.	Task Time (Min)	Time Saved per Task (Min)	Time Saved per Day
3	180	60	-	-
2	208	51.9	8.1	1h/4m/48s
1	252	42.9	17.1	2h/16m/48s
0.6	279	37.7	22.3	2h/58m/24s
0.3	371	29.1	30.9	4h/7m/12s

  - Determine application response times for each service under consideration.
  - Compare productivity rates.
  - Divide FTE costs by productivity to calculate FTE cost per transaction.

- One alternative is to compare the time required to complete a defined set number of transactions.
- Multiply the time saved by FTE average hourly cost.

Time-to-market revenue acceleration increases top line revenues and bottom-line profits

- Based on current schedules estimate the following:
  - Amount of revenue for each week or month schedule is moved up.
  - Project how much time the reduced application response time performance will accelerate the time-to-market. This can be derived from the increase in productivity based on application response time. If the developers can more than double their productivity, they can more than cut in half the amount of time to complete their project.
  - Apply the projected market growth rate to that revenue for a set period, anywhere from 1 – 10 years. Compare the total revenues to what is would have been had the schedule not been accelerated. The differences are the unique gains. If the database cloud service delays time to market, then the differences are the unrecoverable losses.
  - Example from a large microchip manufacturer:
    - By accelerating delivery of their chip to market by one quarter they were able to realize unique revenues > than $100 million upfront and five times that amount over 3 years.

