cast

2024 Kubernetes Cost Benchmark Report

Analyzing the gap between CPUs and memory provisioned versus utilized for Kubernetes clusters, and the impact it is having on cloud costs.

https://cast.ai

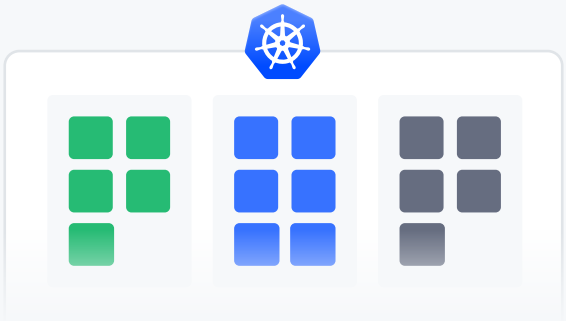
Introduction

Kubernetes has become the de facto container orchestration tool for many organizations. Despite widespread adoption, accurately forecasting the resources Kubernetes applications need remains largely a manual process. As a result, organizations tend to overprovision resources, leading to resource waste, overspending, and operational risks.

Last year, we published our inaugural Kubernetes Cost Benchmark Report to provide insight into Kubernetes cost optimization trends and help companies reduce their cloud costs. The report identified several culprits that cause companies to overspend. Additionally, it shared cost optimization best practices to help DevOps teams reduce cloud costs and maximize return on investment while maintaining performance. Today, we are excited to share our 2024 report.

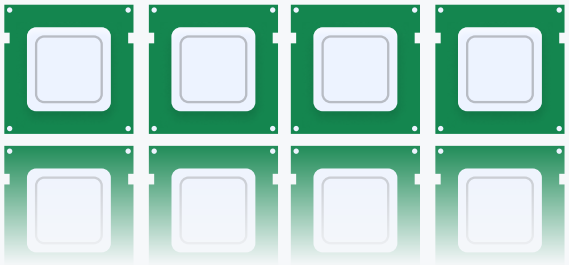
Clusters analyzed

4,000+



Clusters with

50+ CPUs



Methodology

The 2024 Kubernetes Cost Benchmark Report is based on our analysis of 4,000 clusters running on Amazon Web Services (AWS), Google Cloud Platform (GCP), and Microsoft Azure (Azure) between Jan. 1 and Dec. 31, 2023, before they were optimized by our Kubernetes automation platform. This year, we expanded the report to include an analysis of CPU and memory utilization, comparing provisioned, requested, and utilized resources. We excluded clusters with less than 50 CPUs from our analysis.

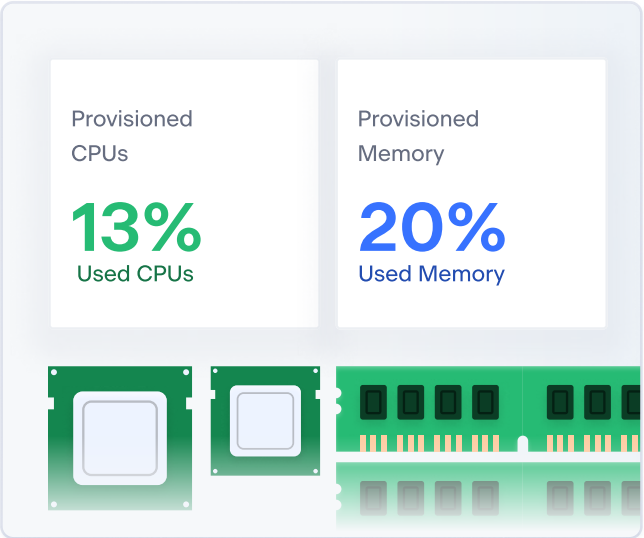
Explore the Report:

1. Key findings →
2. Utilization by the cloud providers →
3. The biggest drivers of overspending →
4. Why avoiding overspending is key →
5. How to cut cloud costs and maintain efficiency at scale →

Key findings

In this year's report, the magnitude of low CPU and memory utilization is striking.

In clusters with 50 CPUs or more, only 13% of the CPUs that were provisioned were utilized, on average. Memory utilization was slightly higher at 20%, on average.

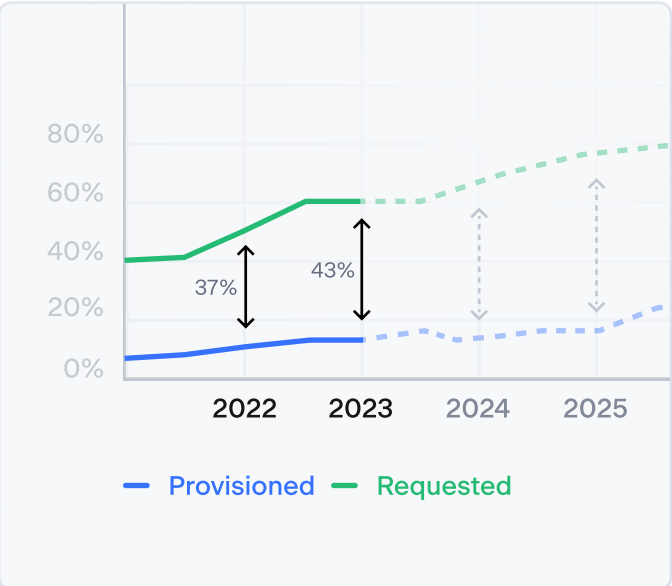


That is likely because DevOps teams generally try to avoid running out of memory. Case in point: The percentage of unutilized memory is virtually identical on AWS, Azure, and GCP, with less than a percentage point difference.

In larger clusters, CPU utilization was only marginally better. In clusters with 1,000 CPUs or more, 17% of provisioned CPUs were utilized. Mega-clusters of 30,000 CPUs or more, which represent less than 1% of our analysis, had a higher utilization rate (44%). That is likely due to the attention they receive from the large DevOps teams managing them.

These findings underscore that many companies running applications on Kubernetes are still in the early stages of optimization. They are grappling with the complexity of managing cloud-native infrastructure and repeating the same costly mistakes that caused overspending in 2022.

The trend appears unlikely to change in the near future given the widening gap between provisioned and requested CPUs between 2022 and 2023 (37% versus 43%). As more companies adopt Kubernetes, cloud waste will likely continue to grow.



Utilization by the cloud providers

Do utilization levels differ by cloud service provider?

To answer that question, we analyzed the gap between CPU and memory capacity provisioned for clusters, versus how much of that capacity was utilized across managed Kubernetes services: Elastic Kubernetes Service (EKS), Azure Kubernetes Service (AKS), and Google Kubernetes Engine (GKE).

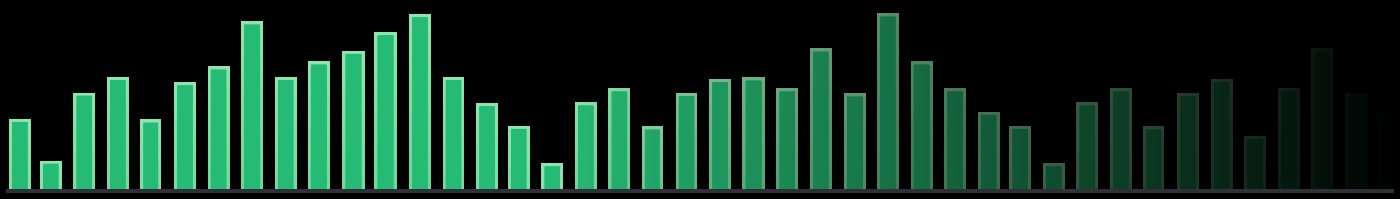
CPU utilization varies little between AWS and Azure; they both share nearly identical utilization rates of 11%. Cloud waste is lower on Google, at 17%. In our samples, GKE clusters tend to be larger than AWS and Azure clusters. GKE users also benefit from custom instances, which use a more precise CPU/Memory ratio selection than the other cloud providers.

For memory, utilization differences are lower across providers:
GCP (18%), AWS (20%), and Azure (22%).

The biggest drivers of overspending

Our analysis uncovered four primary culprits among cloud-native applications:

1. Overprovisioning - Clusters are provisioned with more capacity than they need. The more resources are overprovisioned and left idle in the background, the higher the cloud costs.
2. Unwarranted headroom in pod requests - CPU and memory requests are set higher than what Kubernetes applications actually require, leading to wasted capacity that companies pay for.
3. Low Spot instance usage - Many companies are reluctant to use Spot instances due to concerns over their perceived instability since cloud providers can reclaim them at any time. As a result, there is no noticeable difference between Spot instance usage in 2022 and 2023. Spot instances are still the low-hanging fruit of optimization.
4. For GKE, we see very little use of “custom instance size” - It remains difficult to choose the best CPU and memory ratio, unless the selection of custom instances is dynamic and automated.

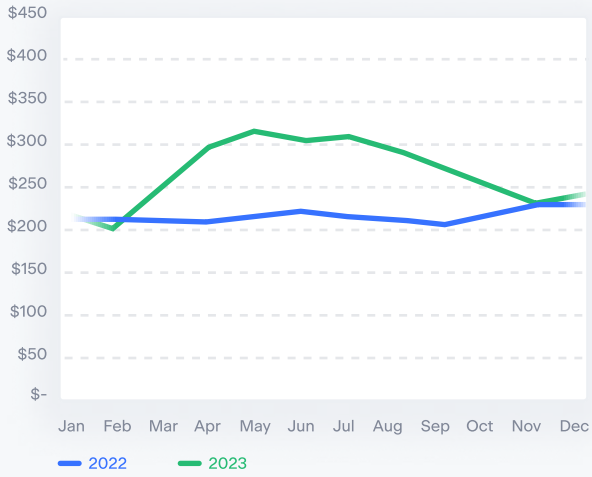


Why avoiding overspending is key

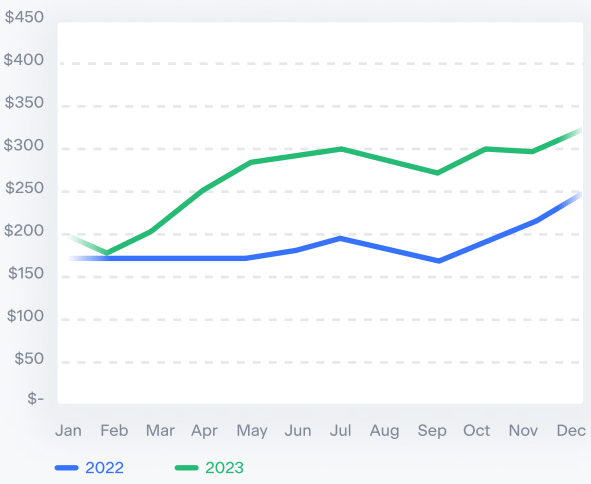
Cloud costs can significantly impact gross margins for businesses across various sectors, leading to excessive financial stress. Reducing overspend is more important than ever due to rising cloud service costs.

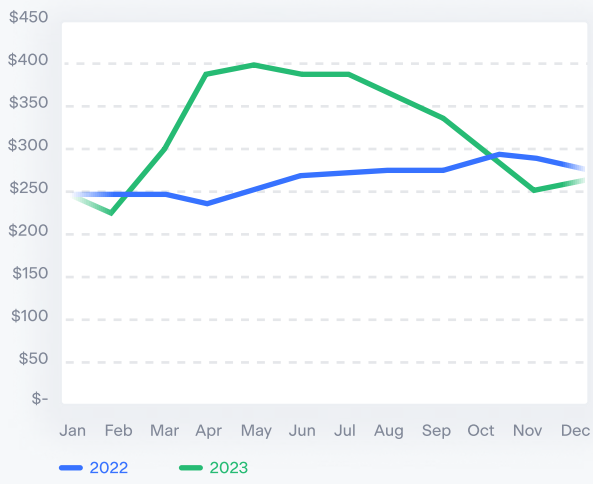
According to Gartner, spending on public cloud services is expected to grow by 20.4% to total $678.8 billion in 2024, up from $563.6 billion in 2023. According to International Data Corp., this will drive 70% of enterprises to become more adept at managing their cloud spending by 2024.

Our in-depth analysis of Spot instance pricing between August 2022 and August 2023 revealed that pricing increased 25% across US AWS regions. For this report, we analyzed Spot instance pricing across the six most popular instances for US-East and US-West regions (excluding Gov regions) and discovered that prices increased 23% between 2022 and 2023.



Some regions had more volatility than others. US-East-2 is traditionally less busy; hence spot prices are much lower than US-East-1, albeit more volatile over time. Spot pricing increased by 41% in US-East-2 in 2023 compared to 2022 and by 21% in US-East-1.





[1] Gartner Forecasts Worldwide Public Cloud End-User Spending to Reach $679 Billion in 2024, November 13, 2023.

[2] Wall Street Journal, Technology Chiefs Seek Help Wrangling Cloud Costs, March 3, 2023.

How to cut cloud costs and maintain efficiency with scale

1. Provision the right type, size, and number of VMs through automation

Choosing the right VMs for clusters is critical to avoid overspending on the cloud. Many DevOps teams start by defining their workload requirements and then search for the best match in their cloud provider's inventory.

This is easier said than done. If a team needs a machine with four CPU cores, AWS offers 600 different EC2 instances; the sheer scale is daunting.

Many teams simply choose instances they know and have used before. But when they deploy an application, they quickly realize that it is underutilizing other resources that they have paid for.

There are also other considerations – like cloud provider pricing plans, processors running in these instances, and choosing between burstable or non-burstable instances.

Provisioning a VM that has just enough resources to keep an application running smoothly is a challenge that can be addressed using machine learning algorithms to analyze and automatically optimize clusters. NielsenIQ, for example, lacked process control measures in certain areas, such as the orchestration of node pools. Automation enabled the company to create an architectural pattern that made it easy to stamp and roll clusters out, reducing the work on engineers' plates and saving the company up to 80% on its cloud costs.

James O'Hare

Principal Platform Engineer at NielsenIQ

NIQ

"Normally, we'd need to take time to find a good fit between cloud resources and workloads. With CAST AI, we can avoid a lot of that and adjust the capacity based on the analysis of workload requirements."

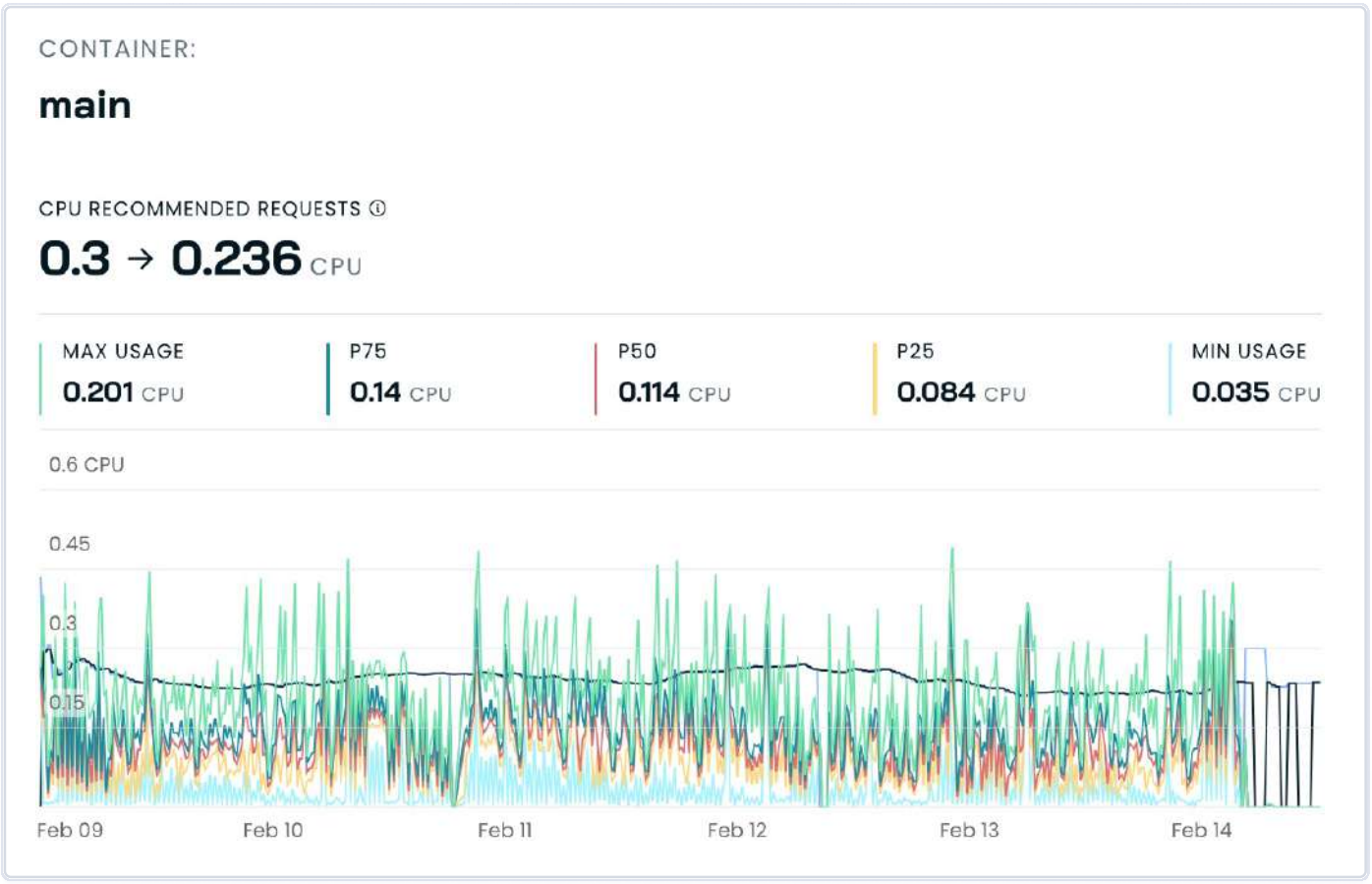
2. Set correct requests for workloads through rightsizing automation

In Kubernetes, workloads are sized using requests and limits, which are set for CPU and memory. Optimizing them is like walking a tightrope. Overprovisioning CPU and memory will keep the lights on, but it is costly. Underprovisioning them risks CPU throttling and out-of-memory kills, which cause applications to perform poorly or even crash.

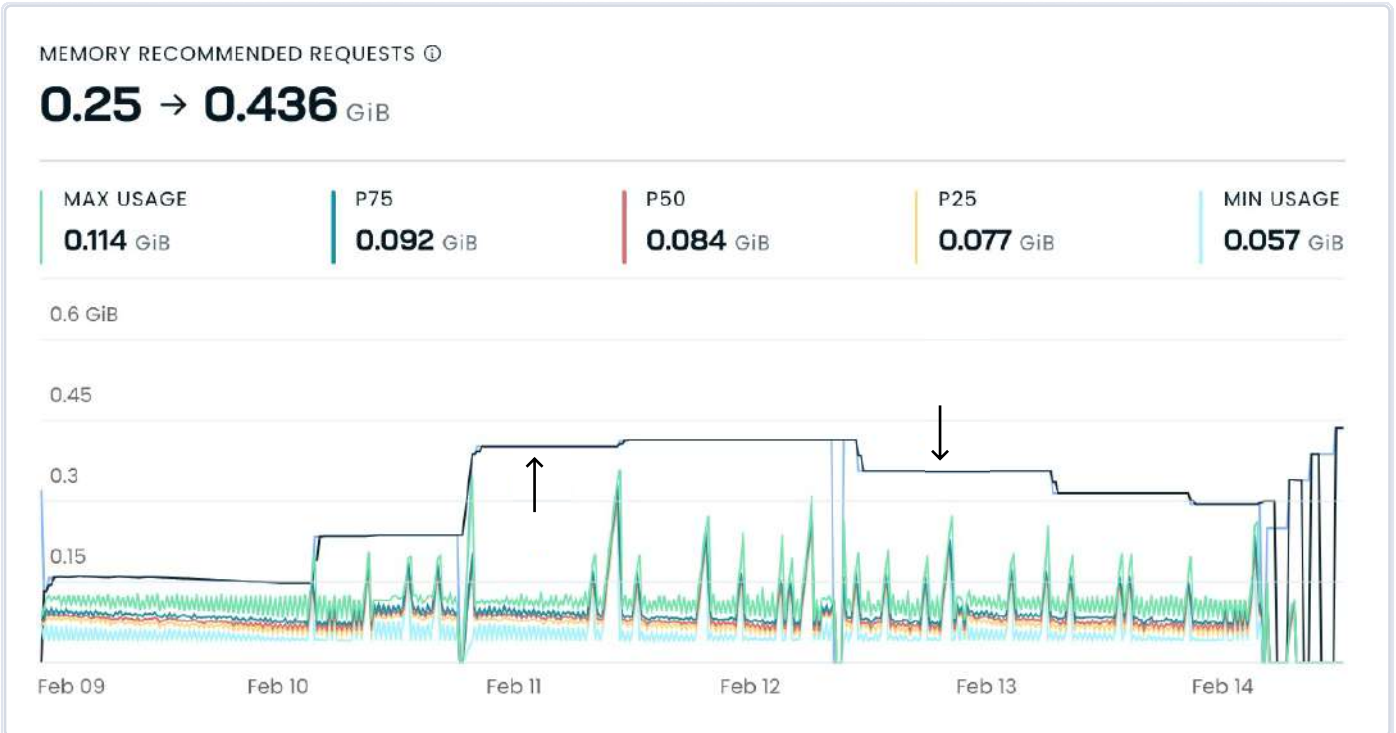
When teams do not fully understand what their container resource requirements are, they often play it safe and provision a lot more CPU and memory than needed.

Setting accurate requests calls for the ability to constantly monitor resource consumption and adjust requests as needed. This task involves every single pod running in the cluster, making manual optimization nearly impossible. This is where automated workload rightsizing comes in.

Below is an example of real-time CPU rightsizing. The requested CPU went from 0.3 CPU to 0.236, on average. The level of CPU is adjusted in real time according to the level of probability best suited for that specific workload.

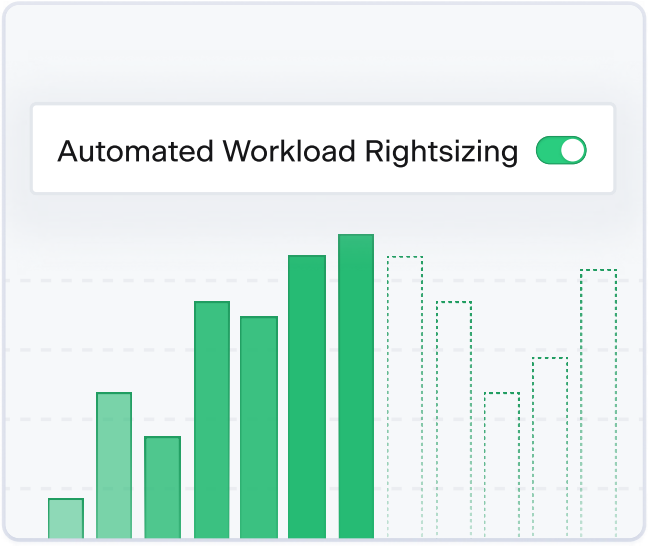


Memory was also adjusted in real time, from 0.25 GiB to 0.436 GiB. For memory, the engine was designed to avoid out-of-memory at all cost; hence the requested memory is set to always be above the real utilization.



3. Autoscale nodes to fight CPU waste

Kubernetes comes with three autoscaling mechanisms teams can use to increase resource utilization and reduce cloud waste.



The tighter these scaling mechanisms are configured, the lower the application's waste and cloud costs. However, their configuration and management are challenging, especially if teams use more than one autoscaler.

There are open-source solutions that provision and optimize nodes automatically, removing them when they are no longer needed. However, they also require time-consuming configuration and constant supervision to run smoothly.

Commercial solutions offer a higher return on investment because they are simple to set up and work automatically; often in line with the previously set policies. They scale cloud capacity up and down to match real-time demand changes without causing downtime.

ShareChat, for example, had trouble optimizing its node pools because Kubernetes lacks dynamic and flexible node pool provisioning.

Jenson C S.
Senior Engineering Manager at ShareChat

ShareChat

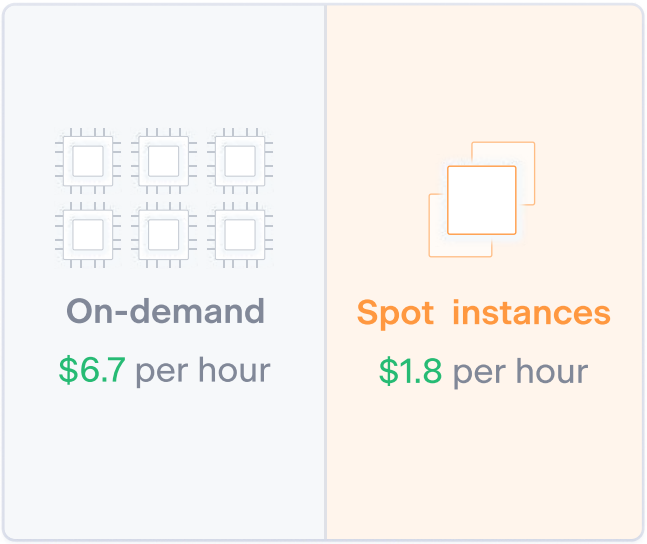
"We don't want to stick with one machine type, size, or family. We need different types to run and scale our platform. We wanted to be able to choose the percentage of nodes that would run on-demand and Spot instances for different types of workloads."

The company adopted CAST AI to automatically replace suboptimal nodes with new ones, and move workloads to help clusters quickly reach an optimal state – resulting in more than $1 million saved annually.

4. Use Spot instances

Despite the price increases highlighted earlier, Spot instances continue to offer an incredible cost-saving opportunity.

Since companies get these instances via a bidding process, they specify the price per hour. For comparison, the cost of an average CPU using on-demand is $6.7 per hour, whereas the cost of an average CPU using Spot instances is $1.8 per hour. Alternatively, teams can also have the price float – in this scenario, prices slowly go up over time, often leading to suboptimal results.



Spot instances come with a caveat: the cloud provider can reclaim them with as little as a 30-second notice. This is common during busy seasons, like Black Friday. Spinning up a new instance takes more time, so workloads could be left without a place to run, causing downtime.

The risks cause Spot instances to seem unstable and cause companies to shy away from using them. But in reality, they are a stable and cost-effective way to run applications if automation is used to provision, manage, and decommission infrastructure.

Like many companies, Spot instances were attractive to Branch because they provided the highest price discount. However, the company encountered significant challenges due to capacity shortages and frequent manual tuning. Branch reverted to using Savings Plans that offered more predictable capacity planning, but required a more expensive one-year to three-year financial commitment. This motivated the company to search for a solution that would automate Spot instance usage, offer real-time cost visibility, and provision the most cost-effective cloud resources to reduce the upfront commitment.

Using CAST AI, Branch is now able to fall back when Spot instances are reclaimed by automatically spinning up new equivalent compute instances of the most cost-efficient and available instance types. Now, workloads are automatically migrated to the old Spot instances that were previously reclaimed. This allowed Branch to deploy Spot instances to all stateless compute workloads in its Kubernetes clusters safely, with zero incidents incurred.

Mark Weiler
former Senior VP of Engineering at Branch

branch

“Partnering with CAST AI has been a big success for Branch, saving us several millions of dollars per year in AWS Cloud compute costs for our Kubernetes clusters, while maintaining our reliability SLAs.”

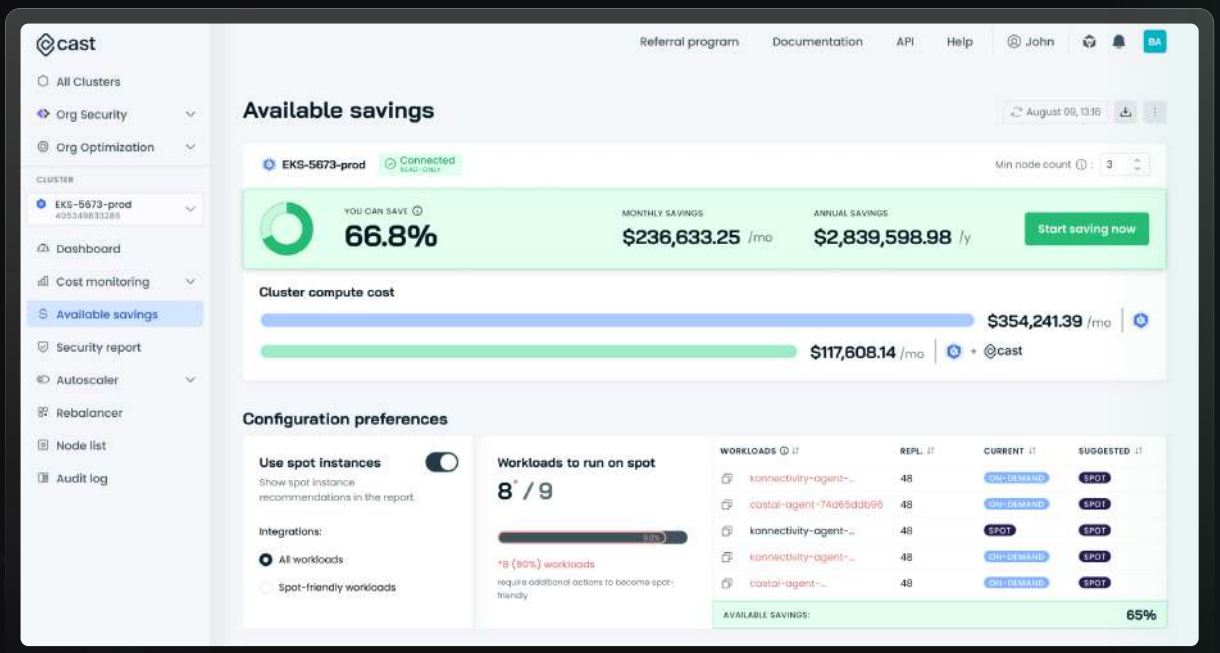
cast

Automate your Kubernetes today and start saving

CAST AI goes beyond monitoring clusters and making recommendations; the platform utilizes advanced machine learning algorithms to analyze and automatically optimize clusters in real time, saving customers 50% or more on their cloud costs, improving performance and reliability, and bolstering DevOps and engineering productivity.

Connect a cluster in minutes and instantly see how much you can save with the CAST AI Kubernetes automation platform.

Learn More



Trusted by fast-growing companies worldwide

ITERABLE project44 ShareChat Surfshark® yotpo.