SADA

Google Cloud Partner

WHITE PAPER

Unit costing: The next frontier in Cloud FinOps

Author: Rich Hoyer
Director of Customer FinOps, SADA

Co-author: Eric Lam
Head of Cloud FinOps, Offering Leader, Google

Contributor: Pathik Sharma
Cost Optimization Practice Lead, Google

Introduction

In this paper we'll examine the nature of and the need for cloud unit costing and examples from cloud-first organizations who've been pioneering this important discipline. Unit costing will become the standard by which FinOps practitioners obtain full business context for their cloud costs.

Our exploration of the topic will begin by describing unit costs conceptually. To do so, we'll examine a range of various methods enterprises use to forecast and budget their cloud costs, starting with the least rigorous methods and working our way up to the most rigorous—for which unit costing is critical.

Unit costing as applied to budgeting and forecasting

Let's begin with the weakest form of budgeting, which can be encountered even in larger enterprises: none! In this scenario, prior periods' expenses are simply tallied, but the figures aren't compared to a particular measure of expected results, nor are budgets prepared for future periods' cloud spend. This isn't useful for planning purposes, nor does it tend to drive efficiency among public cloud consumers.

One step higher up the sophistication ladder would be a budget based on what was spent last period adjusted for some type of (presumably positive) growth factor. For example, "The Western US market spent $X on computing instances last quarter, and we estimate we'll spend about the same next quarter." From a planning perspective, this approach can be somewhat useful because it can help management with basic estimates of profitability and cash flows for future periods. Budgeting, however, is only partly about predicting the performance of future periods. Ideally, budgeting and forecasting drive the vital purposes of encouraging responsible behavior on the part of those who incur cloud computing cost. For the purposes of driving behaviors, using a "last period + "X" approach for budgeting is a poor choice, because it actually inverts desirable incentives: The more a consumer of cloud resources spends in any given period, the more budget they're awarded in future periods!

Moving up the chain of budgeting sophistication, we see a model for budgeting we like to call the "fraction" approach, which works as follows: Cloud costs are estimated to be some fraction of another measure such as revenues. Initially, the denominator of the fraction needs to be calculated (for example revenues), and thereafter the cloud cost component is derived by calculating the relevant fraction. This approach is certainly better than the "last period + X" approach, but it carries a very similar flaw. Specifically, in our experience, these figures are usually based on prior costs incurred without particularly much rigor around why the costs were incurred at the level they were.

Finally, the best method of budgeting is to use unit costing, wherein a unit measure of consumption of cloud services can be measured per unit of production of a product or service. To make the concept more clear, we can consider the much simpler example of a manufacturing environment, where unit costs are often much easier to calculate. For example, the four tires fitted to each new Chevy Corvette cost $X in nominal dollar terms and Y% of the total cost to build the Corvette. As such, the unit cost of tires per Corvette is easy to calculate.

Let's talk : 818 . 927 . 3660 / googlesales@SADA.com / SADA.com

Calculating unit costs

In cloud computing, the ease with which unit costs can be calculated will vary tremendously. To examine a range of scenarios, we can begin with a simple scenario. Imagine a business that allows users to upload their favorite photos to a site and then order coffee mugs, T-shirts, etc, decorated with their photos. Calculating the cloud computing costs for each unit of these products could theoretically be fairly easy. Imagine, for example, the average size of image files needed to be stored (and therefore the associated storage costs) are fairly consistent from product to product. Assume further that the compute time to conduct enhancement of the photos, the compute time needed to conduct facial recognition of the photos, and other associated compute processing is also fairly consistent. From there, the sum of these various costs forms the cloud compute unit cost for each T-shirt, mug, etc. In this scenario, budgeting for cloud costs can be highly accurate insofar as forecasts for unit sales of mugs, T-shirts, etc can be calculated accurately. Using unit cost for budgeting is superior to the alternative methods for the following reasons:

1. **Accuracy and rigor of the calculations is much higher.** Estimates for costs are based on multiplication of units times price and are therefore fully grounded in reality.

2. **Unit costs incentivize good behavior.** It's almost universally the case that the moment estimates are generated for unit costs to produce a unit of product or service output, a fundamental question gets raised: *Why* is the estimated number of units required at the current level? That question is almost always followed by the natural follow on: Could the quantity of units be reduced?

In the context of budgeting and forecasting, unit costing is an extremely accurate tool for planning purposes and helps to drive the desirable incentive of reducing costs. So far, on these pages we have used budgeting and forecasting as a convenient context for describing the benefits of unit costing, but that in no way implies that the budgeting and forecasting are the only contexts where unit costing is valuable. Consider, for example, critical business functions such as product management and pricing. By accurately measuring and managing unit costs, managers can make better decisions about product mix and can more accurately price products and services. Performance measurement is another vital context. When unit costs are employed, at the close of each period a variance analysis can be conducted between forecast and actual costs that can distinguish between volume and pricing variances. When unexpected variances are encountered, management can act to improve efficiency in volume consumption and/or solicit vendors for better pricing to enhance pricing efficiency, for example.

With all of this being said, the nature of cloud computing often makes unit costing considerably more complex to calculate than our idealized "mug / T-shirt photo imprinting" case. Shared services such as databases and containerized workloads, in particular, can introduce considerable challenges to arrive at unit costs. Each enterprise and each individual workload will vary tremendously in the degree to which they lend themselves to unit costing. Furthermore, we recognize that many enterprises that are comparatively new to cloud will have more important near-term priorities in managing their clouds than unit costing. As such, examination of some case studies of successful implementations of unit costing from organizations who've been cloud-first and have the opportunities to drive forward thinking in this unit costing exercise may be the best approach to further explore the topic.

Let's talk : 818 . 927 . 3660 / googlesales@SADA.com / SADA.com

CASE STUDY #1

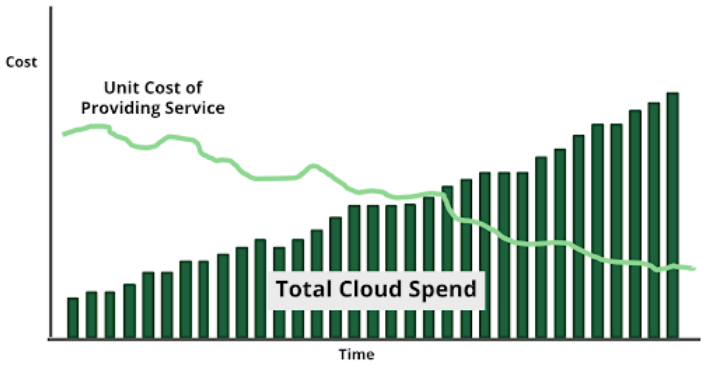
A SaaS provider where unit cost analysis informed the adoption of entirely different cloud services

Background

FinOps unit cost with Kubernetes

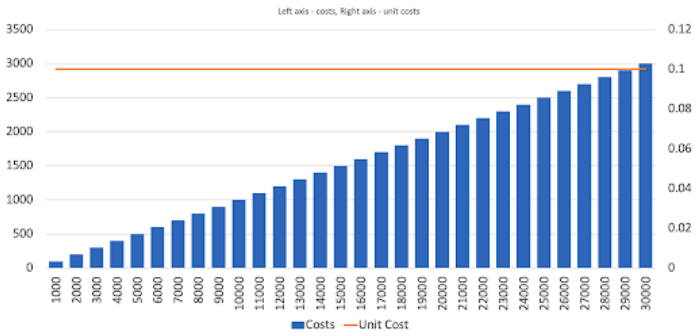
In 2020, a multi-billion dollar SaaS company considered migrating one of its largest and most visible products from traditional cloud compute infrastructure to Kubernetes (K8s). Cost reduction was not a motivating factor and not expected to be of resulting benefit. At the same time, the company wished to ensure the technology change wouldn't materially increase the cost of delivering its product, hindering its ability to profitably scale. Unit costs were used to measure hosting efficiency before, during, and after the transition.

Typically, unit costs are used to track the cost of delivering a product over time. The graph below shows the desired outcome of an effective FinOps practice - unit cost (light green line) is decreasing even though cloud spend (dark green bars) is increasing.



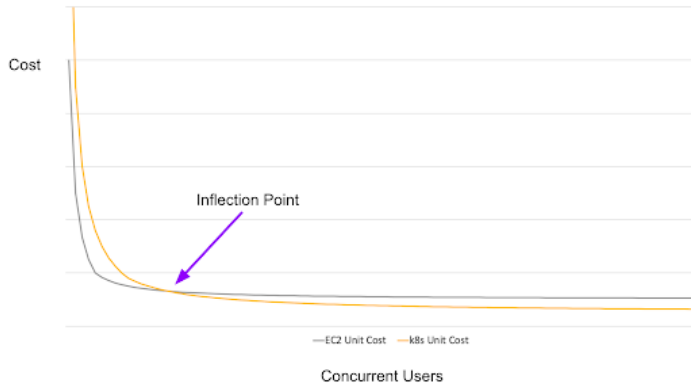
Putting some numbers to the graph: total costs are $10,000, and unit cost is $.333 per unit where a unit can be anything (for example: a customer, a transaction, etc.). At the right-most edge, costs are up 5x to $50,000—which in isolation is bad. Note how unit costs have dropped by ⅔—now only $.111 per unit. In the beginning, the business was only handling 30,000 units, but most recently it is doing over 450,000. So while costs had increased 5x, business has gone up 15x due to a tripling of unit efficiency. For every unit of business, $.222 had been freed for re-investment or profit.

Time is the most common—but not only—dimension you can use in tracking and understanding unit costs. For this SaaS company's technology change, concurrent user count provided a more relevant denominator. As use of the migrating product varied significantly throughout the day, week, and year, looking at things this way would provide understanding of the changes in efficiency of the different technology platforms at various load levels. The below chart paints a simple picture of what you might expect for cloud costs vs. concurrent users. In order to serve more users simultaneously, additional hardware is required. Costs go up as users go up, and we might expect this to scale fairly linearly. In this example, costs go up 100 as users go up by 1000, so there's a consistent unit cost (orange line) of .1.



But you won't know how this curve looks for your application without testing!

The company regularly conducts load-testing of its application at various concurrency levels to ensure stability of the user experience and supporting infrastructure. By performing these load tests against both the computing instances and Kubernetes technology stacks, recording concurrent load at different points in time, and overlaying those on time-corresponding cloud costs, a pair of unit cost graph curves were produced:



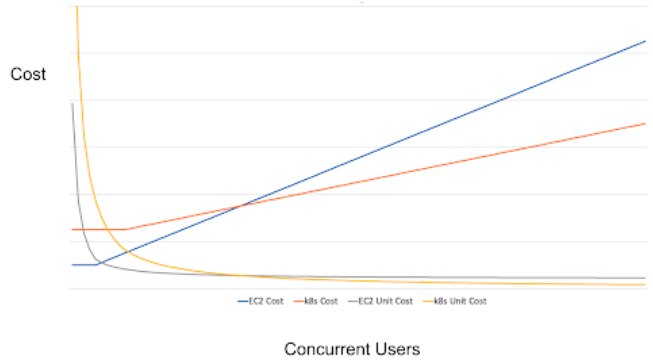
There are a few key takeaways from this analysis.

Firstly, the unit cost line is not linear. At very low load levels, cost per concurrent user is very high. At zero users, it is essentially infinite. High cost at low load levels is indicative of a "cost floor"— the base cost to provide the application's minimum footprint. Where multi-AZ and multi-region high availability strategies are employed, there can be significant costs incurred just maintaining the minimum footprint.

Secondly, the unit cost improves steeply early in the curve. Both technologies see their unit costs curves flatten out to the left of the inflection point. The point of flattening out corresponds to the application's minimum footprint becoming load saturated and the infrastructure starting to horizontally scale.

Next, the inflection point. This was the load level at which, for this application, Kubernetes became less expensive than computing instances on a unit cost basis. Below this point, the legacy compute infrastructure was less expensive, indicating it possessed a lower cost floor. However, as load increased, K8s quickly became the most efficient option.

While the unit cost difference appears small, it equates to a significant cost difference at high load levels:



Considering such results, the best technology to use, if based on cost, likely depends on how much time the application spends at the different load levels. In this case, most of the time was spent to the right of the inflection point. Still, the computing instance's superior efficiency at low loads was recognized as an opportunity for the K8s architecture to improve its low-load efficiency. In the time since, efforts have been made to improve its ability to scale down and reduce its cost floor.

Use of unit costs through this critical transition provided multiple lasting benefits. It provided validation that the technology change wouldn't negatively impact profitability of the business. It established a baseline of unit cost vs. load against which future technology iterations or upheavals can be compared. Lastly, when compared with time-spent-at-load data, it highlighted the circumstances, and thus the infrastructure components offering the most potential upside for investments in efficiency.

As highlighted from the case study, shifting to a cost-optimized Kubernetes platform can significantly lower the unit cost metric. Based on our experience with customers, Google Cloud has outlined the steps with leading practices to run cost-optimized Kubernetes.

1. **Understanding GKE options**
   Cost-optimized Kubernetes applications rely heavily on GKE autoscaling. To balance cost, reliability, and scaling performance on GKE, you must understand how autoscaling works and what options you have.

2. **Prepare cloud-based Kubernetes applications**
   Once you understand the basics of GKE autoscaling and other useful cost-optimized configurations, it's time to prepare your application to run on top of such an environment.

3. **Monitor your environment and enforce cost-optimized configurations and practices**
   Platform/infrastructure teams need to understand which group/application is resource-hungry and need to make sure everybody is following the company's policies. Google's GKE provides a usage metering feature that helps you understand the overall cost structure of your GKE clusters, what team or application is spending the most, which environment or component caused a sudden spike in usage or costs, and which team is being wasteful. By comparing resource requests with actual utilization, you can understand which workloads are either under- or over-provisioned.

4. **Spread the cost savings culture**
   Culture is the heart of every innovation, and to lower cost and get to the brass tacks of unit metrics, providing visibility and transparency to the cost metrics will be important. As shown in Google's DORA research, culture capabilities are some of the main factors that drive better organizational performance, less rework, less burnout, and so on. Cost saving is no different. Giving your employees access to their spending aligns them more closely with business objectives and constraints.

.

CASE STUDY #2

Unit cost analysis allows AdTech leader to completely exit 6 data centers within a year and dramatically lower unit costs

Background

Cost per billion ad requests

Unit cost analysis allowed a global leader in programmatic advertising technology that connects publishers and developers to advertisers to completely exit 6 data centers within a year and dramatically lower their unit costs by migrating to the Google cloud. “Cost per billion ad requests” was the key metric used both to demonstrate the merit of the migration to the cloud and to identify methods of optimizing the new cloud architecture as the migration progressed.

Their case is particularly interesting because the cost per billion ad requests could only be accurately projected and measured on the cloud (as opposed to the data center) infrastructure. This is because the elasticity of cloud makes unit costing much more meaningful than trying to unit cost by allocating large fixed capital expenses over projected sales volumes. Data center infrastructure needs to be architected with enough “headroom” to meet peak demands, which means the presence of excess capacity for some periods of time is unavoidable. In addition to the inefficiencies associated with that excess capacity, measured unit costs will appear to vary dramatically based on volumes. For example, unit costs may appear favorable (and stable) if throughput volumes are (i) generally stable and (ii) at or near the peak capacity of the infrastructure. When throughput volumes are highly variable, however, calculating unit costs will be less meaningful in a data center environment because unit costs will vary tremendously from period to period without any real change in costs incurred by the firm. For example, if throughput in one period drops to half of capacity, the unit costs will appear to have doubled without any real change in the economics of the organization. By contrast, the elasticity of the firm’s cloud infrastructure means cloud consumers like this advertising firm can accurately forecast and track unit costs irrespective of throughput volumes. Their key metric of cost per billion ad requests could be very accurately forecast and measured as their cloud environment scaled up and down with demand.

As the migration began, they made sure their results matched their projections by deploying the right combination of cost measurement tools to maximize visibility into their cloud spend. They developed a set of dashboards that provided cost visibility segmented by project, by service, by region, and by label.

As computing resources made up the substantial portion of the cloud cost, the team decided to focus on the unit cost metrics based on the Google Compute Engine (GCE) and Google Kubernetes Engine (GKE). They prioritized cost visibility around the compute usage types such as preemptible, on-demand, and committed use discounts in order to identify the lowest cost combination of these resources. They then enabled GKE usage metering to measure

Let’s talk : 818 . 927 . 3660 / googlesales@SADA.com / SADA.com

/ UNIT COSTING

Kubernetes costs by application, by cluster, and by region. Integrating these and other external data sets allowed an entirely new set of reports to be generated, including their critical cost per billion ad request metric.

Cost Per Billion Ad Requests

Reporting setup

GCP Console

Tables (mostly BQ)	Custom Data Sources	Reports
Billing Data Export		Unit/Cost Reporting
Public SKU Data		Billing Exceptions
Custom Perf Data		GKE Usage
GKE Usage Meter		Spike Detection
Google Sheets		Ad hoc Analysis
		P&L Reporting

Through a combination of Usage Optimization, Pricing Efficiency, Waste Reduction, and Refactoring, the company reduced per unit costs by more than 60% despite increased ad unit growth. On the graph below, the Y-axis represents the unit cost per billion advertisement requests served over a period of a timeline indicated on X-axis.

//// Conclusion

These two case studies provide good examples of how unit costing can help organizations better understand the impact of cloud costs to the business and can create sustainable business value from their clouds. Unit costs can help companies optimize their mix of products and services, provide more granularity into the per-unit basis for financial forecasting, and enable showback and chargeback modeling. Best of all, unit costing is the first step in allowing business to move toward activity based management, which is a critical step in creating sustainable, positive business outcomes and outperforming competitors.

Let's talk : 818 . 927 . 3660 / googlesales@SADA.com / SADA.com

SADA

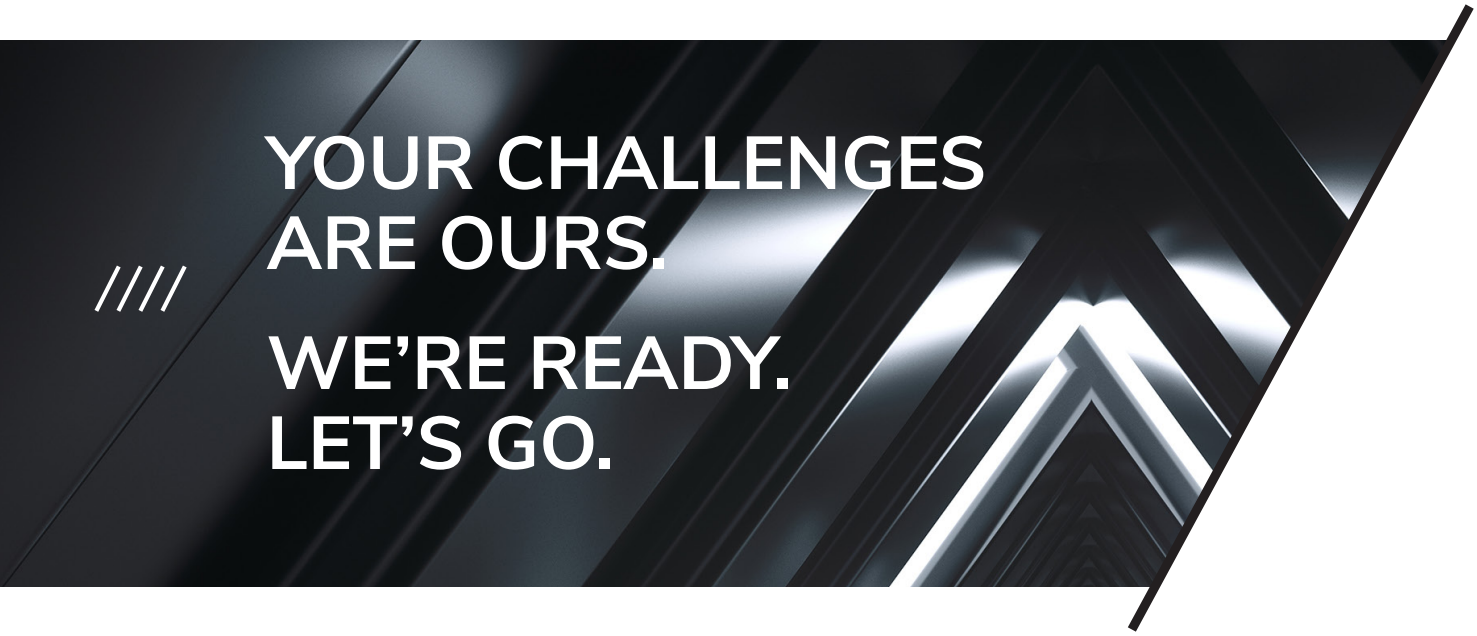
FinOps Certified Service Provider

About SADA

At SADA, we climb every mountain, clear every hurdle, and turn the improbable into possible – over and over again. Simply put, we propel your organization forward.

It's not enough to migrate to the cloud, it's about what you do once you're there. Accelerating application development. Advancing productivity and collaboration. Using your data as a competitive edge. When it comes to Google Cloud, we're not an add-on, we're a must-have, driving the business performance of our clients with its power.

Beyond our expertise and experience, what sets us apart is our people. It's the spirit that carried us from scrappy origins as one of the Google Cloud launch partners to an award-winning global partner year after year. With a client list that spans healthcare, financial services, media and entertainment, retail, manufacturing, public sector and digital natives – we simply get the job done, every step of the way.



"We've benefited from how SADA structured the way we consume Google Cloud services to make them more cost effective. It really has made a difference."

Ray Li
Co-Founder and CTO | Apollo.io

A few of our clients

evite · MADHIVE · MakerBot. · PACKETFABRIC · QuantumMetric

Let's talk : 818 . 927 . 3660 / googlesales@SADA.com / SADA.com