



---

# The Cloud Cost Management Buyer's Guide

---

By: Roxanne Williams . Content Strategy Manager

---



# Table of Contents

04	An Intro to Cloud Cost Management	
05	Top Down vs. Bottom Up	
06	Key	
07	Apptio Cloudability	12 Harness Cloud Cost Management
08	AWS Cost Explorer	13 Kubecost
09	Azure Cost Management	14 Spot
10	CloudHealth by VMware	15 Turbonomic
11	Google Cloud Billing	16 Yotascale
17	Author Appendix	



# Opening Words

**Bill Shock**  
**/bil SHäk/**  
**NOUN**

*The act of opening your cloud bill and experiencing the five stages of grief: denial, anger, bargaining, depression, and acceptance.*

---

It happens to the best of us. We leave an instance running, forget all about it, and experience a hefty dose of bill shock at the end of the month. We'll admit it: it happened to us too.

The great part about cloud cost management tools is that they've become pretty advanced with the help of machine learning. They can now help us avoid the aforementioned issue by alerting us of anomalous spend, checking our instances to make sure resources are being appropriated correctly, and recommending various optimizations for our workloads.

In this buyer's guide, we've evaluated vendors based on their features. How much tag hygiene is required? Can they provide granular visibility into costs, down to the microservice? What about Kubernetes costs? Anomaly detection? We've compiled results in these pages in order to help you make an informed decision during the buying process.

Say goodbye to bill shock, and hello to well-thought-out cloud cost management.

**The Harness Team**

---

## **Editor's Note:**

This is v2 of the CCM Buyer's Guide. We revamped how we rated products and added notable feature updates to the vendors in this list. Our new rating scale just so happens to be the acronym to one of the best TV shows ever, so... Enjoy!



# An Intro to Cloud Cost Management

Cloud cost management is a hot topic, and is the new imperative for any software company these days. Despite the variety of cloud providers or platforms such as Amazon Web Services (AWS), Google Cloud Platform (GCP), or Microsoft Azure, it seems none arm engineering teams well enough to actually handle their cloud costs.

In fact, AWS itself admitted as early as 2017 that about 35% of cloud spend is wastage, and back then, that was \$6.4B of their revenue. In 2020, Gartner predicted that the public cloud market would grow to \$257.9B for the year. That means companies were projected to collectively waste \$90.8B in public cloud spend. The public cloud providers probably aren't complaining about that, but engineering and company budgets definitely are.

## The Importance of Cloud Cost Optimization

The biggest and most obvious reason to optimize your cloud costs is simple: to keep your cloud bill as small as possible. In the same way we don't want to blow up our personal wallets and keep costs down where we can, we want to keep our cloud spend to a manageable level so we have money to spend down the line. You can check out our post about picking the right cloud cost management method to see what might work for you.

Imagine if you had a \$100 cloud budget that you wanted to stretch for a month, but because of non-existent cloud cost management practices or visibility, you went through that in a day. You went bust in the first round. In the same vein, engineering teams have budgets—or discounted commits—with cloud providers that they expect to last the entire year. What happens when costs go out of control and the budget is hit (or vastly overrun) because they don't know what's going on with the cloud bill?

There are horror stories of startups who have straight up run out of money because they made one small mistake that ended up in a massive cost overrun. Getting visibility into cloud expenses and figuring out how to manage and optimize them is becoming an increasingly important issue—and as time goes on, more and more companies are expecting their developers themselves to take a hand in keeping cloud costs optimized. Today, managing cloud costs often falls to an external team that doesn't have the right context, and so while they can find cost savings, it's without an understanding of what's causing the costs.

To learn more, read the full article: ["A Developer's Guide to Reduce Cloud Costs"](#)



# Top Down vs. Bottom Up

There are two schools of thought when it comes to cloud cost management. The Top Down and Bottom Up approaches. Both have their merits—it really all depends on the kind of organization you want to be.

## Top Down

The Top Down approach places responsibility on Finance teams. It's basically what it sounds like: the top has access to cloud cost management tools and reports, they see costs—usually in an abstracted manner—and then trickle down to the bottom to get answers on why x cost y, and to allocate costs to the correct departments/teams.

## Bottom Up

The Bottom Up approach, rising in popularity, spreads accountability down to the engineers doing the work. They get views into cloud cost management tools, usually at a “real-time” cadence (which usually ends up being hourly granularity) and can see immediately if there's something wrong with usage. This approach ensures Finance teams don't get a nasty surprise at the end of the month.

As for which method will work best for your organization, that's really up to you—but we'll leave you with this: there's a reason the Bottom Up approach is gaining traction in the cloud cost world. It only makes sense to distribute responsibility to those who are incurring the cost and allowing them to be involved with the process. An informed engineer is one that will make the right decisions.



# Cloud Cost Management Tools

## Key

At Harness, we've found that cloud cost management challenges primarily fall into the three use cases below. We've built our buyer's guide with these in mind. The icons will act as a key for all reviewed vendors. Look to the right of the vendor's name to see which pain points the solution most addresses.

- G

**Cost Governance:**

There should be guardrails put in place around cloud costs, essentially creating processes that ensure costs stay under control.
- O

**Cost Optimization:**

Organizations regularly spend too much with their cloud providers and they want to bring down costs to pay only for what they need/use.
- T

**Cost Transparency:**

There should be visibility of cloud costs across every layer of the organization, from executives to developers. All stakeholders should be able to understand cloud costs in the context that is relevant to them.

## How We Grade Tools

A tool gets credit for one of the [G] [O] [T] use cases if it has built a suite of capabilities specifically to solve the use case. All use cases below have 4 features, and in order to earn the proverbial star, 3 of the 4 must be included.

For example, if a tool has forecasting capabilities, while that falls under the Cost Governance umbrella, it does not alone qualify a tool to get the [G] rating if two more governance capabilities are nowhere to be found.

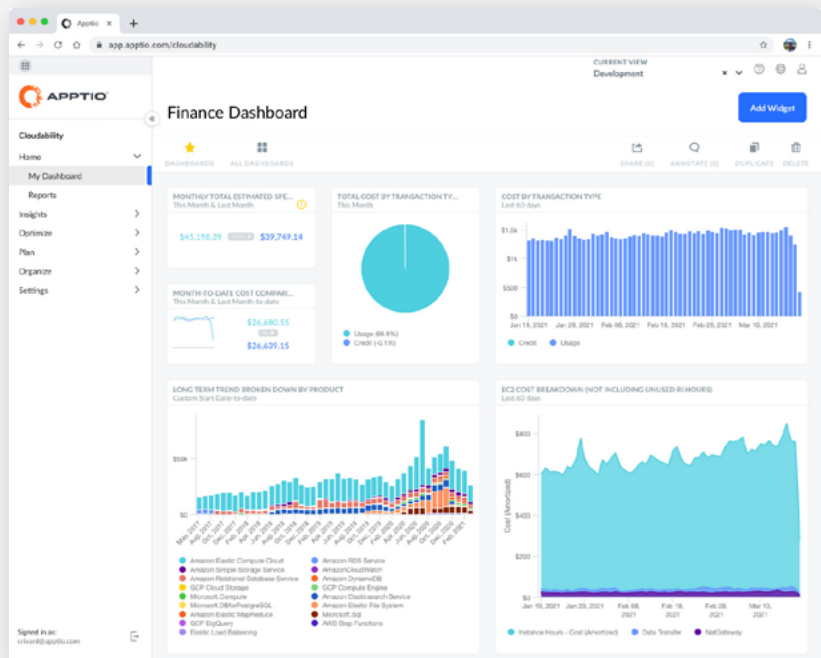
Cost Governance [G]	Cost Optimization [O]	Cost Transparency [T]
Budgets & Forecasting	Savings Recommendations	Kubernetes Utilization Visibility
Collaborative Cost Management	Anomaly Detection	Cost Perspectives
Cost Enforcement Policies	Active Cloud Management	Business Intelligence
Auditing	What-If Analysis	Business Mapping



# Apptio Cloudbility



Cloudbility is geared towards Finance and FinOps teams. Slightly disappointing, especially with the bottoms-up accountability many companies are shifting towards, which the FinOps movement espouses. They provide some good ways to get an understanding of costs, but only provide basic optimization capabilities.



## Pros

For a product geared towards Finance and FinOps, Cloudbility definitely offers those teams what they need to make informed decisions. It also goes the extra mile by providing a few neat features.

- Offers Anomaly Detection that uses user-defined thresholds to find spikes in costs and alerts the end user.
- Provides cost recommendations within cloud providers (not Kubernetes).
- Allows users to set budgets for specified contexts and provides cost forecasts relative to budgets.

## Cons

As mentioned above, complete cost transparency is a concern with Cloudbility. While the information they provide can be enough for Finance teams, it's not enough for engineers that want a real-time view into their costs in order to mitigate any issues. Users may find it lacking in a more robust set of cloud cost management capabilities.

- Not a good solution for hybrid cloud users as it doesn't provide hybrid visibility.
- Only provides views into utilized instances, no idle or unallocated visibility.
- Requires tagging to provide granular cost visibility and to do cost allocation.

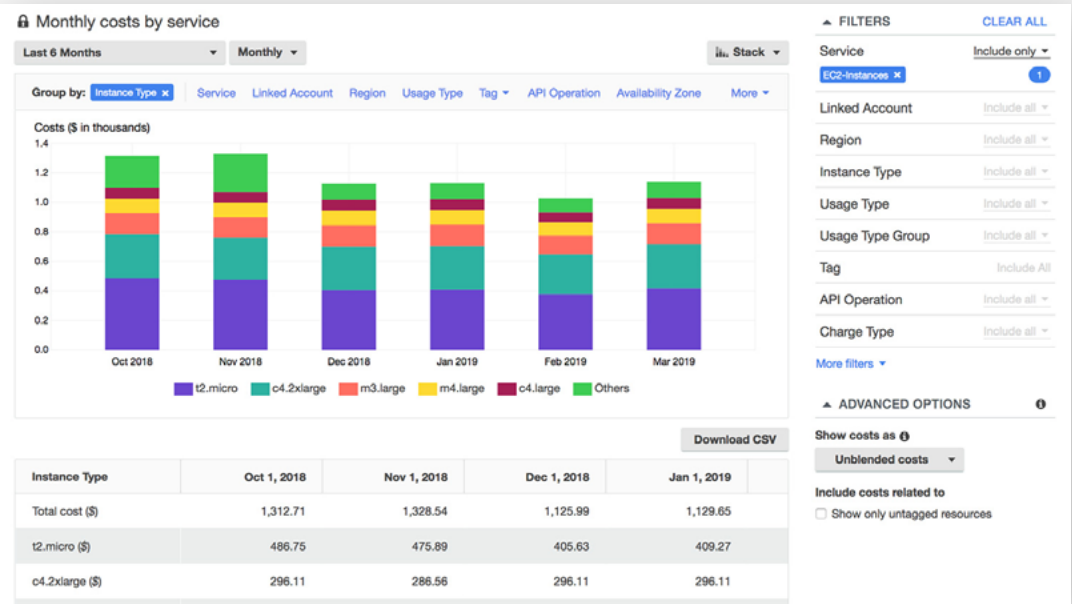


# AWS Cost Explorer



AWS Cost Explorer provides a high-level overview of your cloud costs in a dashboard format—as the name implies, only for AWS cloud users. You can expect to understand overall costs fairly well. Other visibility is achievable, but requires thorough tagging, and the tool is not designed for non-finance contexts. While users will find support for

very basic capabilities, AWS Cost Explorer innovates too slowly to make cloud cost management easier, leaving them behind the curve.



## Pros

AWS Cost Explorer provides a good set of visibility, savings, and forecast tools for AWS. As a free tool available to AWS customers, it provides the basics that Finance teams and budget owners would want to know.

- Free within the AWS Cost Explorer interface.
- Offers hourly granularity and good visibility with tagging.
- Supports a variety of build agents with different resource allocations.
- Allows users to set budgets and see cost forecasts.

## Cons

To fully leverage the capabilities of AWS Cost Explorer and get a deep understanding of costs, there's an inordinate amount of tagging required. Moreover, they don't score well on the GOT scale, demonstrating a real lack in capability as a cost management tool.

- Only for AWS—Kubernetes costs outside of ECS and EKS aren't included.
- Add-on services, like anomaly detection, aren't included natively and cost extra.
- No visibility into utilized, idle, or unallocated costs.



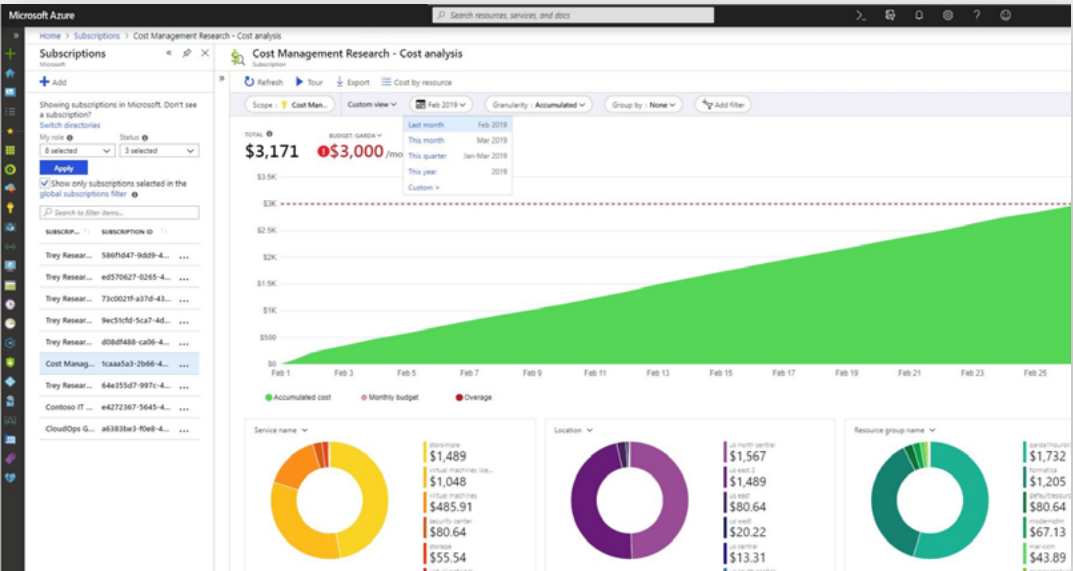


# Azure Cost Management



Azure Cost Management, perhaps surprisingly, is able to provide visibility into AWS spend via Cloudyn—that's right, it's not only for Azure, though we wouldn't use it if we were solely on the AWS cloud. It boasts business intelligence capabilities thanks to PowerBI. It also allows you to set budgets, get insights into forecasted spend,

and receive recommendations on potential savings opportunities. However, Azure Cost Management lacks desirable features such as anomaly detection, What-If Analysis, and Kubernetes utilization visibility—and you'll end up doing math in order to figure out realized savings vs potential savings.



## Pros

Offers decent visibility (through tagging), savings, and forecasted costs. Can pull data into PowerBI to do more complex financial work.

- Built for Azure, but can surface costs in AWS as well (though we wouldn't use it if we were solely on the AWS cloud).
- Offers recommendations and optimization opportunities at no cost to Azure users.
- Can create powerful BI views to slice and dice cloud cost information for any context.
- Allows users to set budgets and see cost forecasts.

## Cons

Azure Cost Management was built with Finance teams in mind. As such, Engineers may find that it doesn't uncover as much data as they'd like in order to get a holistic view into costs. Additionally, as far as Kubernetes goes, only AKS costs will be surfaced. Moreover, they don't score well on the GOT scale, demonstrating a real lack in capability as a cost management tool.

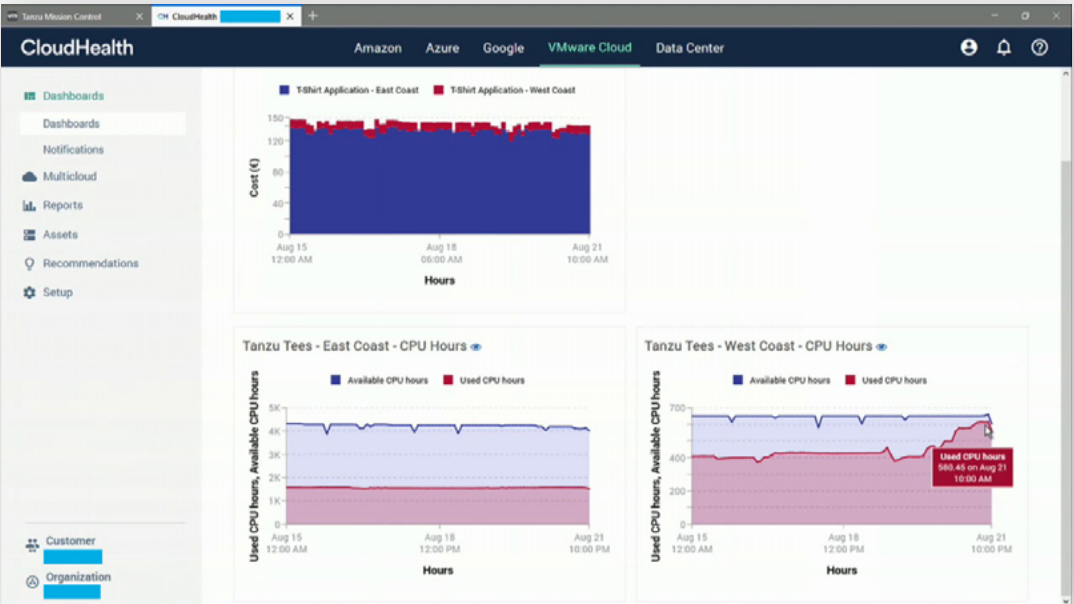
- Requires pristine tagging in order to get granular visibility into costs.
- Lacks basic features, requires time coming up with workarounds.
- No visibility into utilized, idle, or unallocated costs—and no anomaly detection.



# CloudHealth by VMware



Founded in 2012, CloudHealth was acquired by VMware in 2018 in an effort to continue building on their cloud-based services. CloudHealth was designed for Finance teams and provides good governance and reporting capabilities. Good for Finance teams, bad for Engineering.



## Pros

CloudHealth is a decent tool that supports a wide variety of use cases and is especially good for cloud users with no Kubernetes costs. They provide a breadth of tools to understand costs, and have built great governance capabilities.

- Good visibility when tag hygiene is followed closely.
- Supports all major cloud providers and hybrid cloud users.
- Really good governance capabilities built for top down cloud cost management.
- Allows users to set budgets and see cost forecasts.

## Cons

CloudHealth seems to struggle with Kubernetes costs more than other tools. It also requires a lot of human intervention, which can result in errors/missed opportunities.

- Anomaly detection is offered, but users must manually identify anomalous spikes.
- Recommendations do not include Kubernetes resources.
- Not conducive to getting granular or bottom up visibility, focusing heavily on reporting and governance—not built well for folks outside of that niche group to use.

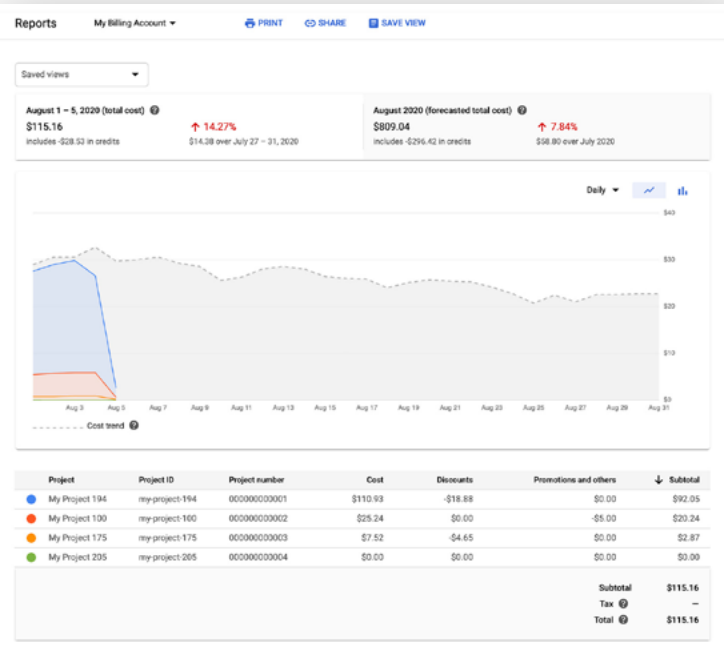


# Google Cloud Billing



Google Cloud Billing, as the name suggests, only works within the Google Cloud Platform. As with other cloud-specific cost tools, it was made with Finance teams in mind, not engineers. It does offer hourly updates, which is nice, and good visibility once you tag resources appropriately. However, Kubernetes costs are limited

to resources within GKE. Overall, the tool provides basic support across the board and like the other native billing tools, only marginally helps with actually managing cloud costs.



## Pros

Once tagging is complete, Google Cloud Billing offers a good set of visibility into cloud costs. Good tool for the top down approach at its most basic level.

- Gives Finance teams good point-in-time views into costs without the nitty gritty engineers want to see.
- Provides cost recommendations for GCP services.
- Allows users to set budgets and see cost forecasts.

## Cons

Google Cloud Billing feels like an incomplete tool—as if you'd need to pair it with a secondary product to get an accurate view into all costs. Moreover, they don't score well on the GOT scale, demonstrating a real lack in capability as a cost management tool.

- No visibility into utilized, idle, and unallocated costs.
- Not a good candidate for multi-cloud setups.
- No anomaly detection, leading to potential cost spikes with no recourse.

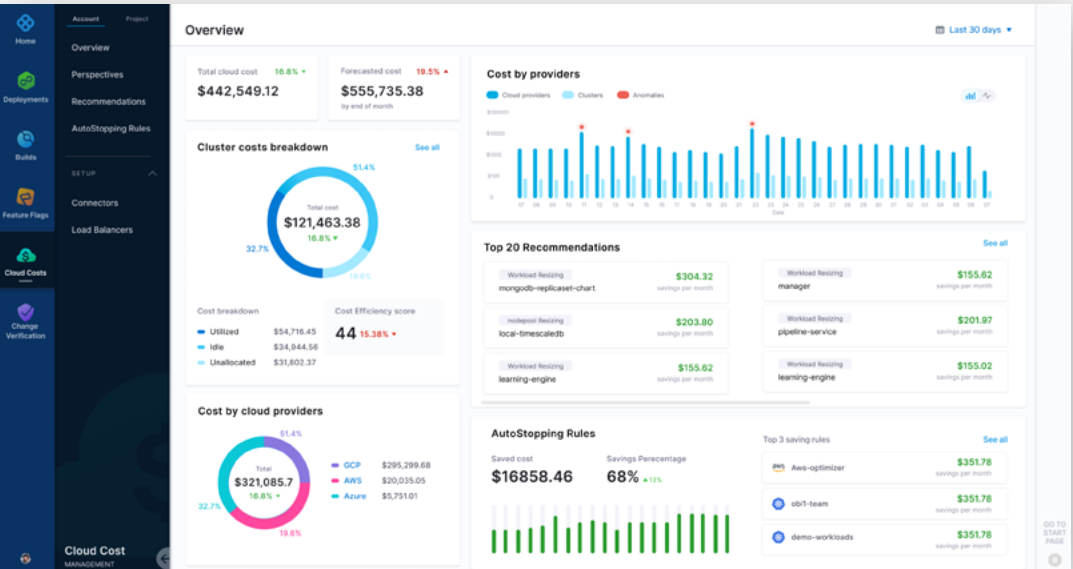


# Harness Cloud Cost Management



Harness Cloud Cost Management was built with Engineering teams in mind, providing deep views into costs without requiring tagging. Additionally, it offers features unique to the industry, such as What-If Analysis, Cloud Cost Business Intelligence, and Intelligent Cloud AutoStopping, which saves customers up to 75% on their

cloud bill. It also provides complete root cost analysis, and granular visibility into cloud spend—including all forms of Kubernetes.



## Pros

Harness CCM hits all three use cases where it hurts, and packs an even bigger punch by providing industry-first features.

- Built with strong focus on Kubernetes costs, providing unparalleled views into granular costs without requiring tagging.
- Provides Cloud Cost Business Intelligence and Intelligent Cloud AutoStopping for smart cloud cost management.
- Supports all major cloud providers, multi-cloud setups, and hybrid cloud.
- Able to link cloud event, like a deployment, to its associated cost.
- Allows users to set budgets and see cost forecasts.

## Cons

Harness CCM was designed with Engineering in mind, and while Finance teams can customize their reporting with BI to get custom, it may require more setup effort upfront.

- Savings recommendations don't include non-container cloud workloads.
- No cost allocation/business mapping.
- Customization options increase effort to set up the tool.

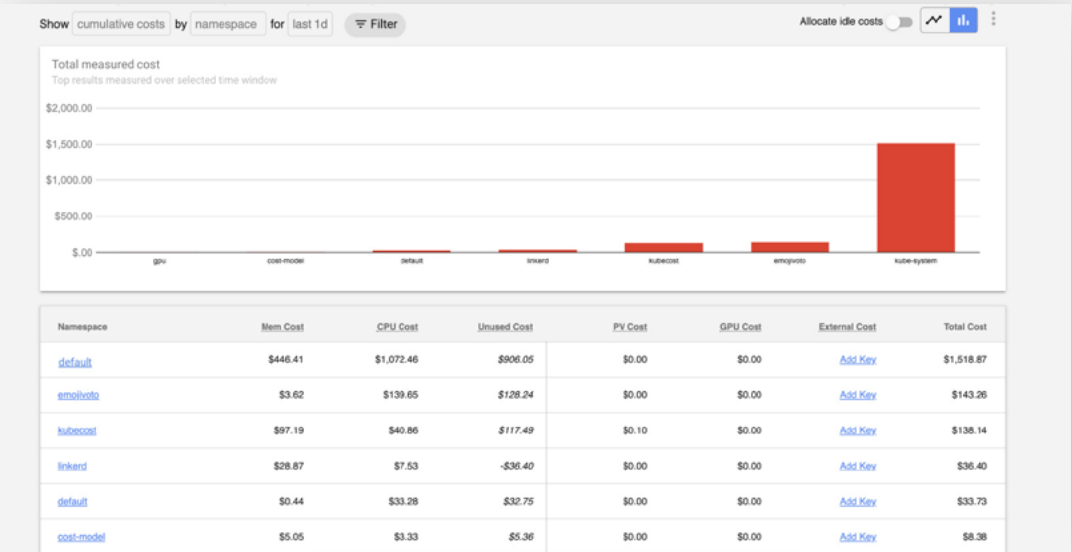


# Kubecost



Kubecost, as the name suggests, was built with Kubernetes costs in mind almost exclusively. That means that for any cloud costs that aren't in K8s, you'd be flying blind and would need a secondary solution—for instance, pairing Kubecost with Google Cloud Billing—for a complete view into costs.

However, what it does for Kubernetes is great—we'll give credit where it's due. Visibility is on point, and it does offer modern features like anomaly detection.



## Pros

Great visibility into Kubernetes costs, which is where the world is heading.

- Supports Kubernetes-based services on all major cloud providers like AKS and EKS.
- Tagging isn't required.
- Provides full root cost analysis views: utilized, idle, and unallocated.

## Cons

Kubecost is great at what it does: Kubernetes costs. Other costs, which probably make up a good portion of your cloud bill, won't be taken into account and will require a second tool for complete insight.

- All visibility comes with the huge caveat of "only for Kubernetes"—no visibility into cloud provider costs.
- Doesn't have access to CD pipelines so no context available for cloud event correlation.
- Users can set budgets and get alerted when they cross thresholds, but cannot see forecasts to drive proactive action.

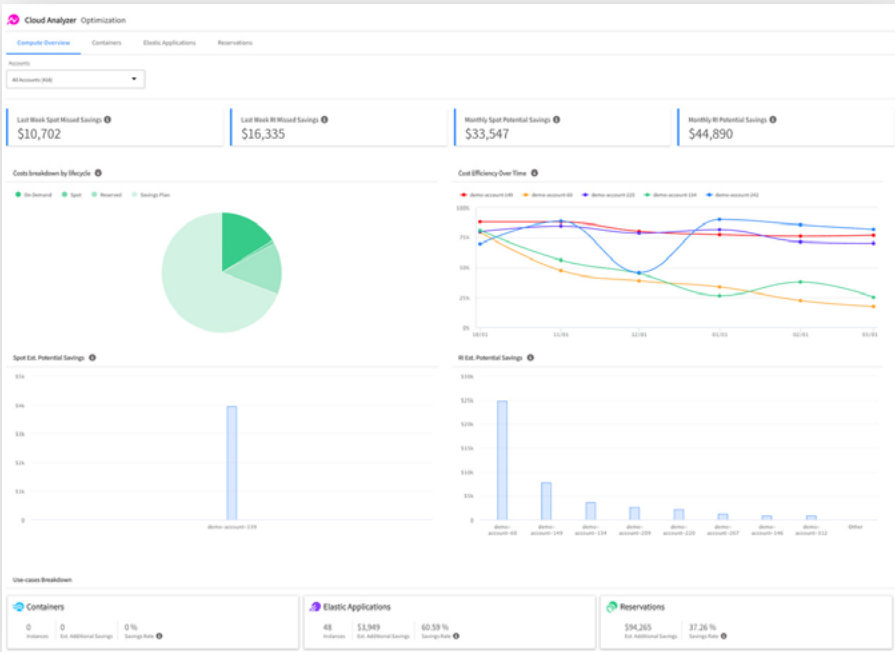


# Spot



Spot, built with Engineering teams in mind, focuses more on active cloud cost management rather than transparency and governance, even though we've found both are necessary to get a complete picture into cloud cost management. Spot is good at what it does, but

there's room for growth that we're sure they'll implement in the future.



## Pros

Decently robust tool that provides very actionable insights and savings opportunities.

- Offers anomaly detection, ensuring cost spikes are addressed immediately.
- Automates savings with spot orchestration.
- Actively manages your cloud and provides visibility into spend.
- Allows users to see cost forecasts relative to their spend, but does not map to budgets.

## Cons

Spot focuses on the savings/active portion of cloud cost management, missing the 'full picture' into cloud costs. Passive aspects, like visibility, are achievable but require manual work.

- Requires tagging, lacks filtering and grouping options for Kubernetes clusters.
- Unable to retrieve context into cloud events—no access to CD pipelines to unearth this information.
- No views into utilized, idle, and unallocated costs—and basic transparency use case support.

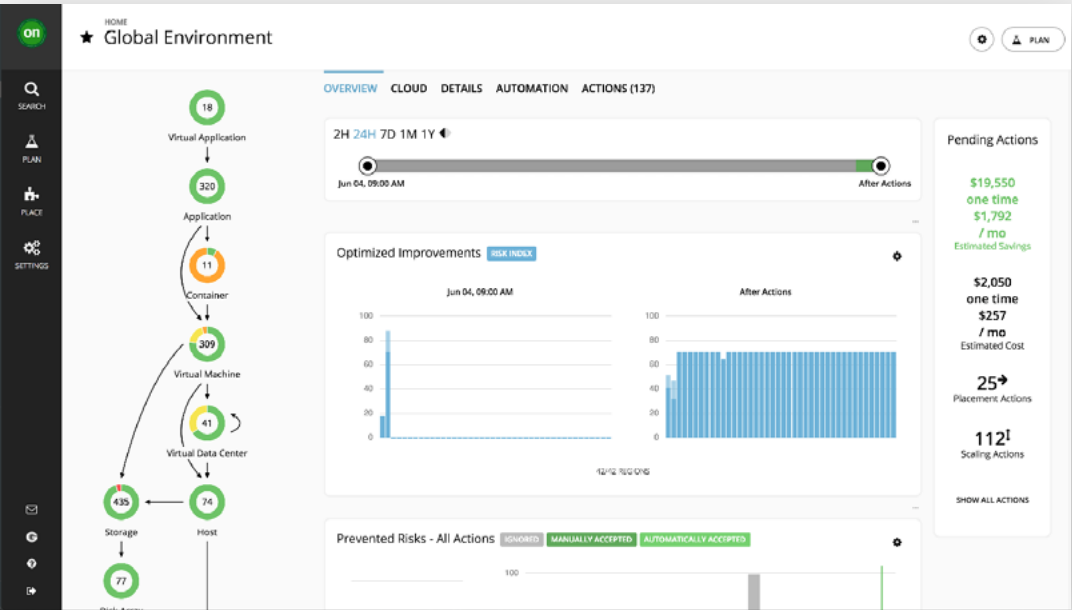


# Turbonomic



Turbonomic isn't really a cloud cost management tool. It offers some of those capabilities, but really, it's an enterprise-grade ARM (Application Resource Management) tool that is extremely complex. It doesn't offer many of the 'standard' features other cloud cost tools readily offer, and visibility is quite minimal.

We absolutely do not recommend this as a cloud cost management tool, unless you have a large data center on premises, or leverage a hybrid cloud environment.



## Pros

Turbonomic uses AI/ML to power their Actions feature, which allows for automatic resource reallocation.

- Alerts you when thresholds are reached.
- Supports all major clouds.

## Cons

Lacks many desirable cloud cost management features, as it is not a cloud cost tool in itself.

- Lacks even basic visibility and reporting.
- No anomaly detection, root cost analysis, or cloud event correlation.
- Extremely complex.

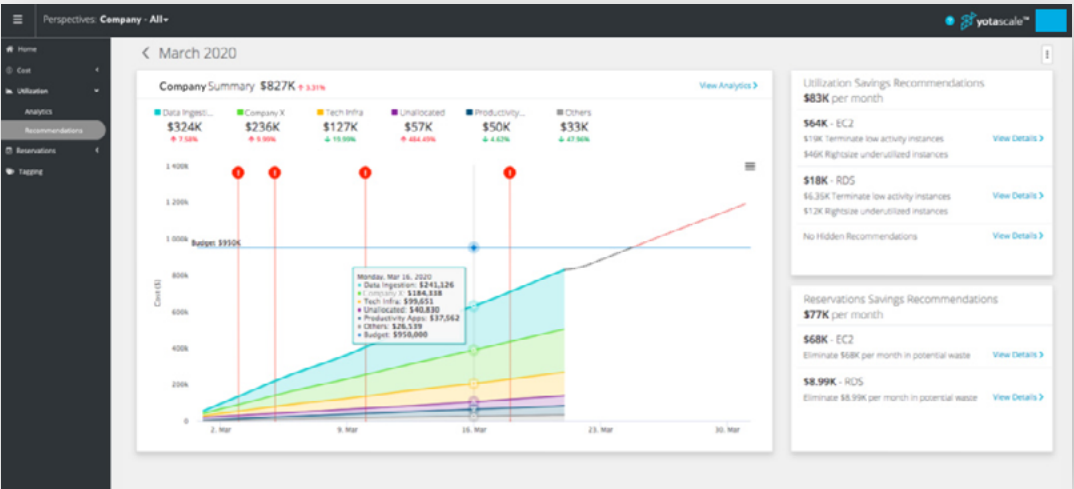


# Yotascale



Overall, Yotascale is a robust cloud cost management tool that offers nearly every sought-after feature. It doesn't require tagging and it offers good visibility—as long as you're in AWS. That's right, Yotascale only supports AWS as of this writing, so unfortunately it's not a good option for users on GCP or Azure—

or multi-cloud, for that matter. But, for AWS users, Yotascale provides a great Engineering experience.



## Pros

As far as AWS-only tools go, Yotascale provides much more in terms of simplicity (no tagging required) than AWS Cost Explorer does. All features are included as well, you don't need to pay extra for 'add-ons' like Cost Explorer does.

- Provides anomaly detection, alerting, and cost recommendations.
- Visibility into AWS is pristine, including allowing users to set budgets and see cost forecasts.
- Provides automated cost allocation without tagging requirements.

## Cons

As Yotascale only functions in the AWS cloud, it won't be a good option for organizations with a multi-cloud setup (or, obviously, GCP/Azure users).

- Does not provide visibility into utilized, idle, and unallocated costs at a glance—can be inferred, but requires mental/manual work, unlike a product where these costs are surfaced out of the box.
- Kubernetes costs, if not in AWS, won't be surfaced, nor will optimization opportunities.
- Unable to retrieve context into cloud events—no access to CD pipelines to unearth this information.





# Author Appendix

---

## Written By:

---

**Roxanne Williams**  
Content Strategy Manager  
Harness

---

Roxanne is a lover of all things DevOps, devoting a large portion of her time at Harness to competitor research. As such, she has developed a good understanding of where platforms shine—and where they can improve.

---

## Reviewed By:

---

**Bhargav Brahmbhatt**  
Technical Product Marketer  
for Cloud Cost Management  
Harness

---

With a background as a software engineer, Bhargav has invested in bringing the next generation of cool engineering tools (and toys) to the people.