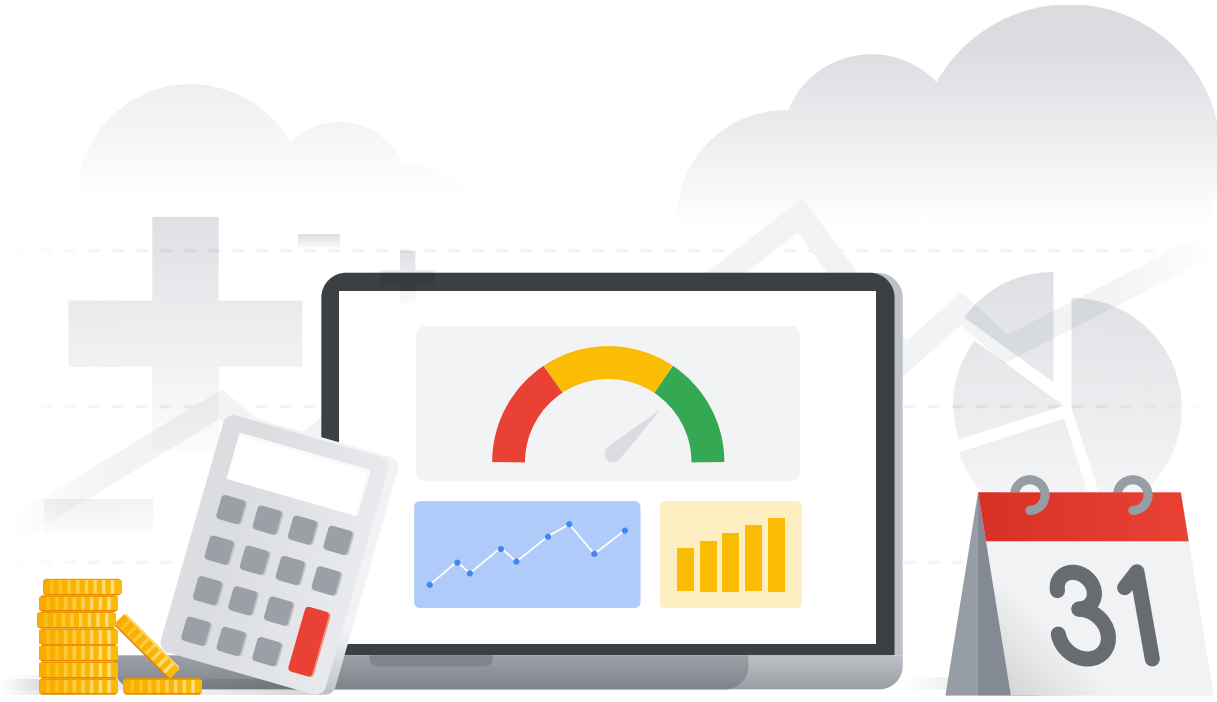
Google Cloud Whitepaper

Understanding the principles of cost optimization



Google Cloud

Google Cloud

Table of Contents

Google Cloud

Table of Contents (continued)

Contributors

Thanks to contributions from Google Cloud Professional Services team members:

- Justin Lerma
- Pathik Sharma
- Amber Yadron
- Andrew Sallaway
- Akshay Kumbhar

Introduction

There's always some degree of uncertainty when you're making plans and setting goals for your business. You never really know how customers will respond to a product, or whether sales forecasts will be accurate. But you can control how efficiently you run your technology systems, and track how well they're serving your business users. Today, with even greater global uncertainty added to the usual business uncertainty, it's even more important that you use your existing tech resources wisely, and do as much as you can with what you have.

At the end of the day, getting more out of your cloud resources can translate into more customers served, more issues resolved, and more adaptability for the overall business. Using your cloud resources more efficiently can help your team and business adjust to these new realities, and be as effective as possible.

Our own teams have worked for years with IT and operations teams across industries and around the world, listening to your challenges, your successes, and your future plans. While a lot has changed, the ability of technologists to adapt and thrive hasn't. Your ability to shift and change as the business changes matters more than ever, and our cloud technology is designed to support that kind of agility and resilience. We've gathered together the tips and best practices we recommend so that you can get more out of your existing resources—more VMs, more storage, more queries—and help your business meet its goals.

Read on to learn how to use built-in cost optimization features in Google Cloud and find lots of other tips to make sure you're getting the most out of your resources. And, importantly, you'll find tips on setting up processes and working with cross-functional teams to implement standards that make cloud efficiency and cost optimization part of your business culture.

Your ability to shift and change as the business changes matters more than ever.

Chapter 1
Principles and processes to optimize your cloud costs

Cloud is more than just a cost center. Moving to the cloud allows you to enable innovation at a global scale, expedite feature velocity for faster time to market, and drive competitive advantage by quickly responding to customer needs. So it's no surprise that many businesses are looking to transform their organization's digital strategy as soon as possible. But while it makes sense to adopt cloud quickly, it's also important to take time and review key concepts prior to migrating or deploying your applications into cloud. Likewise, if you already have existing applications in the cloud, you'll want to audit your environment to make sure you are following best practices. The goal is to maximize business value while optimizing cost, keeping in mind the most effective and efficient use of cloud resources.

We've been working side by side with some complex customers as they usher in the next generation of applications and services on Google Cloud. When it comes to optimizing costs, there are lots of tools and techniques that organizations can use. But tools can only take you so far. In our experiences, there are several high-level principles that organizations, no matter the size, can follow to make sure they're getting the most out of the cloud.

When it comes to optimizing costs, there are lots of tools and techniques.

Cost optimization with people and processes

As with most things in technology, the greatest standards are only as good as how well they are followed. The limiting factor, more often than not, isn't the capability of the technology, but the people and processes involved. The intersection of executive teams, project leads, finance, and site reliability engineers (SREs) all come into play when it comes to cost optimization. As a first step, these key stakeholders should meet to design a set of standards for the company that outline desired service-level profitability, reliability, and

performance. We highly recommend establishing a tiger team to kickstart this initiative. Later on in this book, we'll discuss specific service-level recommendations, but none of these will be scalable and effective until your framework is enforced programmatically.

Using cloud's enhanced cost visibility

A key benefit of a cloud environment is the enhanced visibility into your utilization data. Each cloud service is tracked and can be measured independently. This can be a double-edged sword: now you have tens of thousands of SKUs and if you don't know who is buying what services and why, then it becomes difficult to understand the total cost of ownership (TCO) for the application(s) or service(s) deployed in the cloud.

This is a common problem when customers make the initial shift from an on-premises capital expenditures (CapEx) model to cloud-based operational expenditures (OpEx). In the old days, a central finance team set a static budget and then procured the needed resources. Forecasting was based on a metric such as historic growth to determine the needs for the next month, quarter, year, or even multiple years. No purchase was made until everyone had the opportunity to meet and weigh in across the company on whether or not it was needed.

Now, in an OpEx environment, an engineering team can spin up resources as desired to optimally run their services. We see that for many cloud customers, it's often something of a Wild West—where engineering spins up resources without standardized guardrails such as setting up budgets and alerts, appropriate resource labeling and frequent cadence to view cost from an engineering and finance perspective. While that empowers velocity, it's not really a good starting position to effectively design a cost-to-value equation for a service—essentially, the value generated by the service—much less optimize spending. We see customers struggling to identify the cost of development vs. production projects in their environments due to lack of standardized labelling practices. In other cases, we see engineers over-provisioning instances to avoid performance issues, only to see considerable overhead during non-peak times. This leads

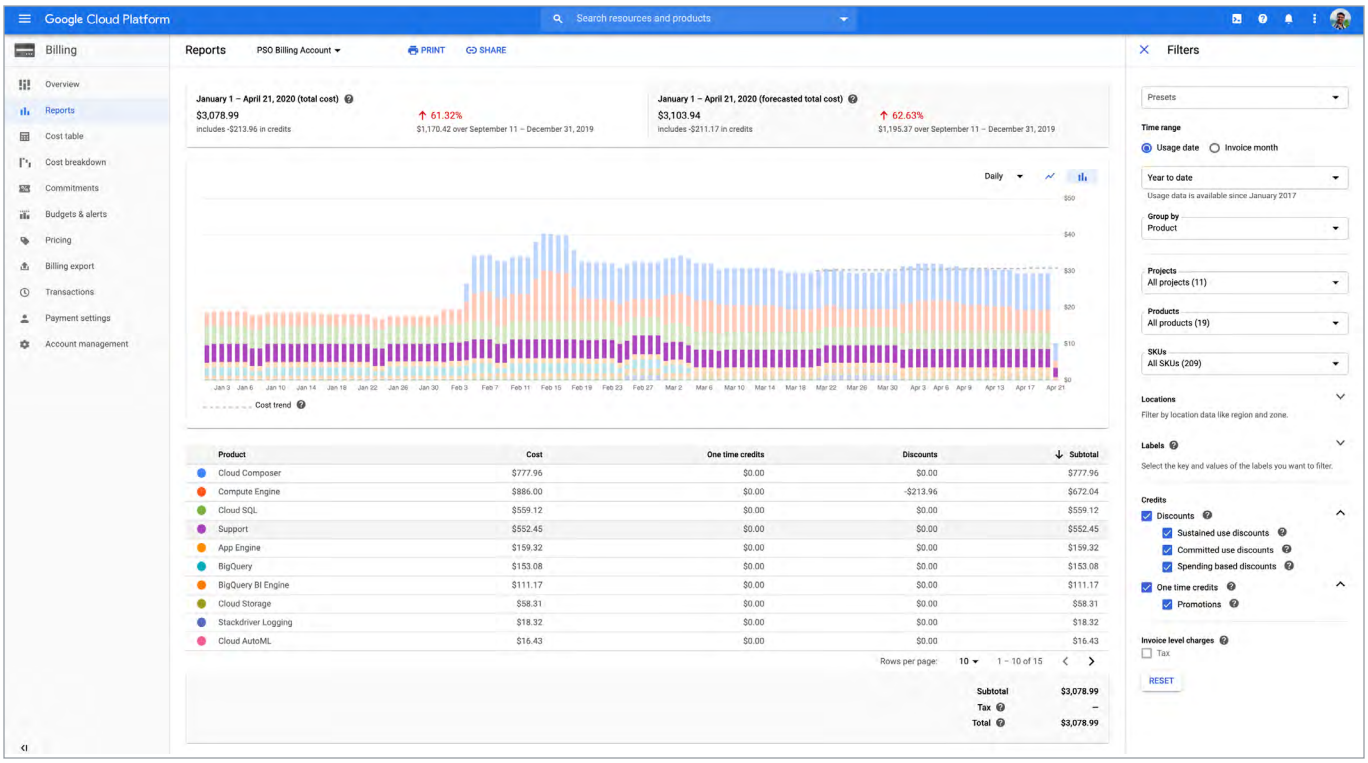
For many cloud customers, it's often something of a Wild West.

to wasted resources in the long run. Creating company-wide standards for what type of resources are available and when to deploy them is paramount to optimizing your cloud costs.

We've seen this dynamic many times, and it's unfortunate that one of the most desirable features of the cloud—elasticity—is sometimes perceived as an issue. When there is an unexpected spike in a bill, some customers might see the increase in cost as worrisome. Unless you attribute the cost to business metrics such as transactions processed or number of users served, you really are missing context to interpret your cloud bill. For many customers, it's easier to see that costs are rising and attribute that increase to a specific business owner or group, but they don't have enough context to give a specific recommendation to the project owner. The team could be spending more money because they are serving more customers—a good thing. Conversely, costs may be rising because someone forgot to shut down an unneeded high-CPU VM running over the weekend—and it's pushing unnecessary traffic to Australia.

One way to fix this problem is to organize and structure your costs in relation to your business needs. Then, you can drill down into the services using Cloud Billing reports to get an at-a-glance view of your costs. You can also get more granular cost views of your environment by attributing costs back to departments or teams using labels, and by building your own custom dashboards. This approach allows you to label a resource based on a predefined business metric, then track its spend over time. Longer term, the goal isn't to understand that you spent "$X on Compute Engine last month," but that "it costs $X to serve customers who bring in $Y revenue." This is the type of analysis you should strive to create.

One of the most desirable features of the cloud — elasticity — is sometimes perceived as an issue.



Billing Reports in the Google Cloud console let you explore granular cost details.

One of the main features of the cloud is that it allows you to expedite feature velocity for faster time to market, and this elasticity is what lets you deploy workloads in a matter of minutes as opposed to waiting months in the traditional on-premises environment. You may not know how fast your business will actually grow, so establishing a cost visibility model up front is essential. And once you go beyond simple cost-per-service metrics, you can start to measure new business metrics like profitability as a performance metric per project.

Understanding value vs. cost

The goal of building a complex cloud system isn't merely to cut costs. Take your fitness goals as an analogy. When attempting to become more fit, many people fixate on losing weight. But losing weight isn't always a great key indicator in and of itself. You can lose weight as an outcome of being sick or dehydrated. When we aim for an indicator like weight loss, what we actually care about is our overall fitness or how we look and feel when being active, like the ability to play with your kids, live a long life, dance—that sort of thing. Similarly, in the world of cost optimization, it's not about just cutting costs. It's about identifying waste and ensuring you are maximizing the value of every dollar spent.

Similarly, our most sophisticated customers aren't fixated on a specific cost-cutting number, they're asking a variety of questions to get at their overall operational fitness:

- What are we actually providing for our customers (unit)?
- How much does it cost me to provide that thing and only that thing?
- How can I optimize all correlated spend per unit created?

In short, they have gone ahead and created their own unit economics model. They ask these questions up front, and then work to build a system that enables them to answer these key questions as well as audit their behavior. This is not something we typically see in a crawl state customer, but many of those that are in the walk state are employing some of these concepts as they design their system for the future.

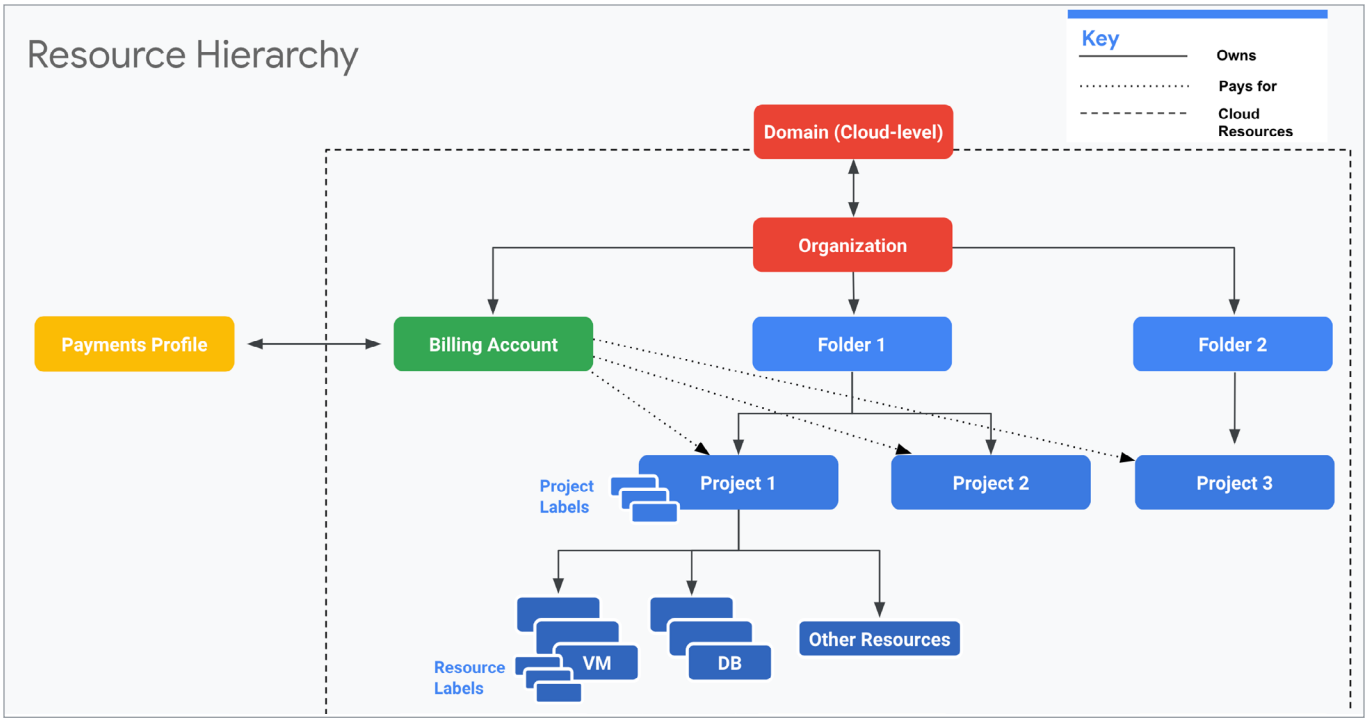
Implementing standardized processes from the get-go

Ensuring that you are implementing these recommendations consistently is something that must be designed and enforced systematically. Automation tools like Terraform and Cloud Deployment Manager can help create guardrails before you deploy a cloud resource. It is much more difficult to implement a standard retroactively. We have seen everything from IT Ops shutting off or threatening to shut off untagged resources to established "walls of shame" for people who didn't adhere to standards. (We're fans of positive reinforcement, such as a pizza, or a trophy, or even a pizza trophy.)

What's an example of an optimization process that you might want to standardize early on? Deploying resources, for one. Should every engineer really be able to deploy any amount of any resource? Probably not. We see this as an area where creating a standard up front can make a big difference.

Structuring your resources for effective cost management is important too. It's best to adopt the simplest structure that satisfies your initial requirements, then adjust your resource hierarchy as your requirements evolve. You can use the setup wizard to guide you

through recommendations and steps to create your optimal environment. Within this resource hierarchy, you can use projects, folders, and labels to help create logical groupings of resources that support your management and cost attribution requirements.

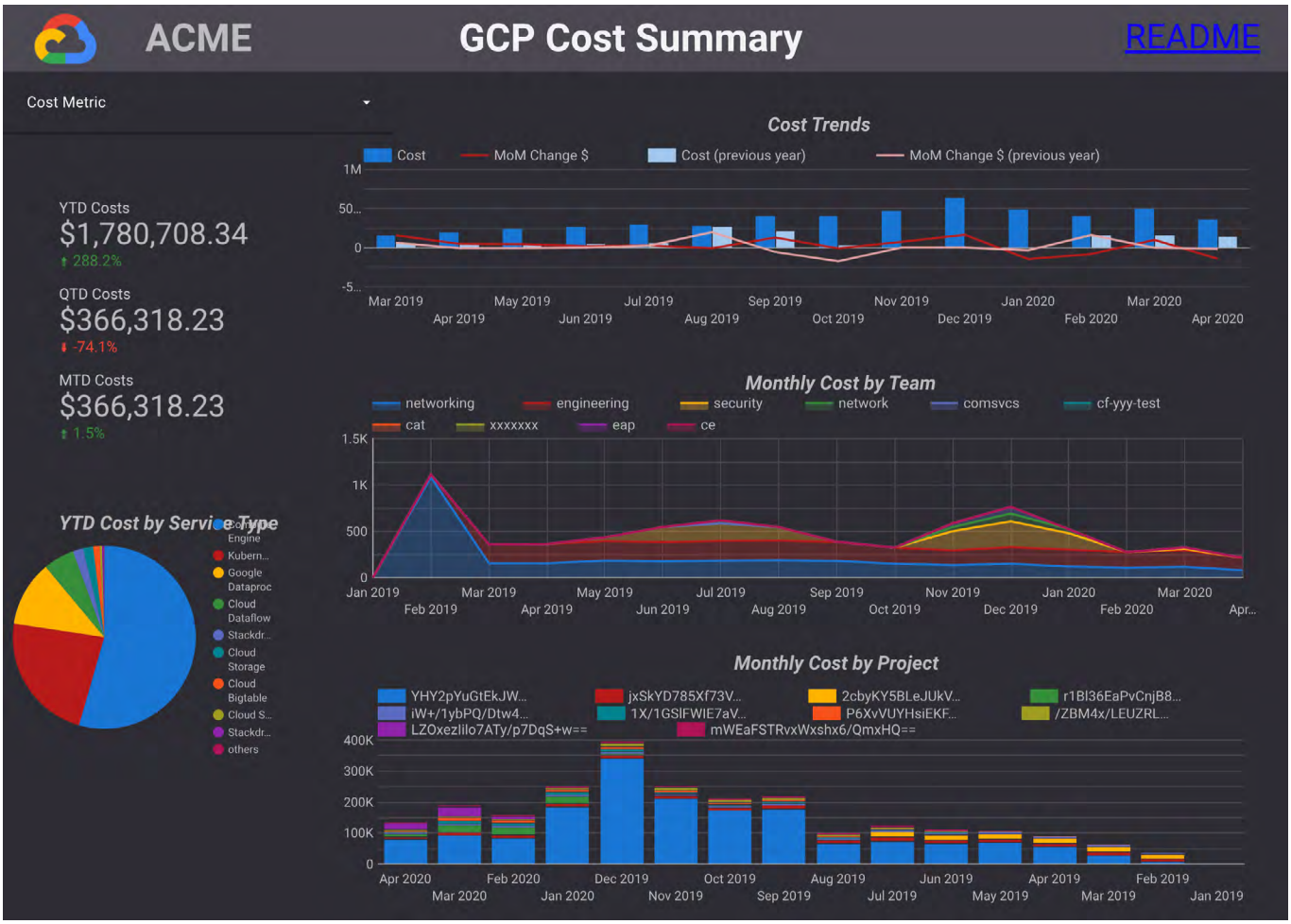


Example of a resource hierarchy for cloud

In your resource hierarchy, labeling resources is a top priority for organizations interested in managing costs. This is essentially your ability to attribute costs back to a specific business, service, unit, leader, etc. Without labeling resources, it's incredibly difficult to decipher how much it costs you to do any specific thing. Rather than saying you spent $36,000 on Compute Engine, it's preferable to be able to say you spent $36,000 to deliver memes to 400,000 users last month. The second statement is much more insightful than the first. We highly recommend creating standardized labels together with the engineering and finance teams, and using labels for as many resources as you can.

Review and repeat for best results

As a general practice, you should meet regularly with the appropriate teams to review usage trends and also adjust forecasting as necessary. The Cloud Billing console makes it easy to review and audit your cloud spend on a regular basis, while custom dashboards provide more granular cost views. Without regular reviews and appropriate unit economics, as well as visibility into your spend, it's hard to move beyond being reactive when you observe a spike in your bill.



Visualize spend over time with Google Data Studio

If you're a stable customer, you can review your spending less frequently, as the opportunities to tweak your strategies will be reliant on items like new Google Cloud features vs. a business change on your product roadmap. But if you're deploying many new applications and spending millions of dollars per month, a small investment in conducting more frequent cost reviews can lead to big savings in a short amount of time. In some cases, our more advanced customers meet and adjust forecasts as often as every day. When you're spending millions of dollars a month, even a small percentage shift in your overall bill can take money away from things like experimenting with new technologies or hiring additional engineers.

To truly operate efficiently and maximize the value of the cloud takes multiple teams with various backgrounds working together to design a system catered to your specific business needs. Some best practices are to establish a review cadence based on how fast you are building and spending in the cloud. The Iron Triangle is a commonly used framework that measures cost vs. speed vs. quality. You can work with your teams to set up an agreed-upon framework that works for your business. From there, you can either tighten your belt, or invest more.

The tools of the cost optimization trade

Once you have a firm grasp on how to approach cost optimization in the cloud, it's time to think about the various tools at your disposal. At a high level, cost management on Google Cloud relies on three broad kinds of tools.

1. **Cost visibility**—this includes knowing what you spend in detail, how specific services are billed, and the ability to display how (or why) you spent a specific amount to achieve a business outcome. Here, keep in mind key capabilities such as the ability to create shared accountability, hold frequent cost reviews, analyze trends, and visualize the impact of your actions on a near-real-time basis. Using a standardized strategy for organizing your resources, you can accurately map your costs to your organization's operational structure to create a showback/chargeback model. You can also use cost controls like budget alerts and quotas to keep your costs in check over time.

2. **Resource usage optimization**—this is reducing waste in your environment by optimizing usage. The goal is to implement a specific set of standards that draws an appropriate intersection between cost and performance within an environment. This is the lens to look through when reviewing whether there are idle resources, better services on which to deploy an app, or even whether launching a custom VM shape might be more appropriate. Most companies that are successful at avoiding waste are optimizing resource usage in a decentralized fashion, as individual application owners are usually the best equipped to shut down or resize resources due to their intimate familiarity with the workloads. In addition, you can use Recommender to help detect issues like under- or over-provisioned VM instances or idle resources. Enabling your team to surface these recommendations automatically is the aim of any great optimization effort.

3. **Pricing efficiency**—this includes capabilities such as sustained use discounts, committed use discounts, flat-rate pricing, per-second billing or other volume discounting features that allow you to optimize rates for a specific service. These capabilities are best leveraged by more centralized teams within your company, such as a Cloud Center of Excellence (CCoE) or FinOps team that can lower the potential for waste while optimizing coverage across all business units. This is something to continue to review both pre-cloud migration as well as regularly once you go live.

Considering both people and processes will go a long way toward making sure your standards are useful and aligned to what your business needs. Similarly, understanding Google Cloud's cost visibility, resource usage optimization, and pricing efficiency features will give you the tools you need to optimize costs across all your technologies and teams.

How to prioritize recommendations

With lots of competing initiatives, it can be difficult to prioritize cost optimization recommendations and ensure your organization is making the time to review these efforts consistently. Having visibility into the amount of engineering effort as well as potential cost savings can help your team establish its priorities. Some customers focus solely on innovation and speed of migration for years on end, and over time their bad optimization habits compound, leading to substantial waste. These funds could have gone towards developing new features, purchasing additional infrastructure, or hiring more engineers to improve their feature development velocity. It's important to find a balance between cost and velocity and understand the ramifications of leaning too far in one direction over another.

To help you prioritize one cost optimization recommendation over another, we tag every recommendation we make with our customers with two characteristics:

- **Effort**: Estimated level of work (in weeks) required by the customer to coordinate the resources and implement a cost optimization recommendation.
- **Savings**: Amount of estimated potential savings (in percentage per service) that customers will realize by implementing a cost optimization recommendation.

Effort	Duration	Savings	% per service
Low	Up to 2 weeks	Low	0-10%
Medium	2-6 weeks	Medium	10-20%
High	More than 6 weeks	High	20%+

While it's not always possible to estimate with pinpoint accuracy how much a cost savings measure will save you before testing, it's important to try and make an educated guess for each effort. For instance, knowing that a certain change could potentially save you 60% on your Cloud Storage for project X should be enough to help with the prioritization matrix and establishing engineering priorities with your team. Sometimes you can estimate actual savings. Especially with purchasing options, a FinOps team can estimate the potential savings by taking advantage of features like committed use discounts for a specific amount of their infrastructure. By performing this exercise, you want the team to be able to make informed decisions on where engineering is going, so they can focus their energy from a culture standpoint.

Chapter 2
Optimizing compute costs

The three cost optimization principles we laid out in the introduction—cost visibility, usage optimization, and pricing efficiency—can be applied as you're evaluating and setting up standards for various parts of your cloud infrastructure. We'll start with compute, the backbone of cloud infrastructure.

When customers migrate to Google Cloud, their first step is often to adopt Compute Engine, which makes it easy to procure and set up virtual machines (VMs) in the cloud that provide large amounts of computing power. Launched in 2012, Compute Engine offers multiple machine types, many innovative features, and is available at publication time in 23 regions and 70 zones.

Compute Engine's predefined and custom machine types make it easy to choose VMs closest to your on-premises infrastructure, accelerating the workload migration process cost effectively. Cloud allows you the pricing advantage of "pay as you go" and also provides significant savings as you use more compute with sustained use discounts. Here are some recommendations we've developed based on our experience working with enterprise customers to analyze their monthly spend and find optimization opportunities.

Getting ready to save

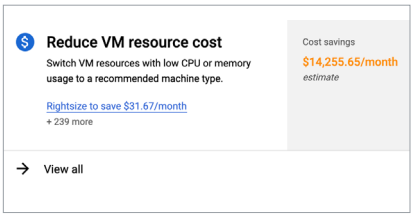
When you're looking to optimize, you should familiarize yourself with the VM instance pricing page—required reading for anyone who needs to understand the Compute Engine billing model and resource-based pricing in detail. This will help greatly in the journey to enhance your cost visibility.

The next important step to gain visibility into your Compute Engine costs is to use Cloud Billing reports in the Google Cloud Console, and customize your views based on filtering and grouping by projects, labels, and more. Here, you'll find information about the various Compute Engine machine types, committed use discounts, and how to view your usage, among other things. If you need an advanced view, you can even export Compute Engine usage details to BigQuery for more granular analysis. This allows you to query the datastore to understand your project's vCPU usage trends and how many vCPUs can be reclaimed. If you have defined thresholds for the number of cores per project, usage trends can help you spot anomalies and take proactive actions. These actions could be rightsizing the VMs or reclaiming idle VMs.

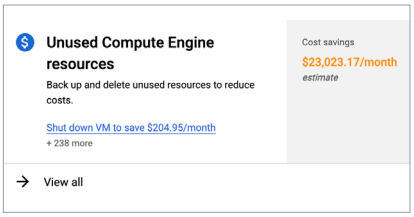
Now, with better cost visibility available, let's go over the five ways you can optimize your Compute Engine resources that we believe will offer the most immediate benefit.

1. Only pay for compute you need

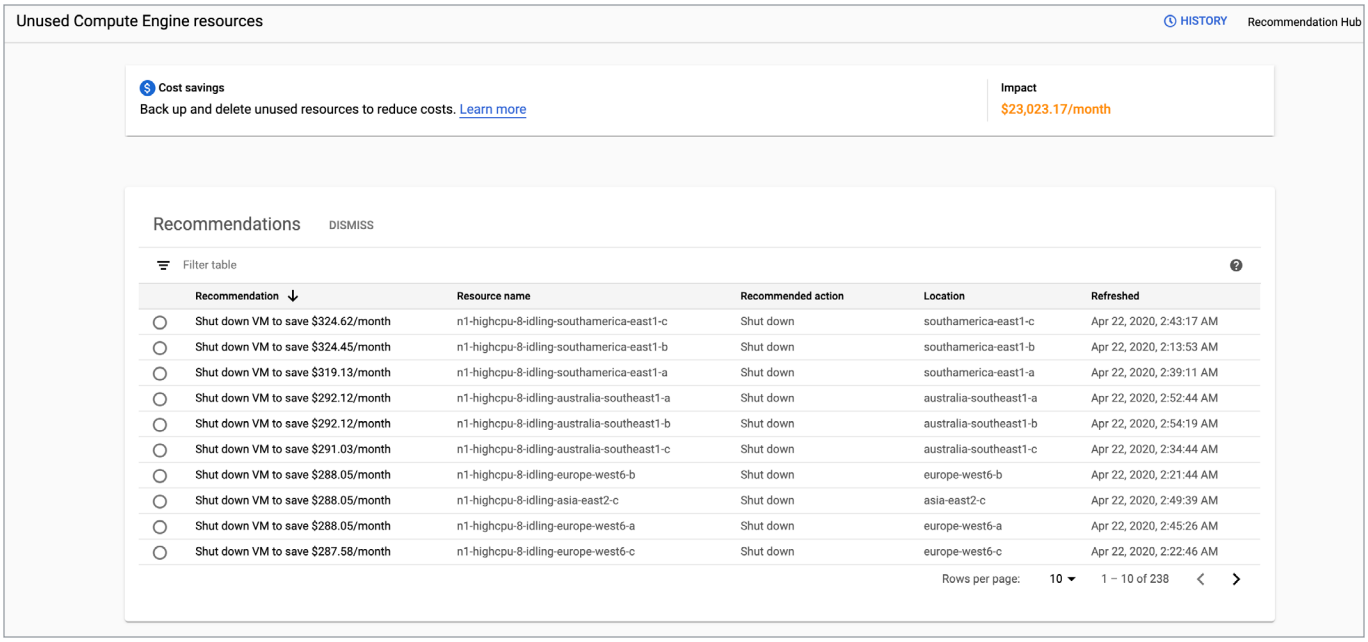
Effort ●●○ **Savings** ●●●



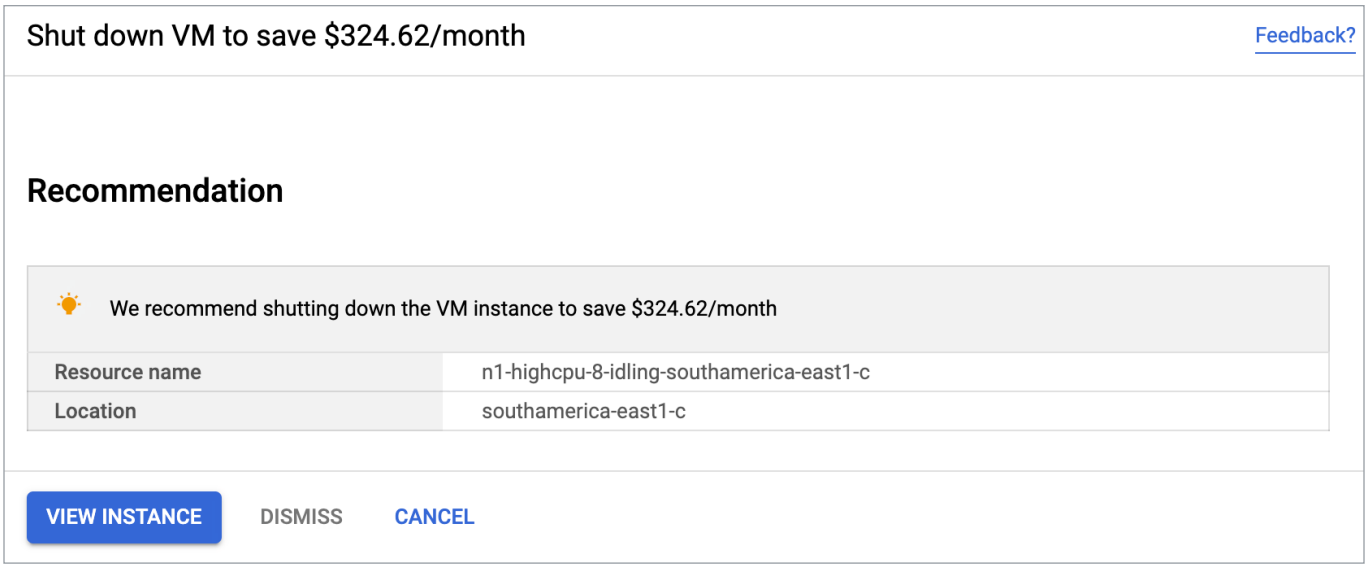
Identify idle VMs (and disks): The easiest way to reduce your Google Cloud bill is to get rid of resources that are no longer being used. Think about those proof-of-concept projects that have since been deprioritized, or zombie instances that nobody bothered to delete. Google Cloud offers several Recommenders that can help you optimize these resources, including an idle VM recommender that identifies inactive VMs and persistent disks based on usage metrics.



Always tread carefully when deleting a VM, though. Before deleting a resource, ask yourself, "What potential impact will deleting this resource have and how can I recreate it, if necessary?" Deleting instances gets rid of the underlying disk(s) and all of its data. One best practice is to take a snapshot of the instance before deleting it. Alternatively, you can choose to simply stop the VM, which terminates the instance, but keeps resources like disks or IP addresses until you detach or delete them.



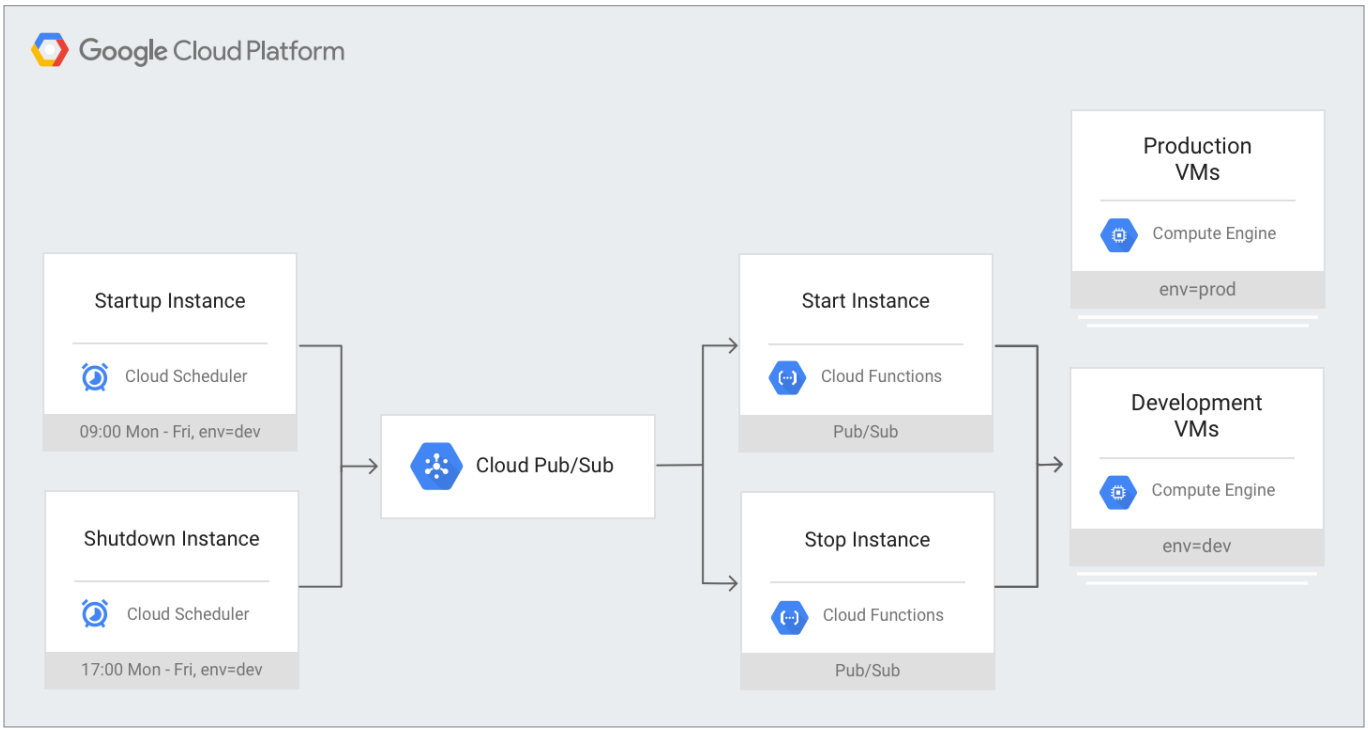
Google Cloud's Recommenders offer specific guidance on potential cost savings.



Recommenders can offer detailed tips on shutting down VMs.

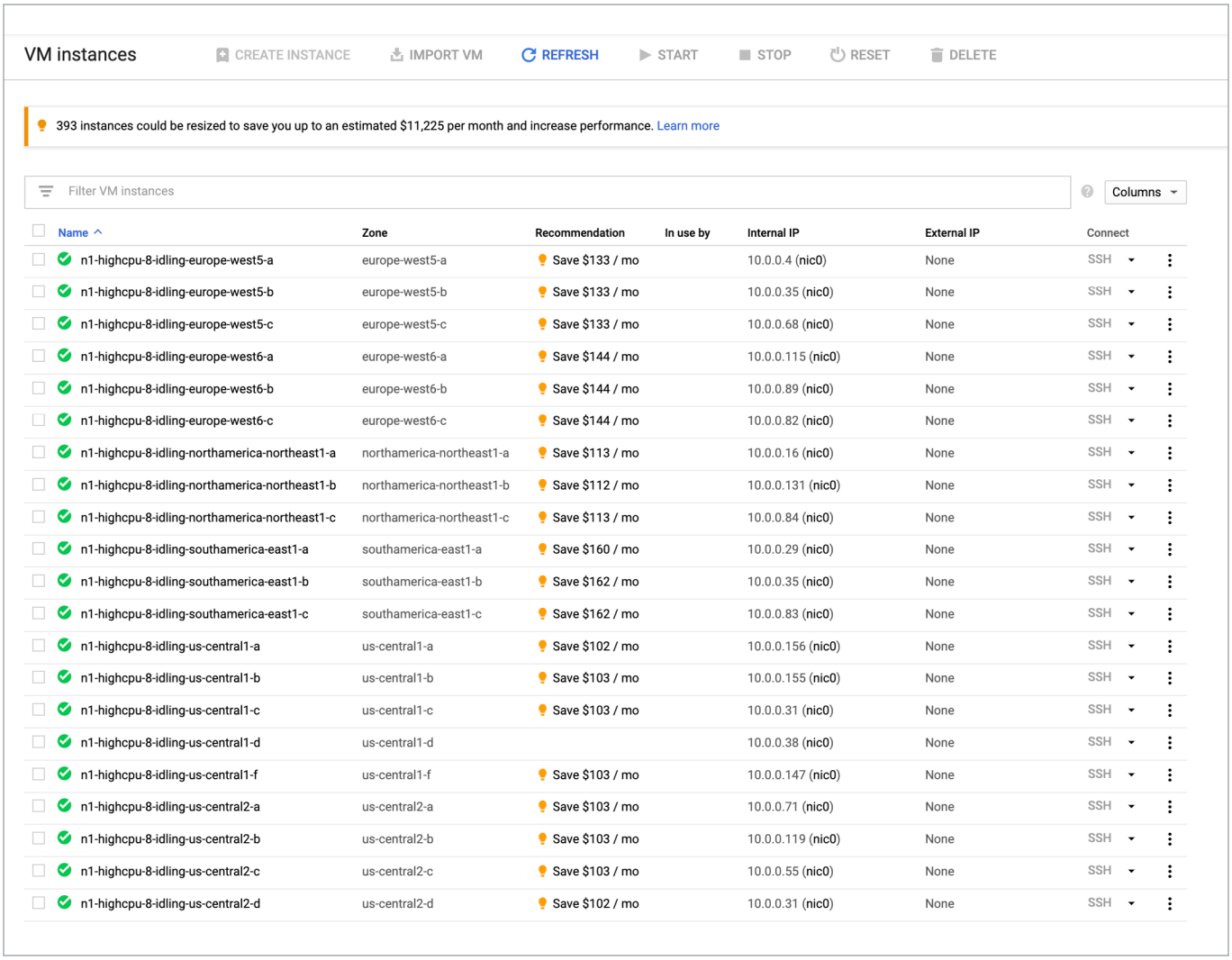
For more info, read the Recommender documentation. And stay tuned as we add more usage-based recommenders to the portfolio.

Schedule VMs to auto start and stop: The benefit of a platform like Compute Engine is that you only pay for the compute resources that you use. Production systems tend to run 24/7; however, VMs in development, test, or personal environments tend to only be used during business hours, and turning them off can save you a lot of money! For example, a VM that runs for 10 hours per day, Monday through Friday, costs 75% less to run per month compared to leaving it running.



it's worth exploring how VMs are set up and sized across your cloud infrastructure to find cost savings.

Rightsize VMs: On Google Cloud, you can already get significant savings by creating custom machine types with the right amount of CPU and RAM to meet your needs. But workload requirements can change over time. Instances that were once optimized may now be serving fewer users and traffic. To help, our rightsizing recommendations can show you how to effectively downsize your machine type based on changes in vCPU and RAM usage. These rightsizing recommendations for your instance's machine type (or managed instance group) are generated using system metrics gathered by Cloud Monitoring over the previous eight days.



Explore rightsizing recommendations for cloud VMs.

If your organization uses infrastructure as code to manage your environment, check out this guide, which will show you how to deploy VM rightsizing recommendations at scale.

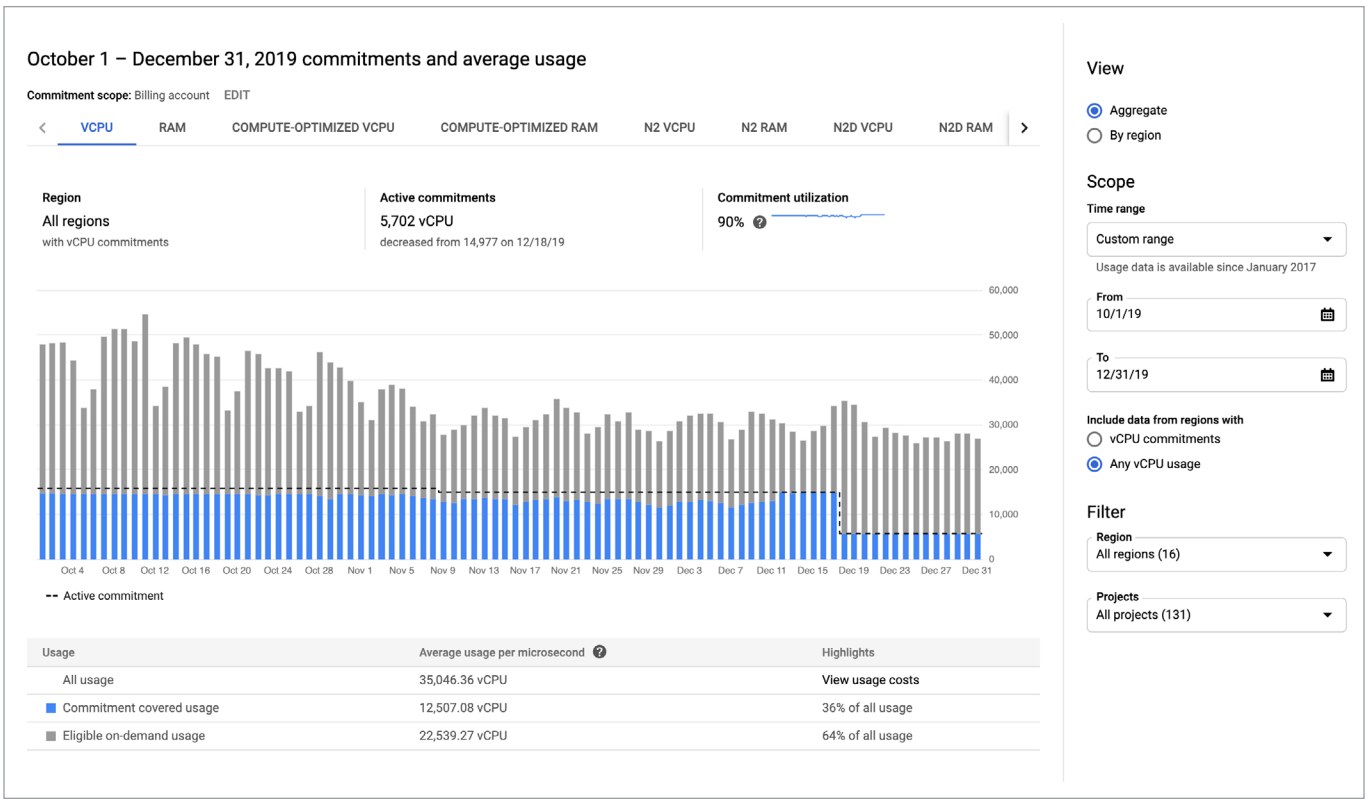
2. Purchase commitments

Effort ●○○ **Savings** ●●●

Pricing efficiency is another key concept to employ as part of a cloud optimization effort. The ability to purchase commitments and receive the related discounts is a huge step forward to ensuring you are truly optimizing your cloud costs.

Our customers have diverse workloads running on Google Cloud, with differing availability requirements. Many customers follow a 70/30 rule when it comes to managing their VM fleet—they have constant year-round usage of about 70%, and a seasonal burst of about 30% during holidays or special events.

If this sounds like you, you are probably provisioning resources for peak capacity. However, after migrating to Google Cloud, you can baseline your usage and take advantage of deeper discounts for compute workloads. Committed use discounts are ideal if you have a predictable steady-state workload, as you can purchase a one- or three-year commitment in exchange for a substantial discount on your VM usage. We recently released a committed use discount analysis report in the Cloud Console that helps you understand and analyze the effectiveness of the commitments you've purchased and even estimate what your resource floor looks like based on historical data.



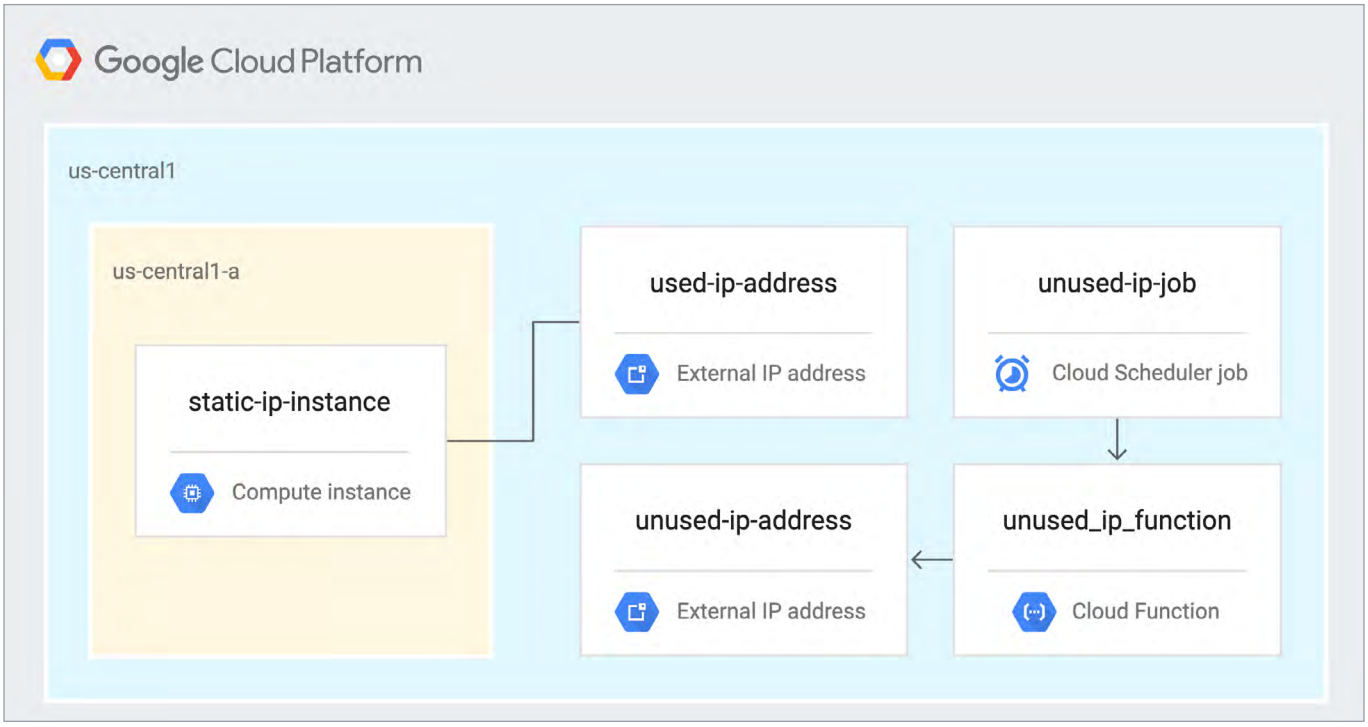
Committed use discount reports can display possible savings.

In addition to this, discuss your usage with your internal team and get a sense of whether or not a committed use discount makes sense for your workload. You can work proactively with them to increase your committed use discount coverage and maximize your savings.

3. Automate cost optimizations

Effort ●●● **Savings** ●●●

The best way to make sure that your team is always following cost optimization best practices is to automate them, reducing manual intervention. Automation is greatly simplified using a label—a key-value pair applied to various Google Cloud services. For example, you could label instances that only developers use during business hours with "env: development." You could then use Cloud Scheduler to schedule a serverless Cloud Function to shut them down over the weekend or after business hours, then restart them when needed. Use this architecture diagram and code samples to do this yourself. This is a huge leap forward in your ability to optimize your resource usage.



Using labels can automate cost optimizations.

Using Cloud Functions to automate the cleanup of other Compute Engine resources can also save you engineering time and money. For example, customers often forget about unattached (orphaned) persistent disks, or unused IP addresses. These accrue costs, even if they are not attached to a virtual machine instance. VMs with the deletion rule option set to "keep disk" will retain persistent disks even after the VM is deleted. That's great if you need to save the data on that disk for a later time, but those orphaned persistent disks can add up quickly. This Google Cloud solutions article describes the architecture and sample code for using Cloud Functions, Cloud Scheduler, and Cloud Monitoring to automatically look for these orphaned disks, take a snapshot of them, and remove them. This solution can be used as a blueprint for other cost automations such as cleaning up unused IP addresses or stopping idle VMs.

4. Use preemptible VMs

Effort ●●● **Savings** ●●●

If you have workloads that are fault-tolerant, like HPC, big data, media transcoding, CI/CD pipelines, or stateless web applications, using preemptible VMs to batch-process them can provide massive cost savings, using both resource usage optimization and pricing efficiency strategies. For example, our customer Descartes Labs reduced their analysis costs by more than 70% by using preemptible VMs to process satellite imagery and help businesses and governments predict global food supplies.

Preemptible VMs are short-lived—they run a maximum of 24 hours and may be shut down before the 24-hour mark as well. A 30-second preemption notice is sent to the instance when a VM needs to be reclaimed, and you can use a shutdown script to clean up in that 30-second period. Be sure to fully review the entire list of stipulations when considering preemptible VMs for your workload. All machine types are available as preemptible VMs, and you can launch one simply by adding "-preemptible" to the gcloud command line or selecting the option from the Cloud Console.

Using preemptible VMs in your architecture is a great way to scale compute at a discounted rate, but you need to be sure that the workload can handle the potential interruptions if the VM needs to be reclaimed. One way to handle this is to ensure your application is checkpointing as it processes data—i.e., that it's writing to storage outside the VM itself, like Google Cloud Storage or a database. As an example, try this sample code for using a shutdown script to write a checkpoint file into a Cloud Storage bucket. For web applications behind a load balancer, consider using the 30-second preemption notice to drain connections to that VM so the traffic can be shifted to another VM. Some customers also choose to automate the shutdown of preemptible VMs on a rolling basis before the 24-hour period is over to avoid having multiple VMs shut down at the same time if they were launched together.

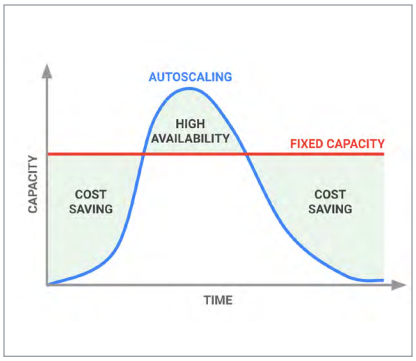
You don't have to limit preemptible VMs to a Compute Engine environment. GPUs, GKE clusters and secondary instances in Dataproc can also use preemptible VMs. You can also reduce your Cloud Dataflow batch analytics costs by using Flexible Resource Scheduling to supplement regular instances with preemptible VMs.

5. Try autoscaling

Effort ●○○ **Savings** ●●○

Another great way to save on costs is to run only as much capacity as you need when you need it. This is a key tenet of resource usage optimization. As we mentioned earlier, typically around 70% of capacity is needed for steady-state usage, but when you need extra capacity, it's critical to have it available.

In an on-prem environment, you need to purchase that extra capacity ahead of time, wait for it to be shipped, and then rack and stack it. In the cloud, you can use autoscaling to automatically flex to increased capacity only when you need it. Compute Engine managed instance groups are what give you this autoscaling capability in Google Cloud. You can scale up gracefully to handle an increase in traffic, and then automatically scale down again when the need for instances goes down (downscaling). You can scale based on CPU utilization, HTTP load balancing capacity, or Cloud Monitoring metrics. This gives you the flexibility to scale based on what matters most to your application.



High costs do not compute

As we've shown above, there are many ways to optimize your Compute Engine costs. Monitoring your environment and understanding your usage patterns is key to understanding the best options to start with, taking the time to model your baseline costs up front. Then, there are a wide variety of strategies to implement depending on your workload and current operating model.

Chapter 3
Optimizing storage costs

Whether you're part of a multi-billion dollar conglomerate trying to review sales data from the first half of the year, or you're just trying to upload a video of your cat playing the piano, you need someplace to store that data.

For Google Cloud customers, that usually means Cloud Storage, our unified object store. Cloud Storage features a robust API that lets you integrate with a multitude of services. It also includes features to help you follow the cost optimization principles of usage optimization, cost visibility, and price efficiency. While storing an object in the cloud in itself is an easy task, making sure you have the soundest approach for the situation you are in requires a bit more forethought.

One of the benefits of having a scalable, limitless storage service is that, much like an infinitely scalable attic in your house, there are going to be some boxes and items (or buckets and objects) that you could easily hold on to—but that you really shouldn't. Storing these items incur a cost over time, and whether you need them for business purposes or are just holding onto them on the off chance that they might someday be useful (like those wooden nunchucks you love), the first step is creating a practice around how to identify the usefulness of an object or bucket to your business. So let's get the broom and dustpan, and get to work!

Having a scalable, limitless storage service means there are going to be some buckets and objects that you could easily hold on to, but shouldn't.

Cleaning up your storage when you're moving to cloud

There are multiple factors to consider when looking into storage cost optimization. The trick here is to ensure that you don't negatively impact performance and that you don't throw anything out that may need to be retained for future purposes, whether that be for compliance, legal, or simply business value reasons. With data emerging as a top business commodity, you'll want to use appropriate

storage classes in the near term as well as for longitudinal analysis. And Cloud Storage offers a multitude of storage classes to choose from, all with varying costs, durability, and resiliency.

When it comes to cloud architecture, there's rarely a one-size-fits-all approach. However, there are some recurring themes we have noticed as we work alongside our customers. These lessons learned can apply to any environment, whether you're storing images or building advanced machine learning models.

When it comes to cloud architecture, there's rarely a one-size-fits-all approach.

The natural starting point is to first understand "What costs me money?" when using Cloud Storage. The pricing page is incredibly useful, but when analyzing your Cloud Storage usage, you also need to consider:

1. Retention
2. Access patterns
3. Performance

There can be many additional use cases with cost implications, but we'll focus on recommendations around these themes. Here are more details on each.

Retention considerations and tips

Effort ●○○ **Savings** ●●○

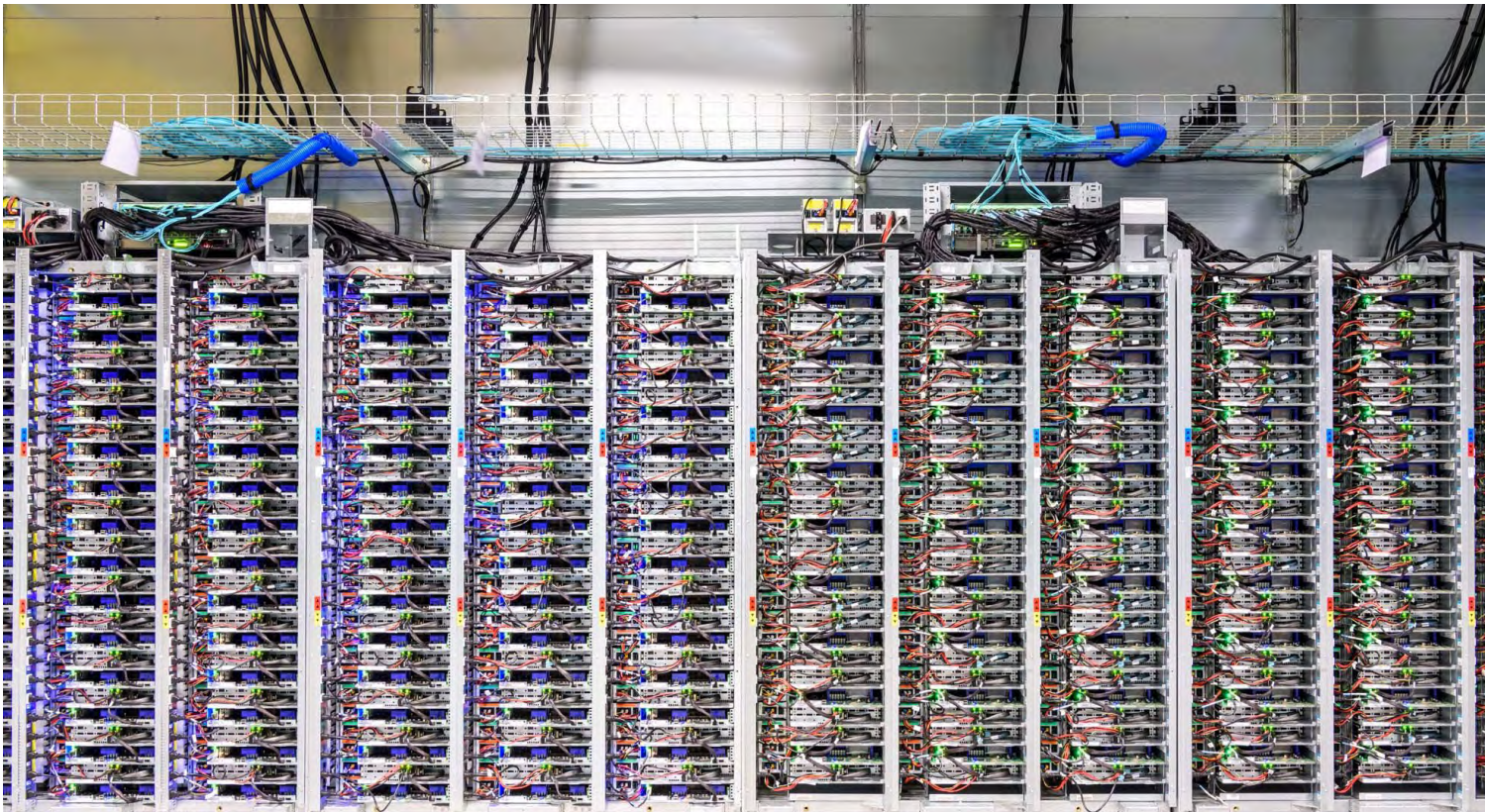
The first thing to consider when looking at a data type is its retention period. Asking yourself questions like "Why is this object valuable?" and "For how long will this be valuable?" are critical to help determine the appropriate lifecycle policy. Setting a lifecycle policy lets you tag specific objects or buckets and creates an automatic rule that will delete or even transform storage classes for that particular object or bucket type based on a set of conditions. Think of this as your own personal butler who will systematically ensure that your attic is organized and clean—except instead of costing money, this butler is saving you money for these operations.

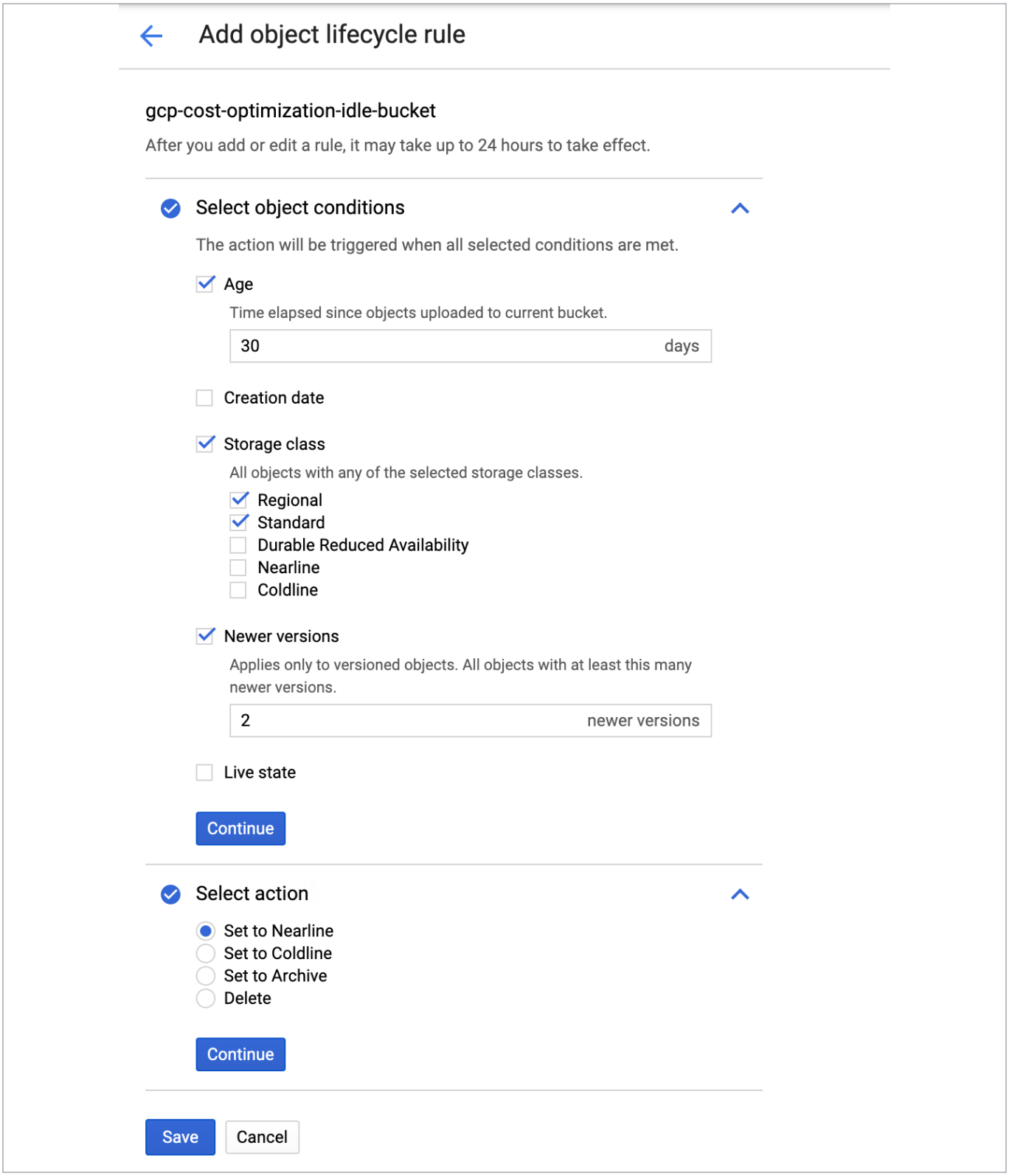
The first thing to consider when looking at a data type is its retention period.

We see customers use lifecycle policies in a multitude of ways with great success. A great application is for compliance in legal discovery. Depending on your industry and data type, there are laws that regulate the data types that need to be retained and for how long. Using a Cloud Storage lifecycle policy, you can instantly tag an object for deletion once it has met the minimum threshold for legal compliance needs, ensuring you aren't charged for retaining it longer than is needed and you don't have to remember which data expires when.

We see customers use lifecycle policies in a multitude of ways with great success.

Within Cloud Storage, you can also set policies to transform a storage type to a different class. This is particularly useful for data that will be accessed relatively frequently for a short period of time, but then won't be needed for frequent access in the long term. You might want to retain these particular objects for a longer period of time for legal or security purposes, or even general long-term business value. A good place to put this in practice is in a lab environment. Once you complete an experiment, you likely want to analyze the results quite a bit in the near term, but in the long term won't access that data very frequently. Having a policy set up to convert this storage to nearline or coldline storage classes after a month is a great way to save on its long-term data costs.





Set Cloud Storage policies to move data automatically.

Another common source of waste in storage environments is duplicate data. Of course, there are times when it's necessary. For instance, you may want to duplicate a dataset across multiple geographic regions so that local teams can access it quickly. However, in our experience working with customers, a lot of duplicate data is the result of lax version control, and the resulting duplicates can be cumbersome and expensive to manage.

Luckily, there are lots of ways to prevent duplicate data, as well as tools to prevent data from being deleted in error. Here are a few things to consider:

- If you're trying to maintain resiliency with a single source of truth, it may make more sense to use a multi-region bucket rather than creating multiple copies in various buckets. With this feature, you get geo-redundancy enabled for objects stored. This ensures your data is replicated asynchronously across two or more locations, and protects against regional failures in the event of a natural disaster.

- A lot of duplicate data comes from not properly using the Cloud Storage object versioning feature. Object versioning prevents data from being overwritten or accidentally deleted, but the duplicates it creates can really add up. Do you really need five copies of your data? One might be enough as long as it's protected. Worried you won't be able to roll back? You can set up object versioning policies to ensure you have an appropriate number of copies. Still worried about losing something accidentally? Consider using the bucket lock feature, which helps ensure that items aren't deleted before a specific date or time. This is really useful for demonstrating compliance with several important regulations. In short, if you use object versioning, there are several features you can use to keep your data safe without wasting space unnecessarily.

Access pattern considerations and tips

Effort ●●● **Savings** ●●●

Cloud Storage offers a variety of storage classes—standard, nearline, coldline and archival, all with varying costs and their own best-fit use cases. If you only use the standard class, it might be time to take a look at your workloads and reevaluate how frequently your data is being accessed. In our experience, many companies use standard class storage for archival purposes, and could reduce their spend by taking advantage of nearline or coldline class storage. And in some cases, if you are holding onto objects for cold-storage use cases like legal discovery, the archival class of storage might offer even more savings.

Many companies could reduce their spend by taking advantage of nearline or coldline class storage.

The ability to transform objects into lower-cost storage classes is a powerful tool, but one that must be used with caution. While long-term storage is cheaper to maintain for an object that is accessed at a lower frequency, there will be additional charges incurred if you suddenly need to frequently access the data or metadata that has been moved to a colder storage option. There are also cost implications when looking to remove that data from a particular storage class. For instance, there's currently a minimum time of 30 days for an object to sit in nearline storage. If you need to access that data with an increased frequency, you can make a copy in a regional storage class instead to avoid increased access charges.

When considering cost savings opportunities, you should also think about whether your data will need to be accessed in the long term, and how frequently it will be accessed if it does become valuable again. For example, if you are a CFO looking at a quarterly report on cloud expenses and only need to pull that information every three months, you might not need to worry about the increased charges accrued for the retrieval of that data, because it will still be cheaper than maintaining the storage in a regional bucket year round. But some retrieval costs on longer-term storage classes can be substantial and should be carefully reviewed when making storage class decisions.

Performance considerations and tips

Effort ●●○ **Savings** ●○○

"Where is this data going to be accessed from?" is a major question to consider when you're considering performance and trying to establish the best storage class for your particular use case. Locality can directly influence how fast content is pushed to and retrieved from your selected storage location. For instance, a "hot object" with global utilization (such as a database that is accessed frequently, like your employee time-tracking application) would fit well in a multi-regional location, which enables an object to be stored in multiple locations. This can potentially bring the content closer to your end users as well as enhance your overall availability. Another example is a gaming application with a broad geo-distribution of users. This brings the content closer to the user for a better experience (less lag) and ensures that your last saved file is distributed across several locations, so you don't lose your hard-earned loot in the event of a regional outage.

One thing to keep in mind when considering this option is that storage in multi-regional locations allows for better performance and higher availability, but comes at a premium and could increase network egress charges, depending on your application's design. During the application design phase, this is an important factor to consider. Another option when you're thinking about performance is buckets in regional locations—a good choice if your region is relatively close to your end users. You can select a specific region that your data will reside in, and get guaranteed redundancy within that region. This location type is typically a safe bet when you have a team working in a particular area and accessing a dataset with relatively high frequency. This is the most commonly used storage location type that we see, as it handles most workloads' needs quite well. It's fast to access, redundant within the region, and affordable overall as an object store.

For something as simple-sounding as a bucket, cloud-based object storage actually offers vast amounts of possibility, all with varying cost and performance implications. As you can see, there are many ways to fine-tune your own company's storage needs to help save some space and some cash in a well thought-out, automated way. Google Cloud provides many features to help ensure you are getting the most out of your Google Cloud investment.

Chapter 4
Optimizing network costs

Every cloud deployment needs a network over which to move data. Without a network, you can't view cat videos or upload your selfies, much less allow microservices to talk to one another.

Google Cloud provides a global, scalable, flexible network for your cloud-based workloads and services, and how you use that network impacts four critical aspects of your deployment: cost, security, performance, and availability. To design a reliable, sound, yet cost-effective network architecture, you'll want multiple teams within the company to weigh in on these four elements to help determine your priorities. The following tips highlight a few considerations you should think about when architecting your network solution.

You'll want multiple teams within the company to weigh in on network architecture elements to help determine your priorities.

Understanding network traffic flows

The first step when reviewing your overall networking spend strategy is to understand what you're using—namely, what traffic is flowing in and out of your Google Cloud environment. This is easy to do with VPC Flow Logs, which keep a record of the network flows sent from and received by VM instances. Each flow log entry records details such as source IP, destination IP, and bytes sent and received for each network connection—exactly the type of information needed when trying to understand your network traffic. These logs are collected in Cloud Logging, and you can then export them to BigQuery to help visualize your trends.

Some of the use cases for VPC Flow Logs include network monitoring, forensics, real-time security analysis, and for today's purposes, cost optimization. And when it comes to optimizing networking spend, the most relevant information in VPC Flow Logs is:

- Traffic between regions and zones
- Traffic to specific countries on the Internet
- Top talkers

Here are step-by-step instructions on how to enable VPC Flow logs.

Identify your “top talkers”

Effort ●●● **Savings** ●●●

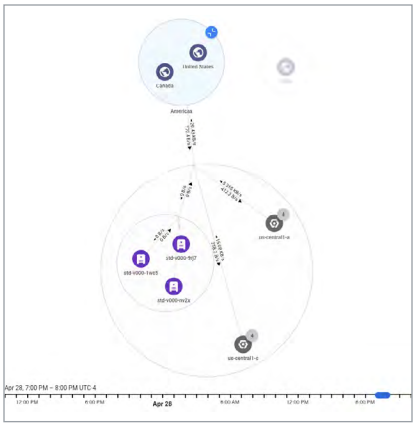
The information you get from VPC Flow Logs can help you determine where you might be able to save on your existing network costs. For example, geographic location is an important factor to consider when architecting for optimal spend. Not all network charges are created equal; different regions have varying network costs.

It’s also important to know your network layout and how traffic flows between your applications and users. Network Topology, a module of Network Intelligence Center, provides comprehensive visibility into your global Google Cloud deployment and its interaction with the public internet, including an organization-wide view of the topology and associated network performance metrics. This allows you to identify inefficient deployments and take necessary actions to optimize your regional and intercontinental network egress costs.

For general internet egress charges, i.e., a group of web servers that serve content to the internet, prices can vary depending on the region where those servers are located. For instance, the price per GB in us-central1 is cheaper than the price per GB in asia-southeast1. Another example is traffic flowing between Google Cloud regions, which can vary significantly depending on the location of those regions—even if it isn’t egressing out to the Internet. For example, the cost to synchronize data between asia-south1 (India) and asia-east1 (Taiwan) is five times as much as synchronizing traffic between us-east1 (South Carolina) and us-west1 (Oregon).

As well as regional considerations, it’s important to consider which zones your workloads are in. Depending on their availability requirements, you may be able to architect them to use intrazone network traffic at no cost. You read that right, at no cost! Consider your VMs communicating via public, external IP addresses, but that are in the same region or zone. By configuring them to communicate via their internal IP addresses, you can save on the cost of what you would have paid for that traffic communicating via external IP addresses.

Not all network charges are created equal; different regions have varying network costs.



Example topology using Network Topology, Network Intelligence Center

Keep in mind, you'll need to weigh any potential network cost savings with the availability implications of a single-zone architecture. Deploying to only a single zone is not recommended for workloads that require high availability, but it can make sense to have certain services use a virtual private cloud (VPC) network within the same zone. One example could be to use a single-zone approach in regions that have higher costs (Asia), but a multi-zone or multi-regional architecture in North America, where the costs are lower.

Once you have established your network costs for an average month, you may want to consider a few different approaches to better allocate spending. Some customers re-architect solutions to bring applications closer to their user base, and some employ Cloud CDN to reduce traffic volume and latency, as well as potentially take advantage of Cloud CDN's lower costs to serve content to users. Both of these are viable options that can reduce costs and/or enhance performance.

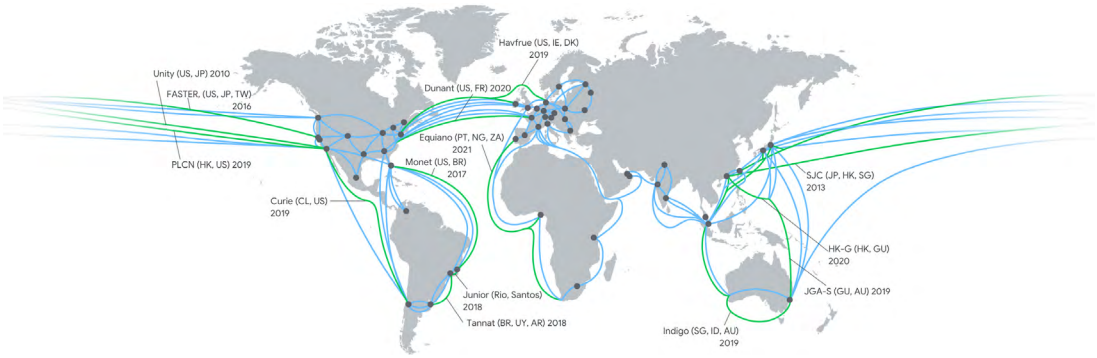
Deciding when to use a VPN

Effort ●●● **Savings** ●●○

Next in line when reviewing overall networking spend is total bytes transferred. Using VPC Flow Logs, you can see the "top talkers" within your environment. If you're pushing large amounts of data (think TBs/PBs), you want to ensure that you take advantage of any potential discounts you might be entitled to.

We have seen many customers who push large amounts of data on a daily basis from their on-premises environment to Google Cloud, either using a VPN or perhaps directly over the Internet (encrypted with SSL, hopefully!). Some customers, for example, have databases on dedicated, on-prem hardware, whereas their front-end applications are serving requests in Google Cloud. If this describes your situation, consider whether you should use a Dedicated Interconnect or Partner Interconnect. If you push large amounts of data on a consistent basis, it can be cheaper to establish a dedicated connection vs. accruing costs associated with your traffic traversing the public internet or using a VPN.

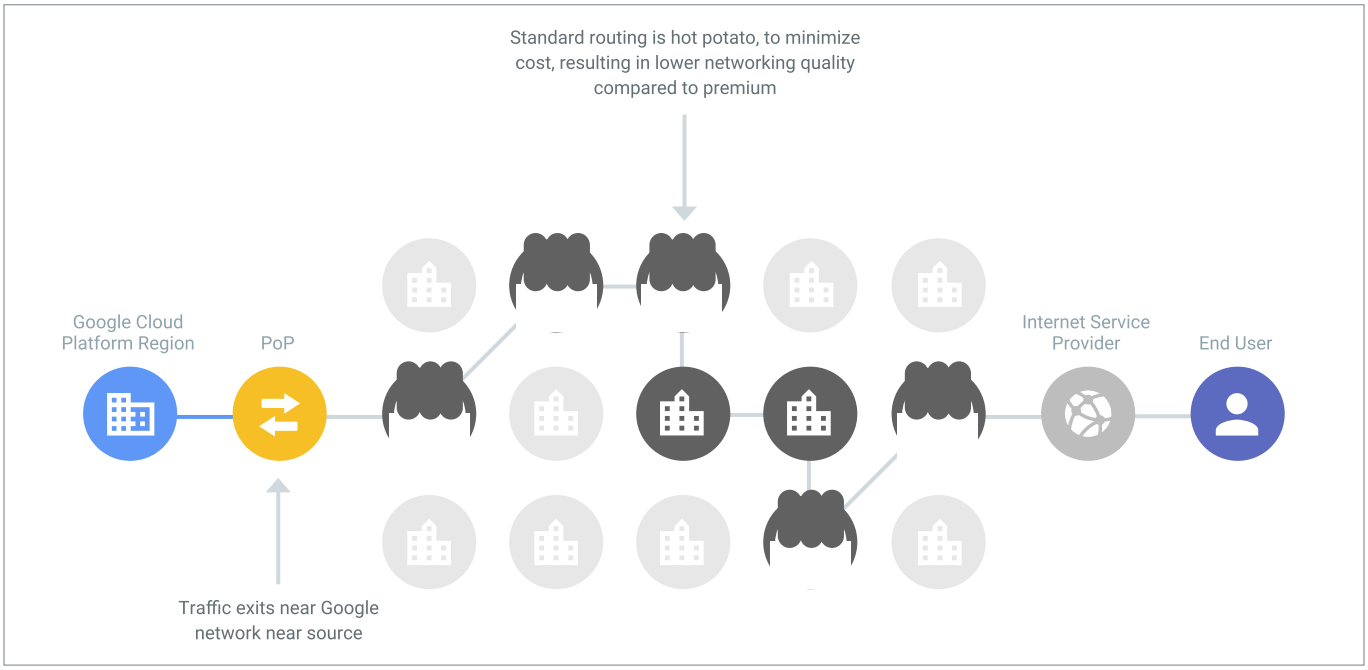
Check out the details of the architectural considerations to review when selecting an interconnect.



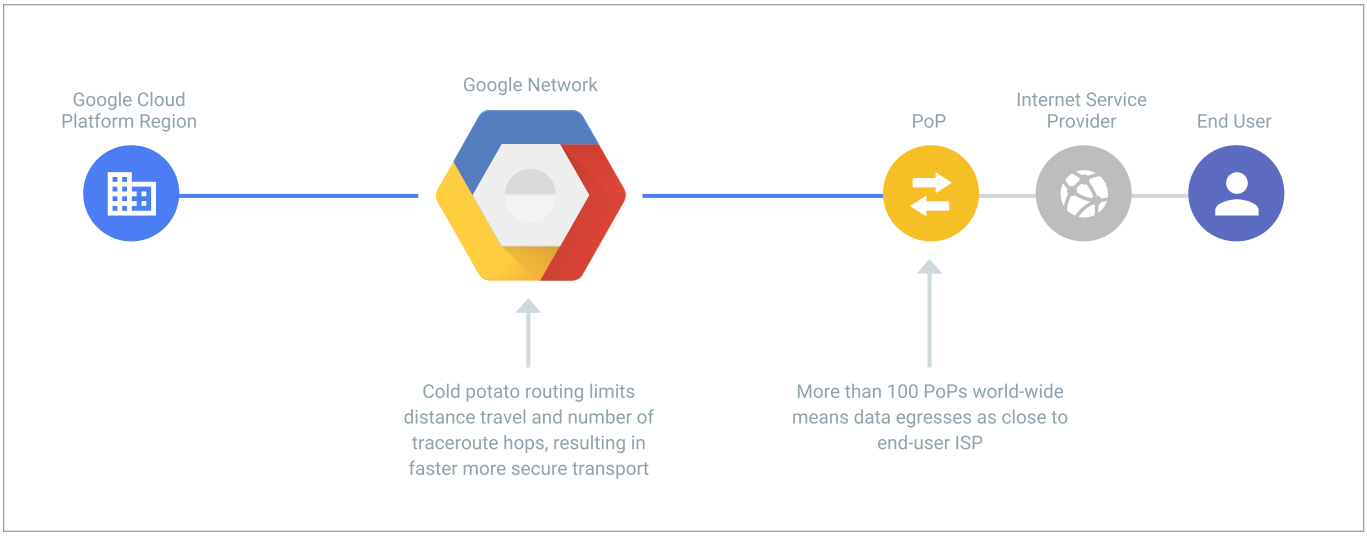
Your network optimized your way with tiers

Effort ●○○ **Savings** ●●○

One of Google Cloud's biggest differentiators is that you get access to Google's premium network backbone, which is used by default for all services. But you might not need that performance and low latency for all your services. An example might be the distribution of a daily sales report that doesn't need to be immediately available around the globe. With services for which you are willing to trade off between performance and cost, we offer Network Service Tiers.



Choosing the Standard networking tier can save on costs.



Choosing the Premium networking tier brings performance and low latency.

By choosing either Standard or Premium Tier, you can allocate the appropriate connectivity between your services, fine-tuning the network to the needs of your application and potentially reducing costs on services that might tolerate more latency and don't require an SLA.

There are some limitations when leveraging the Standard tier for its pricing benefits. At a high level, these include compliance needs around traffic traversing the public internet, as well as HTTP(S), SSL proxy, TCP proxy load balancing, or usage of Cloud CDN. After reviewing some of the recommendations, you'll be empowered to review your services with your team and determine whether you can benefit from lower Standard Tier pricing without impacting the performance of your external-facing services.

Optimizing usage for your network

Effort ●●○ **Savings** ●●○

The above topics are some of the larger levers you can pull when conducting a networking cost review. But overall you should ensure that you are taking advantage of one of the greatest cloud benefits: Pay only for what you use. With this in mind, we recommend reviewing the following to ensure you get the most out of your Google Cloud investment:

- **Cloud Logging—**You may not know it, but you do have control over network traffic visibility by filtering out logs that you no longer need. Check out some common examples of logs that you can safely exclude. The same applies to data access audit logs, which can be quite large and incur additional costs. For example, you probably don't need to log them for development projects. For VPC Flow Logs and Cloud Load Balancing, you can also enable sampling, which can dramatically reduce the volume of log traffic being written into the database. You can set this from 1.0 (100% of log entries are kept) to 0.0 (0%, no logs are kept). For troubleshooting or custom use cases, you can always choose to collect telemetry for a particular VPC network or subnet or drill down further to monitor a specific VM instance or virtual interface.

- **Private Access for enterprise or high-volume customers—** Leverage Private Google Access when possible to reduce cost and improve your security posture.

- **External IP addresses—**Starting in 2020, external IP addresses that don't fall under the Free Tier will incur a small cost. However, as a general security best practice, it's a good idea to use internal IP addresses where applicable. For information on how to migrate to internal IPs, refer to our guides for building internet connectivity for private VMs or setting up a private cluster on Google Kubernetes Engine.

Reviewing the above will ensure you are eliminating wasteful spending within your design, and also ensure that you are taking full advantage of your cloud-based solution. Balancing costs with performance, availability, and security is no simple feat, and often requires collaboration across multiple teams. We like to think that there are many approaches to consider, and more often than not, cost optimization is not so much a one-time review, but your application teams' philosophy. Find the methods that work best for your teams and company.

Chapter 5
Optimizing data analytics costs with BigQuery

Running and managing legacy data warehouses can be frustrating and time-consuming, especially now, where data is everywhere and in everything we do. Scaling systems to meet this increase in data has made it even more challenging to maintain daily operations. There's also the additional hassle of upgrading your data warehouse with minimal downtime and supporting ML and AI initiatives to meet business needs. We hear from our customers that they choose BigQuery, Google Cloud's serverless enterprise data warehouse, so they can focus on analytics and be more productive instead of managing infrastructure.

With BigQuery, you can run blazing fast queries, get real-time insights with streaming data, and start using advanced and predictive analytics with built-in machine learning capabilities. An Enterprise Strategy Group (ESG) analysis revealed that BigQuery can provide up to a 26% to 34% lower total cost of ownership (TCO) over a three-year period compared to other cloud data warehouse alternatives. But that doesn't mean there's no room for further optimizations for your data housed in BigQuery. Since cost is one of the prominent drivers behind technology decisions in this cloud computing era, the natural follow-up questions we hear from our customers are about billing details and how to continually optimize costs.

We've put together this list of actions you can take to help you optimize your costs—and in turn, business outcomes—based on our experiences and product knowledge. One particular benefit of optimizing costs in BigQuery is that because of its serverless architecture, those optimizations also yield better performance, so you won't have to make stressful tradeoffs of choosing performance over cost or vice versa.

Understanding the basics of pricing in BigQuery

Let's look at the pricing for BigQuery, then explore each billing subcategory to offer tips to reduce your BigQuery spending. For any location, the BigQuery pricing is broken down like this (and you'll find more details below):

- **Query processing**
  - On-demand: This is based on the amount of data processed by each query you run.
  - Flat-rate: This is best for customers who desire cost predictability. Customers purchase dedicated resources for query processing and are not charged for individual queries.
- **Storage**
  - Active storage: A monthly charge for data stored in tables or in partitions that have been modified in the last 90 days.
  - Long-term storage: A lower monthly charge for data stored in tables or in partitions that have not been modified in the last 90 days.
  - Streaming inserts

For BigQuery ML, see BigQuery ML pricing. For BigQuery Data Transfer Service, see BigQuery Data Transfer Service pricing.

Before diving in to those, here are the BigQuery operations that are free of charge in any location:

- Batch loading data into BigQuery
- Automatic re-clustering (which requires no setup and maintenance)
- Exporting data operation

```
-- No cost, since no tabl
DECLARE x DATE DEFAULT CU
-- Incurs the cost of sca
DECLARE y STRING DEFAULT
-- Incurs the cost of cop
-- table is created, you
-- script runs.
CREATE TEMP TABLE t AS SE
-- Incurs the cost of sca
SELECT column1 FROM t;
-- No cost, since y = 'fo
IF y = 'foo' THEN
  -- Incurs the cost of s
  -- y was equal to 'foo'
  SELECT * FROM dataset.c
ELSE
  -- Incurs the cost of s
  -- y was not equal to '
  UPDATE dataset.differen
  SET col = 10
  WHERE true;
END IF;
-- Incurs the cost of sca
-- iteration of the loop
WHILE x < (SELECT MIN(dat
  -- No cost, since the e
  SET x = DATE_ADD(x, INT
  -- No cost, since the e
  IF true THEN
    -- LEAVE has no assoc
    LEAVE;
  END IF;
  -- Never executed, sinc
  -- a cost.
  SELECT * FROM dataset.b
END WHILE;
```

- Deleting tables, views, partitions, functions and datasets
- Metadata operations
- Cached queries
- Queries that result in error
- Storage for first 10 GB of data per month
- Query data processed for first 1 TB of data per month (advantageous to users using on-demand pricing)

Understanding flat-rate vs. on-demand pricing

Effort ●○○ **Savings** ●●●

By default, BigQuery charges you variable on-demand pricing based on bytes processed by your queries. If you are a high-volume customer with stable workloads, you may find it more cost effective to switch from on-demand to flat-rate pricing, which gives you an ability to process unlimited bytes for a fixed predictable cost. When you enroll in flat-rate pricing, you purchase slot commitments—dedicated query processing capacity, measured in BigQuery slots. With the introduction of BigQuery Reservations, customers now have an easy and flexible self-service way to take advantage of BigQuery flat-rate pricing. A good starting point to decide how many slots to buy is to visualize your slot utilization for the last month using Cloud Monitoring.

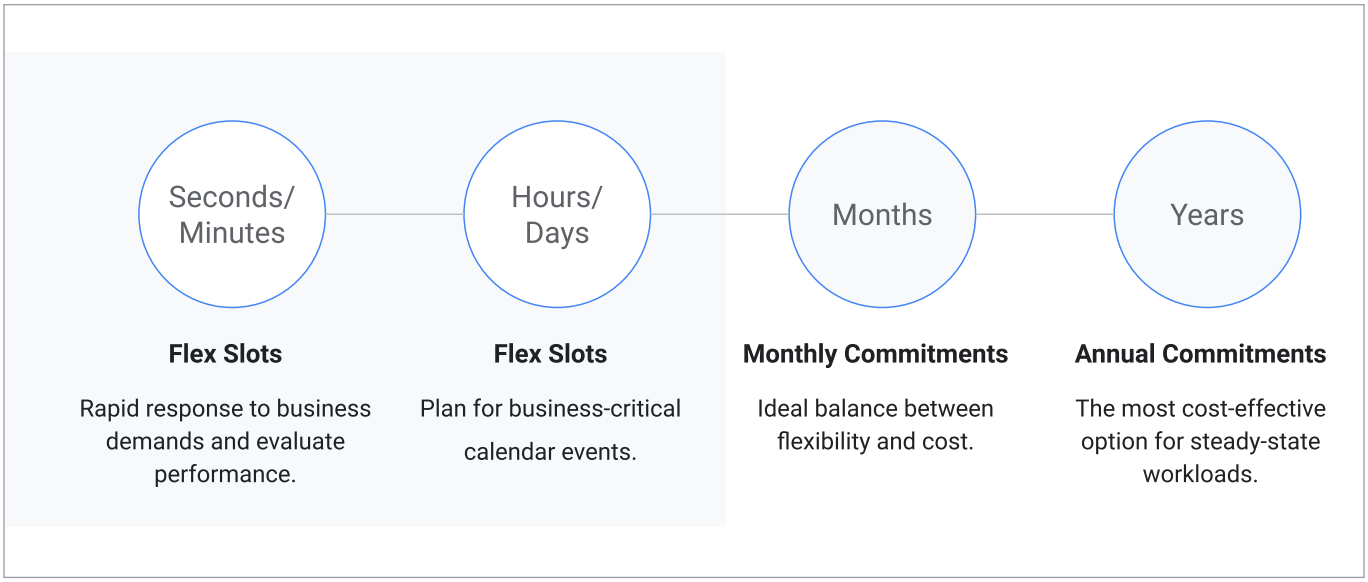
Note: If your queries exceed flat-rate capacity, BigQuery will run proportionally more slowly until the slots become available.

You might be tempted to think that you don't have to worry about query optimizations with flat-rate at all. The reality is that it still impacts performance. The faster your query (job) executes, the more jobs you will be able to complete in the same amount of time as with fixed slots. If you think about it, that's cost optimization in itself.

You might be tempted to think that you don't have to worry about query optimizations with flat-rate at all.

Buying too few slots can impact performance, while buying too many slots will introduce idle processing capacity, resulting in cost implications. In order to find your sweet spot, you can start with a monthly flat-rate plan, which allows more flexibility to downgrade or cancel after 30 days. Once you have a good enough ballpark estimate on the number of slots you need, switch to an annual flat-rate plan for further savings. In addition, BigQuery Reservations helps you use flat-rate pricing even more efficiently and plan your spending.

Given rapidly changing business requirements, we recently introduced Flex Slots, a new way to purchase BigQuery slots for durations as short as 60 seconds, on top of monthly and annual flat-rate commitments.

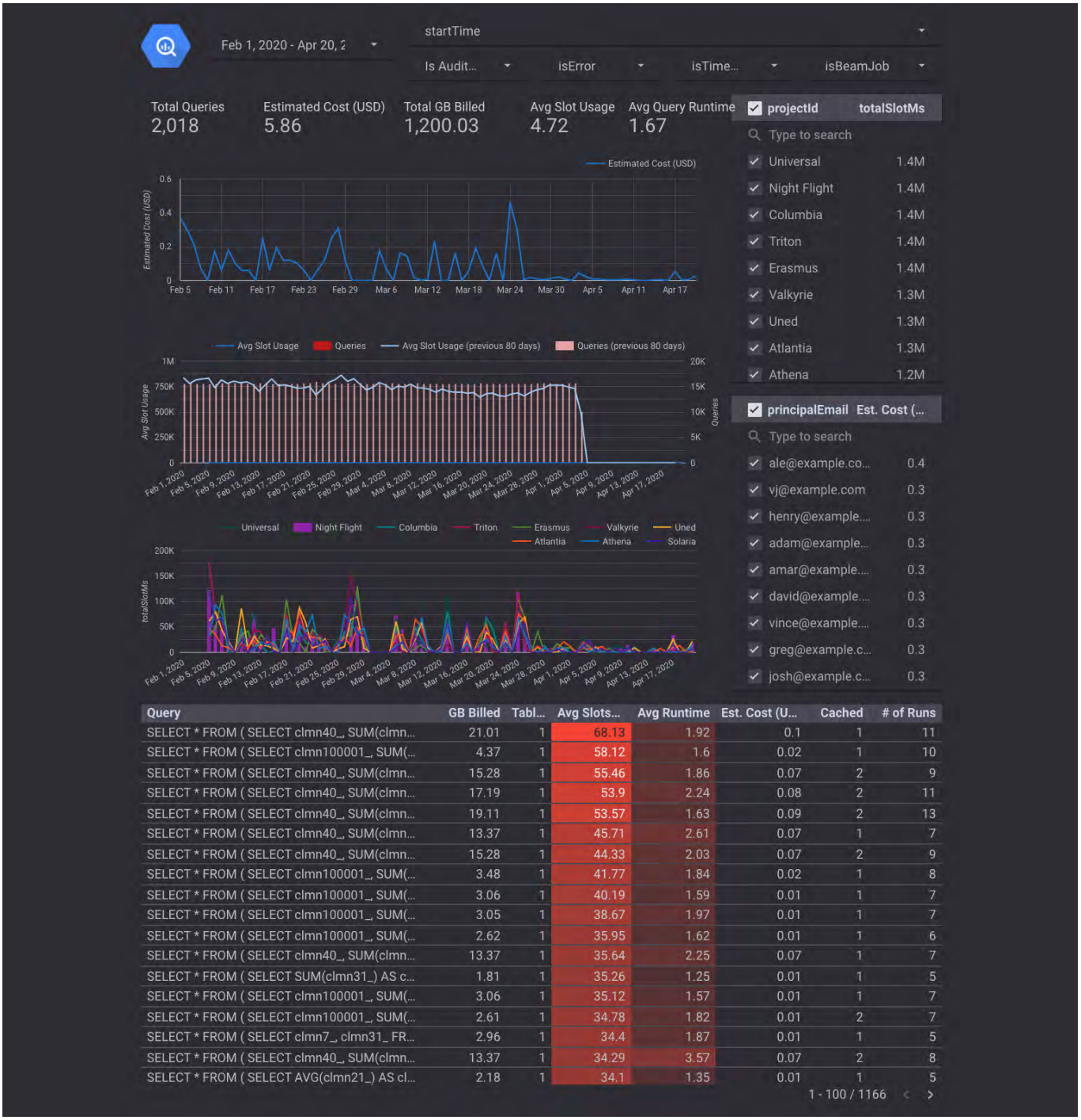


New Flex Slots offers flexibility to meet demand.

With this combination of on-demand and flat-rate pricing, you can respond quickly and cost-effectively to changing demand for analytics. Now that you understand the fundamentals of BigQuery pricing, let's take a look at how you get started with cost optimization.

Visualize your BigQuery costs

When you're getting started understanding costs within your organization, take a few minutes to run a quick report of your BigQuery usage for the past month to get a sense of your costs. You can use Billing Reports in the Cloud Console, or simply export your billing data into BigQuery. A detailed Data Studio dashboard is also available that allows you to identify costly queries so that you can optimize for cost and query performance. It will also provide insight into the usage patterns and resource utilization associated with your workload. Follow these step-by-step instructions to create a dashboard, as shown below.



BigQuery reports offer a look at usage.

Cost optimization techniques in BigQuery: query processing

You'll likely query your BigQuery data for analytics and to satisfy business use cases like predictive analysis or real-time inventory management.

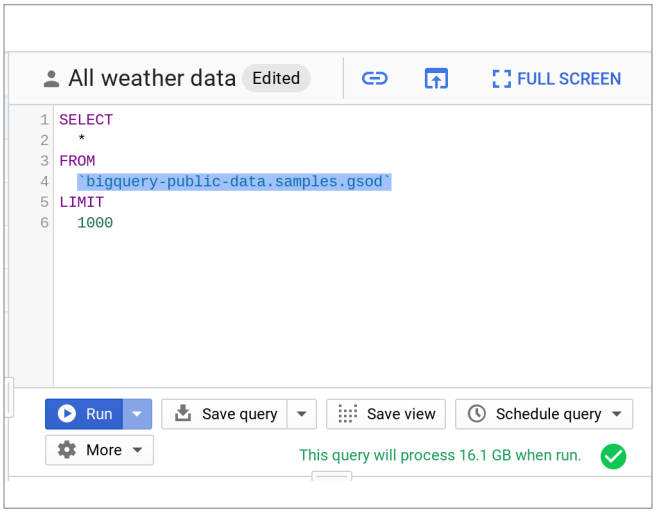
On-demand pricing is what most users and businesses choose when starting with BigQuery. You are charged for the number of bytes processed, regardless of the data housed in BigQuery or external data sources involved. There are some ways you can reduce the number of bytes processed. Let's go through the best practices to reduce the cost of running your queries, such as SQL commands, jobs, user-defined functions, and more.

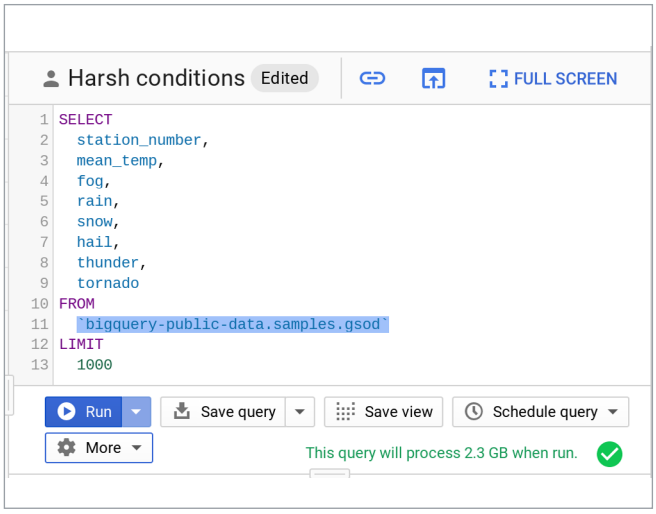
1. Only query the data you need. (We mean it!)

Effort ●○○ **Savings** ●○○

BigQuery can provide incredible performance because it stores data as a columnar data structure. This means SELECT * is the most expensive way to query data. This is because it will perform a full query scan across every column present in the table(s), including the ones you might not need. (We know the guilty feeling that comes with adding up the number of times you've used SELECT * in the last month.)

Let's look at an example of how much data a query will process. Here we're querying one of the public weather datasets available in BigQuery:





Querying public weather datasets in BigQuery

As you can see, by selecting the necessary columns, we can reduce the bytes processed by about eight-fold, which is a quick way to optimize for cost. Also note that applying the LIMIT clause to your query doesn't have an effect on cost.

If you do need to explore the data and understand its semantics, you can always use the no-charge data preview option.

Also remember you are charged for bytes processed in the first stage of query execution. Avoid creating a complex multistage query just to optimize for bytes processed in the intermediate stages, since there are no cost implications anyway (though you may achieve performance gains).

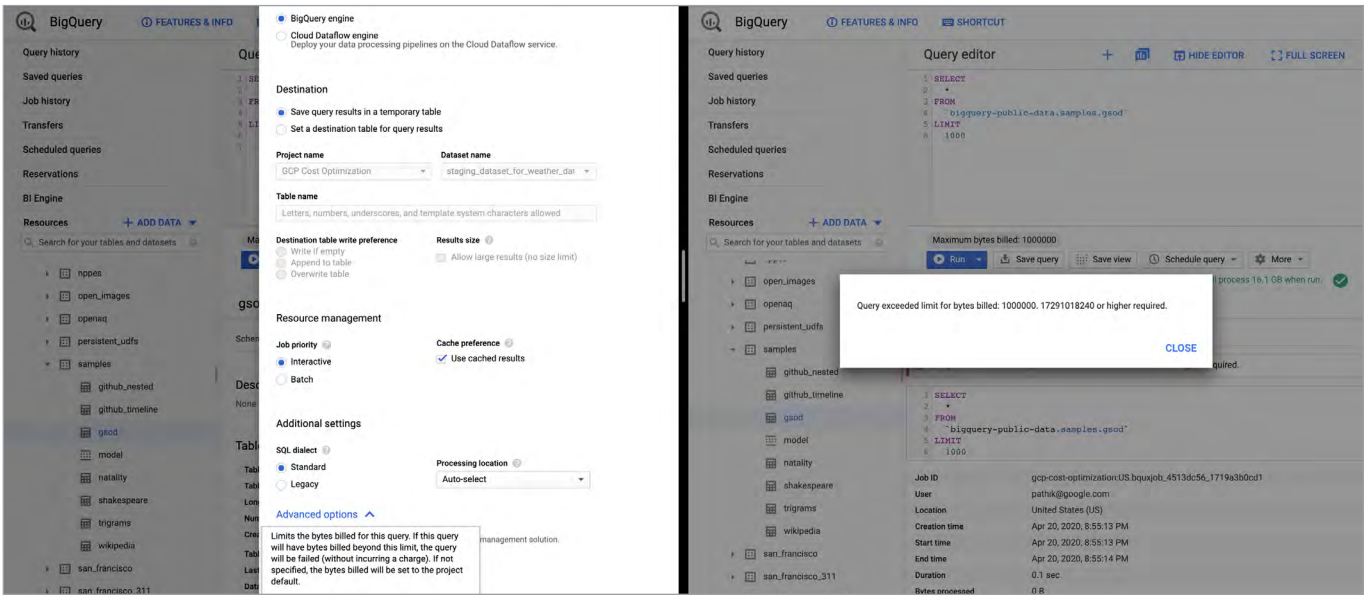
Filter your query as early and as often as you can to reduce cost and improve performance in BigQuery.

2. Set up controls for accidental human errors.

Effort ●○○ **Savings** ●●○

The above query was on the magnitude of GB, a mishap that can cost you a few cents, which is acceptable for most businesses. However, when you have dataset tables that are in the magnitude of TBs or PBs and are accessed by multiple individuals, unknowingly querying all columns could result in a substantial query cost.

In this case, use the maximum bytes billed setting to limit query cost. Going above the limit will cause the query to fail without incurring the cost of the query, as shown below.



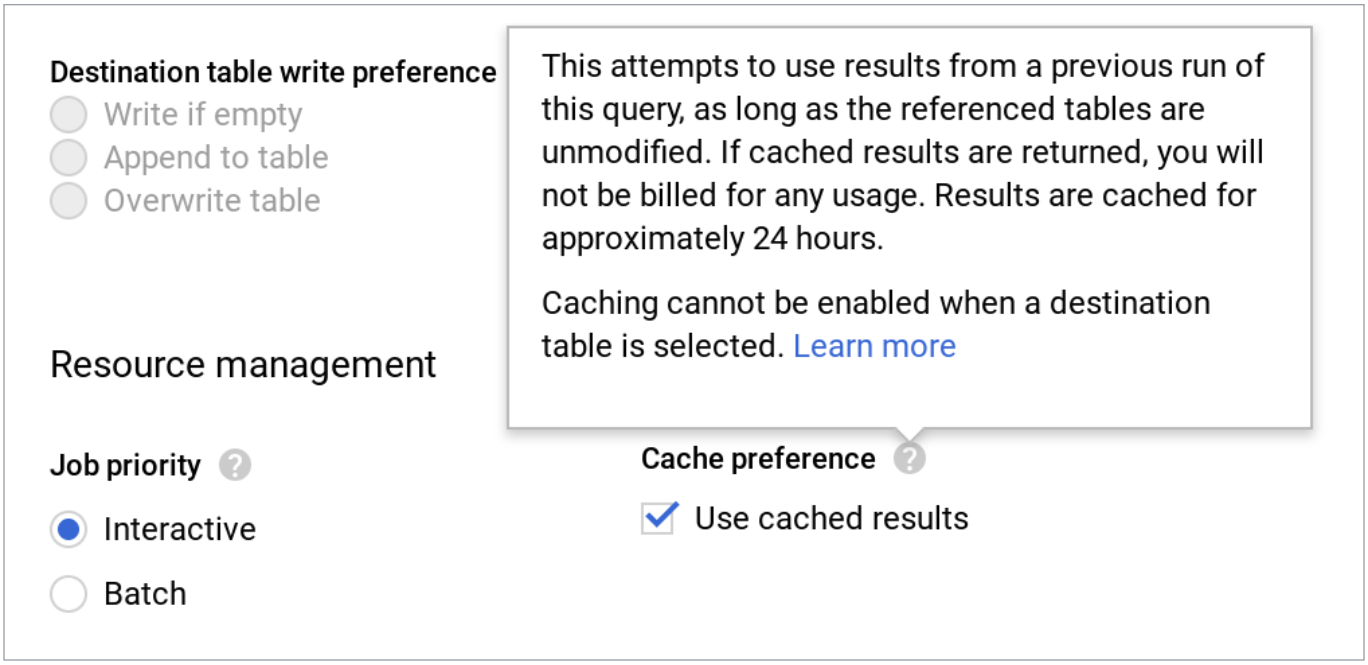
A customer once asked why custom control is so important. To put things into perspective, let's say you have 10 TB of data in a U.S. (multi-regional) location, for which you are charged about $200 per month for storage. If 10 users sweep all the data using [SELECT * ..] 10 times a month, your BigQuery bill is now about $5,000, because you are sweeping 1 PB of data per month. Applying thoughtful limits can help you prevent these types of accidental queries. Note that cancelling a running query may incur up to the full cost of the query as if it was allowed to complete.

Along with enabling cost control on a query level, you can apply similar logic to the user level and project level as well.

3. Use caching intelligently.

Effort ●○○ **Savings** ●○○

With few exceptions, caching can actually boost your query performance, and you won't be charged for the results retrieved from the cached tables. By default, cache preference is turned on. Check cache settings in your Google Cloud console by clicking More -> Query settings on your query editor, as shown here: Also, keep in mind that caching is per user, per project.



Caching can boost your query performance

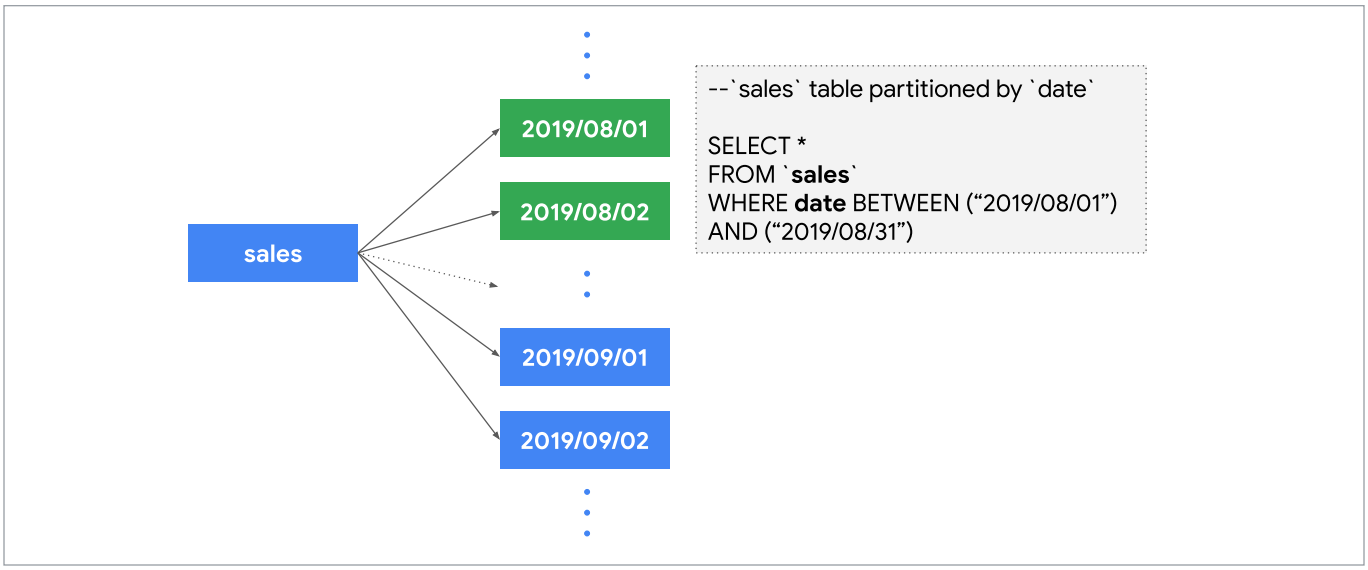
Let's take a real-world example, where you have a Data Studio dashboard backed by BigQuery and accessed by hundreds or even thousands of users. This will show right away that there is a need for intelligently caching your queries across multiple users.

To significantly increase the cache hit across multiple users, use a single service account to query BigQuery, or use community connectors, as shown in this Next '19 demo.

4. Partition your tables.

Effort ●○○ **Savings** ●●○

Partitioning your tables, whenever possible, can help reduce the cost of processing queries as well as improve performance. Today, you can partition a table based on ingestion time, date, or any timestamp column. Let's say you partition a sales table that contains data for the last 12 months. This results in smaller partitions containing data for each day, as shown below.



Partitioning your tables can reduce the cost of processing queries.

Now, when you query to analyze `sales` data for the month of August, you only pay for data processed in those 31 partitions, not the entire table.

One more benefit is that each partition is separately considered for long-term storage, as discussed earlier. Considering our above example, `sales` data is often loaded and modified for the last few months. So all the partitions that were not modified in the last 90 days are already saving you some storage costs. To really get the benefits of querying a partitioned table, you should filter the table using a partition column.

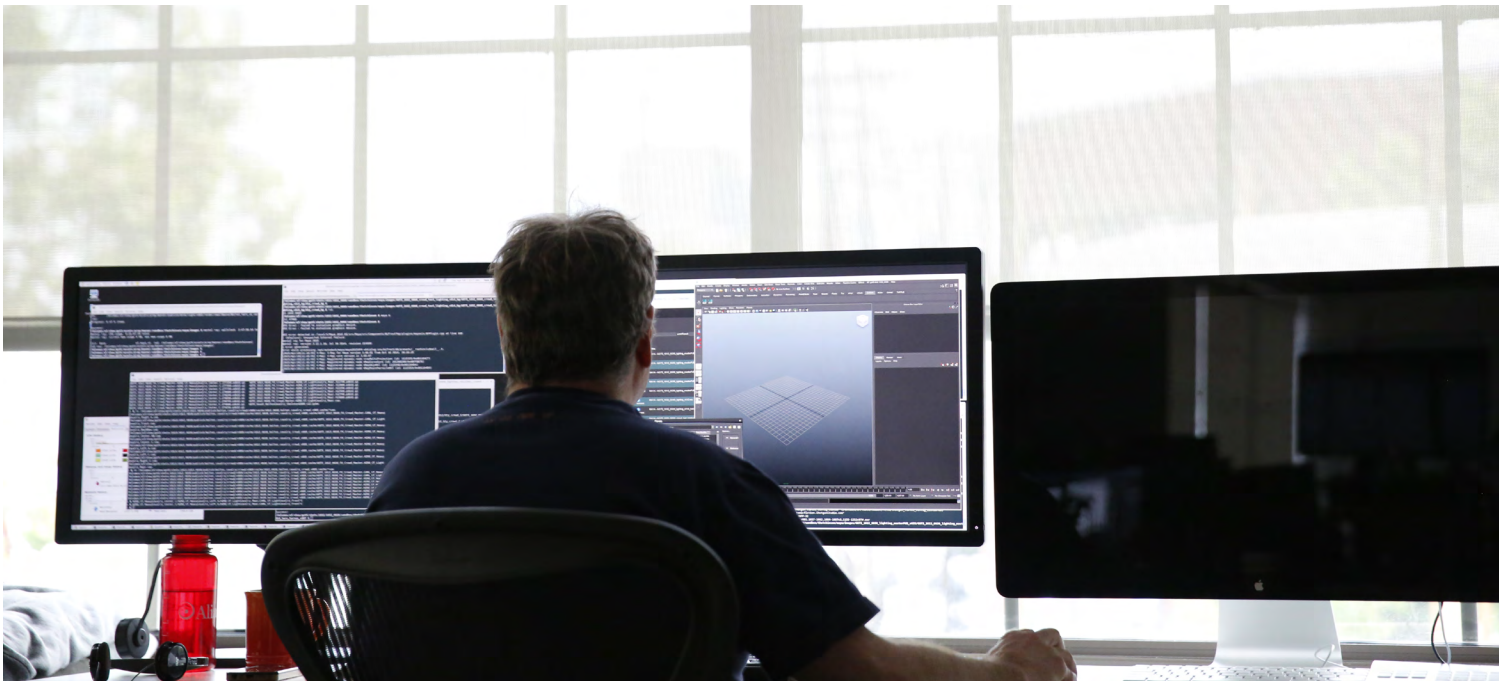
While creating or updating partitioned tables, you can enable Require partition filter, which will force users to include a WHERE clause that specifies the partition column, or else the query will result in error.

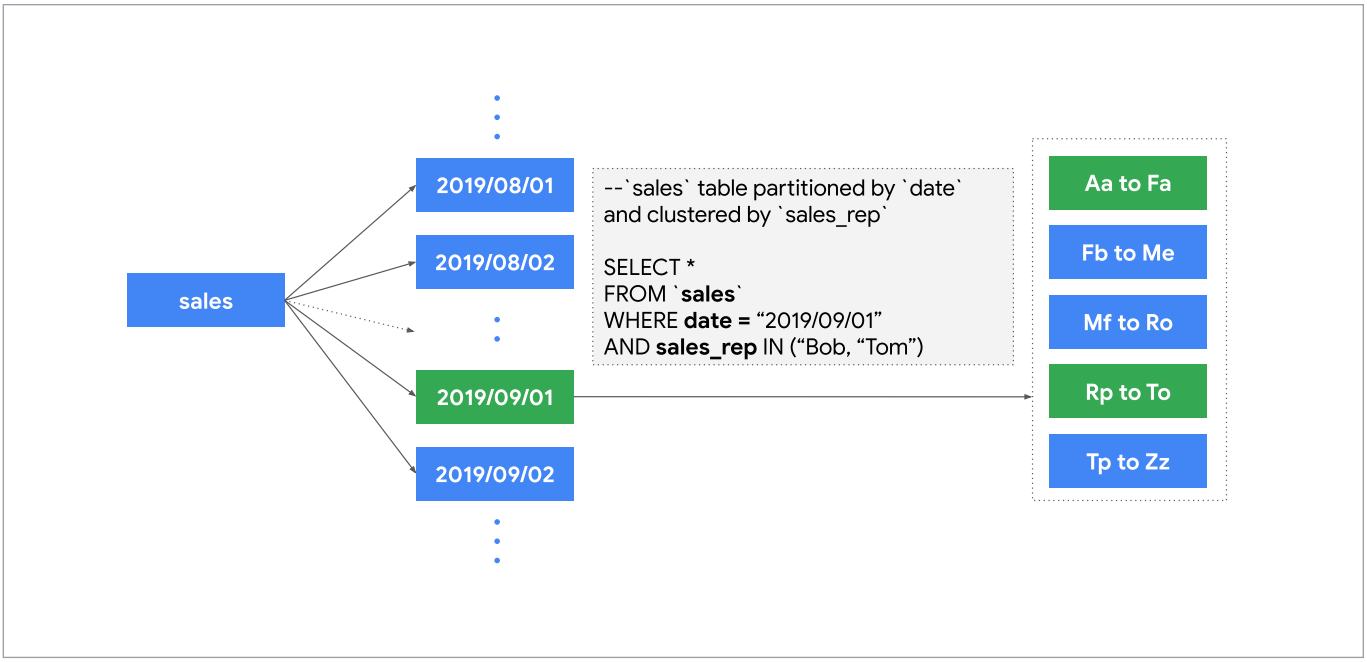
5. Further reduce sweeping your data using clustering.

Effort ●○○ **Savings** ●○○

After partitioning, you can now cluster your table, which organizes your data based on the content for up to four columns. BigQuery then sorts the data based on the order of columns specified and organizes them into a block. When you use query filters using these columns, BigQuery intelligently scans only the relevant blocks using a process referred to as block pruning.

For example, below, sales leadership needs a dashboard that displays relevant metrics for specific sales representatives. Enabling clustering on the sales_rep column is a good strategy, as it is going to be used often as a filter. As shown below, you can see that BigQuery only scans one partition (2019/09/01) and the two blocks where sales representatives Bob and Tom can be found. The rest of the blocks in that partition are pruned. This reduces the number of bytes processed and thus the associated querying cost.





Enabling clustering can help reduce costs.

Clustering is allowed only on partitioned data. You can always use partitioning based on ingestion data, or introduce a fake date or timestamp column to enable clustering on your table.

You can find much more here on clustering. And explore new Materialized Views for improved performance and efficiency, along with cost savings.

Cost optimization techniques in BigQuery: storage

Once data is loaded into BigQuery, charges are based on the amount of data stored in your tables per second. Here are a few tips to optimize your BigQuery storage costs.

1. Keep your data only as long as you need it.

Effort ●○○ **Savings** ●●○

By default, data stored in BigQuery's Capacitor columnar data format is already encrypted and compressed. Configure default table expiration on your dataset for temporary staging data that you don't need to preserve.

For instance, in this example, we only need to query the staging weather dataset until the downstream job cleans the data and pushes it to a production dataset. Here, we can set seven days for the default table expiration.

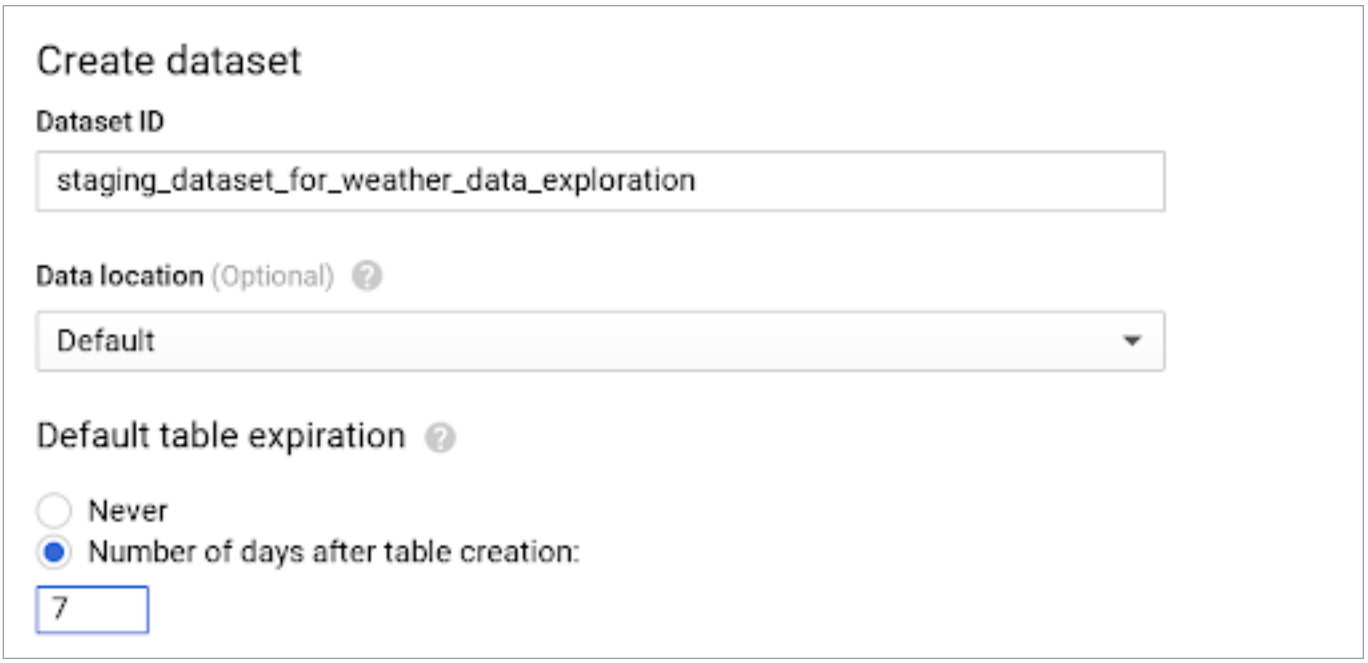


Table expiration dates can help save resources.

Note that if you're updating the default table expiration for a dataset, it will only apply to the new tables created. Use a DDL statement to alter your existing tables.

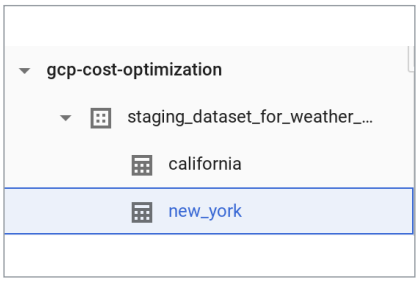
BigQuery also offers the flexibility to provide different table expiration dates within the same dataset. So this table called new_york in the same dataset needs data retained for longer.

As shown in the images next page, new_york will retain its data for six months, and because we haven't specified table expiration for california, its expiration will default to seven days.

Similar to dataset-level and table-level expiration, you can also set up expiration at the partition level. Check out our public documentation for default behaviors.

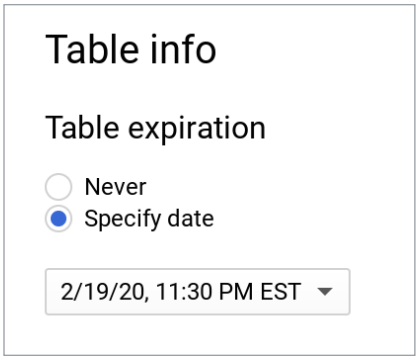
2. Be mindful of how you edit your data.

Effort ●○○ **Savings** ●●○



If your table or partition of a table has not been edited for 90 days, the price of the data stored in the table automatically drops by about 50%. There is no degradation of performance, durability, availability, or any other functionality when a table or partition is considered for long-term storage.

To get the most out of long-term storage, be mindful of any actions that edit your table data, such as streaming, copying, or loading data, including any DML or DDL actions. This will bring your data back to active storage and reset the 90-day timer. To avoid this, you can consider loading the new batch of data to a new table or a partition of a table if it makes sense for your use case.



Querying the table data, along with few other actions, does not reset the 90-day timer and the pricing continues to be considered as long-term storage.

In most cases, keeping the data in BigQuery is advantageous unless you are certain that the data in the table will be accessed at most once a year, like storing archives for legal or regulatory reasons. In that case, explore the option of exporting the table data into the coldline class of a Cloud Storage bucket for even better pricing than BigQuery's long-term storage.

3. Avoid duplicate copies of data.

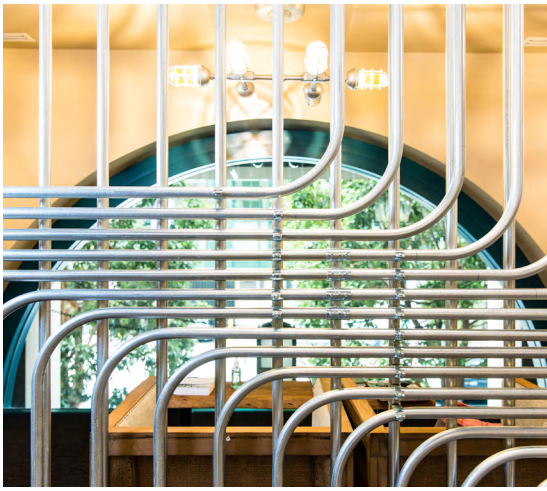
Effort ●○○ **Savings** ●○○

BigQuery uses a federated data access model that allows you to query data directly from external data sources like Cloud Bigtable, Cloud Storage, Google Drive, and Cloud SQL. This is useful for avoiding duplicate copies of data, thus reducing storage costs. It's also helpful for reading data in one pass from an external source or accessing a small amount of frequently changed data that doesn't need to be loaded in BigQuery every time it is changed.

Choose this technique for the use cases where it makes the most sense. Typically, queries that run on external sources don't perform as well compared to queries executed on the same data stored on BigQuery, since data stored on BigQuery is in a columnar format that yields much better performance.

4. See whether you're using the streaming insert to load your data.

Effort ●○○ **Savings** ●●○



You can load data into BigQuery in two ways: as a batch load job, or with real-time streaming, using streaming inserts. When optimizing your BigQuery costs, the first thing to do is check your bill and see if you are being charged for streaming inserts. And if you are, ask yourself, "Do I need data to be immediately available (seconds instead of hours) in BigQuery?" and "Am I using this data for any real-time use case once the data is available in BigQuery?" If the answer to either of these questions is no, then we recommend you to switch to batch loading data, which is free.

5. Understand BigQuery's backup and DR processes.

Effort ●○○ **Savings** ●○○

BigQuery maintains a seven-day history of changes to your table, which allows you to query a point-in-time snapshot of your data. This means you can revert back the data without restoring from recovery backups. If the table is deleted, its history is flushed after two days.

To find the number of rows from a snapshot of a table one hour ago, use the following query:

```
Select COUNT(*) FROM [Project _ ID:Dataset.Table@-3600000]
```

Find more examples in the documentation.

For business-critical data, follow the Disaster Recovery Scenarios for Data guide for a data backup, especially if you are using BigQuery in a regional location.

Celebrate your success

Done right, BigQuery can satisfy all your modern data warehousing needs at a very reasonable price. Once the cost optimization actions are implemented, you should see a visible drop in your BigQuery bill (unless you've followed best practices since day one). In either case, celebrate your success! You deserve it.

Cost optimization in action

Following these principles may save on cost and open up capacity in the short term—but the long-term ROI pays plenty of dividends. The customers we talk to have seen drastically reduced data costs and infrastructure savings, letting them add brand-new capabilities and explore innovative solutions like machine learning. They've also gained better performance and simplicity across their IT environments. It becomes much easier to scale as needed, too.

One study, for example, found that Google Cloud's Spanner database brings a total cost of ownership (TCO) that's 78% lower than on-premises databases. Optiva migrated its legacy Oracle databases to Google Cloud for cost efficiency and achieved better scale.

And customers across industries have been able to do much more with cloud, and help teams be more productive, compared to other cloud providers or legacy technology. That can lead to better customer experiences, new product innovation, and adding new business initiatives. Machine learning platform provider MD Insider often experienced

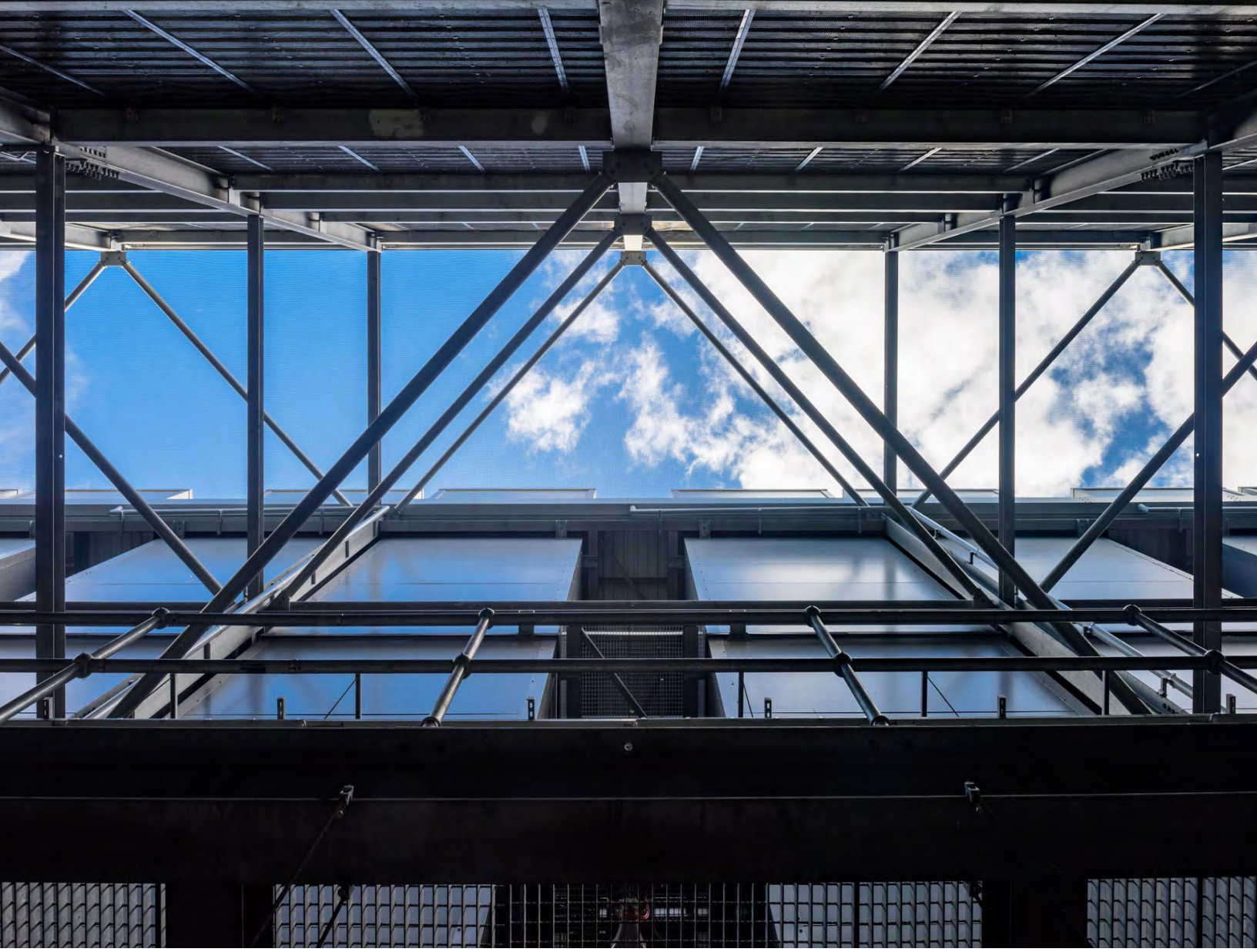
slow network performance, and failure rates and costs grew accordingly. Using Google Cloud for data services has led to a 5x performance improvement and accelerated time to market of up to 30%. The physician performance site's developers are now much more productive, and time previously spent managing infrastructure is now spent on product development.

Raycatch reduced infrastructure costs for its AI-led renewable energy tech by 80% using Compute Engine, Cloud Bigtable, and other technologies. The company has added flexibility and stability to its systems, and opened up capacity from its previous cloud, which means developers can work faster. Raycatch has been able to reduce work hours needed to oversee one analysis task by 60 times, so the IT team can do more valuable optimization work.

There are many cost management strategies and best practices that can help you get the most from the cloud. Use what works best for your team and organization. With a little bit of effort, the cloud offers efficiencies today and the potential for significant ROI into the future.

Learn more about cost optimization

Discover ways to reduce and optimize your IT spend for immediate and long-term growth. Talk to a Google Cloud partner to begin.



Google Cloud

cloud.google.com