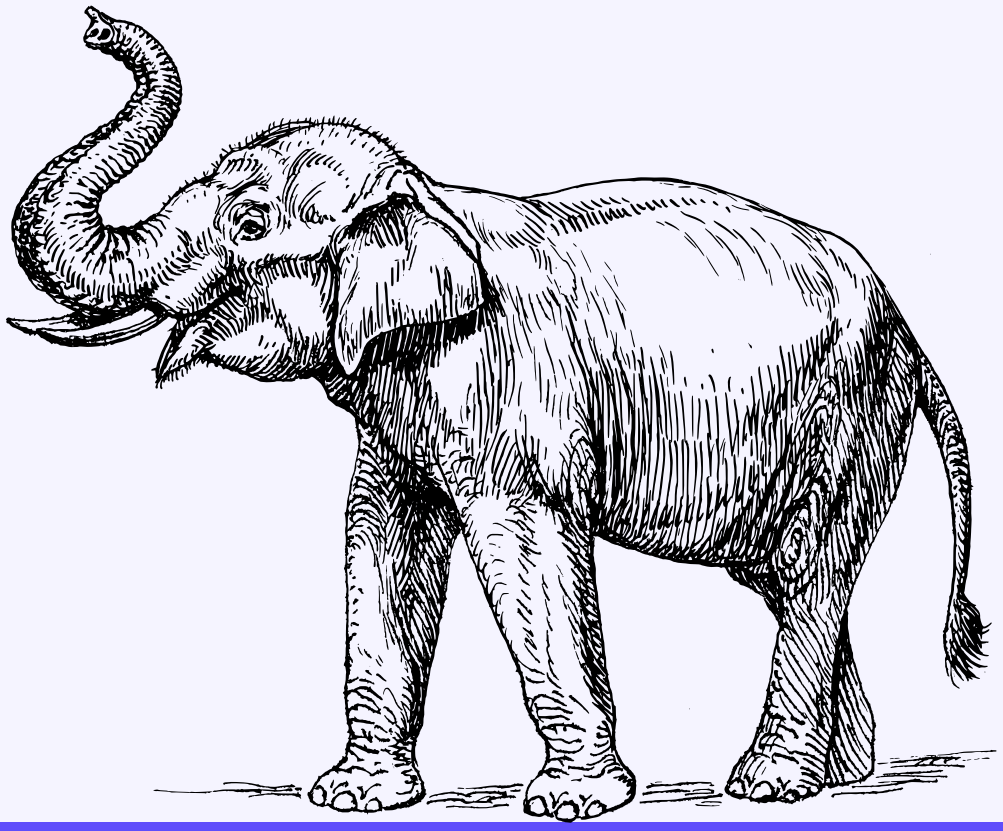
Addressing all the challenges
organizations face with AWS



Cloud Cost Optimization

the ultimate handbook for
engineers and architects

TENACITY

www.tenacity.ai

Table of Contents

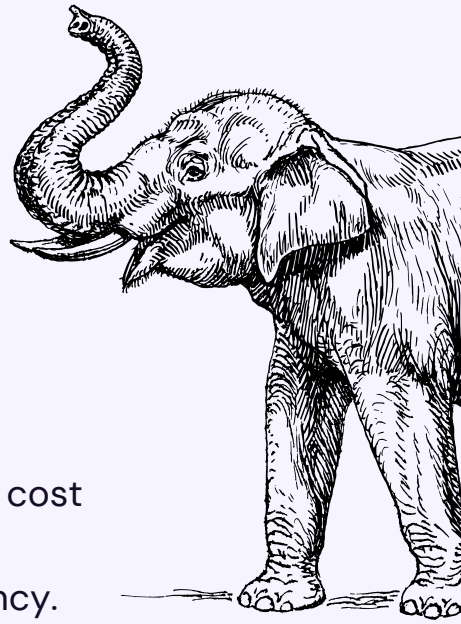
Intro

Cloud computing has become an essential part of modern business operations, providing organizations with the flexibility and scalability needed to meet the demands of a fast-paced digital economy. However, as organizations continue to adopt cloud services, many are finding that the costs associated with these services become increasingly inefficient and unpredictable especially as scale and complexity grow.

This handbook provides an in-depth look at the challenges organizations face when it comes to cloud cost optimization and offers practical solutions for reducing costs and maximizing the value of cloud investments.

This handbook will review the different strategies and tools that organizations can use to optimize their cloud costs. These include:

- Optimizing resource usage
- Right-sizing resources
- Automating cost management
- Utilizing reserved instances and savings plans
- Implementing cost allocation and chargeback mechanisms
- Managing multi-cloud environments



We will conclude by highlighting the benefits of cloud cost optimization, including improved cost visibility, better resource utilization, and increased operational efficiency.

Overview

To start, it is important to understand why cloud optimization is elusive - the primary driver being the complexity of cloud billing. In fact, a cloud bill can comprise of hundreds of millions or billions of rows of data

There are several reasons cloud billing is difficult for organizations to master:

- **Usage-based pricing:** Many cloud services are billed based on usage, which can make it difficult for organizations to predict and budget for costs. The usage can be based on factors such as data storage, data transfer, compute power, and number of requests, and prices can vary depending on the region, type of service, and provider.
- **Multi-cloud environments:** Many organizations use multiple cloud service providers, each with its own pricing structure and billing system. This can make it difficult to track and manage costs across different providers and services.
- **Dynamic scaling:** Cloud services can scale automatically based on usage, which can make it difficult to predict costs and stay within budget.
- **Lack of visibility:** The lack of visibility into cloud costs can make it difficult to understand where costs are coming from, and how to optimize spending.
- **Lack of standardization:** Cloud providers use different terminology, metrics and methods of billing, which can make it difficult to understand and compare costs.
- **Hidden costs:** Some costs such as data transfer, data egress, or data storage can be hidden and are not easily identifiable, making it hard to track and manage.

As environments scale, each of these contributes to hiding waste, orphaned and abandoned resources, under-utilized infrastructure, misconfiguration, excessive duplication, legacy projects, and more. When factoring all of these contributions in, it is commonly estimated that approximately 30% of cloud spend is wasted. In addition to these elements of technical debt, the nature of the cloud services themselves may inflict additional cost drivers.

In the next section we will start with understanding the cost drivers when managing cloud spending. From there we provide a simple framework for thinking about the problem of Cloud Cost Optimization and how to address it.

Cost Drivers

There are several key drivers that contribute to the costs associated with cloud computing:

- **Cloud service usage:** The amount of resources consumed, such as data storage, data transfer, and compute power, is a major driver of cloud costs.
- **Cloud service type:** Different cloud services have different pricing models, with some services being more expensive than others. For example, using a dedicated host will generally be more expensive than using a shared cloud service.
- **Cloud service provider:** Different cloud service providers have different pricing structures and cost drivers. Some providers may charge more for certain services or have higher costs associated with certain usage scenarios.
- **Data storage and data transfer:** Data storage and data transfer are major cost drivers in cloud computing. The more data that is stored and transferred, the higher the costs will be.
- **Cloud service region:** The costs for cloud services can vary depending on the region in which the service is provided.
- **Cloud service features:** Some features like load balancer, auto scaling, or databases are more costly than others
- **Cloud service scalability:** costs can increase as the usage increases, for example, as the number of users or the amount of data processed increases.
- **Cloud service security:** Security features such as encryption, firewalls, and intrusion detection systems can add to the overall cost of cloud services.

Cost Optimization Framework

Cloud optimization is the process of identifying and eliminating waste, inefficiencies, and unnecessary spending in order to minimize costs and maximize the value of cloud investments. It involves using strategies and tools to optimize resource usage, right-size resources, automate cost management, utilize reserved instances and savings plans, implement cost allocation and chargeback mechanisms and manage multi-cloud environments.

In this guide we will follow the principles of optimization as Identify, Fix & Save.

Identify is the first step in the optimization process and includes identifying waste, abandoned infrastructure, right-sizing, and upgrade opportunities.

Fix is the second step in the optimization process and automating the ongoing management of cloud costs which include reporting, forecasting, and continuous improvement.

Save is the final step which will take the now clean infrastructure and develop a commitment-based discount strategy to maximize discounts based on your organization's needs. Commit considers current usage, expected variance, growth or reduction, and the time needed to refactor critical applications for maximum savings.

Identify

Analyzing the Current Infrastructure

Before selecting the optimal Committed Use Discount strategy, it is important to understand current cloud spending and the optimization opportunities that exist in the infrastructure. The first and most common factor as cloud environments age is the presence of waste in the form of abandoned, orphaned, or legacy resources and workloads. The sections that follow will take your organization through the process of developing a clean infrastructure from which you can build a solid cloud savings strategy, develop your ongoing finops practices, and finally work on cloud cost optimization through a proactive approach in designing better, more cost-efficient cloud-native approaches by identifying areas for refactor and leverage the full benefits of cloud.

While the following can be done simultaneously, the sections are presented in the order in which they should be addressed. For example, it would be wasted effort to right-size an abandoned resource that is no longer needed. Identifying those abandoned resources first will minimize the work of right-sizing resources in the environment.

Finding Abandoned and Orphaned Resources

Identifying the cause of and developing a plan to address abandoned and orphaned resources early in the cloud journey is crucial to a strong overall cloud cost management strategy.

Abandoned resources are defined as resources that are no longer necessary. They may be testing or QA resources, left over from a retired project, anticipated capacity, legacy resources that have since been refactored, etc.

Orphaned resources are defined as resources that are or were a sub-component to another resource, but that is no longer used. A common example is an unattached storage volume that has been kept “just in case” and never subsequently cleaned up.

These resources should be identified and cleaned up before undertaking other opportunities to reduce cloud spend. Cleaning up these resources not only minimizes the remaining savings optimization work, but also minimizes potential attack surface from a cybersecurity perspective and eliminates an accumulating subset of noise when operating a cloud environment at scale.

On the next page we will walkthrough the recommendations for finding Abandoned Resources.

Recommendations for Finding Abandoned Resources

Unattached EBS Volumes	Storage volumes that are no longer attached to instances and could be deleted.
Unattached Elastic IPs	Elastic IP Addresses (EIP's) that are no longer attached to an instance or VPC. This is an example of a small hourly charge ($0.005) that can build up over time if an internal process is associating more than one EIP with an EC2 instance (for example).
Elastic Load Balancers	Load Balancers with no associated instances (none registered or no healthy instances) should be deleted.
S3 Multipart Uploads (MPU's)	Clean up leftover multi-part upload (MPU) data in S3. S3 allows for large files to be broken up into multiple parts. When an upload fails (e.g. connection loss), some clients don't do a good job at tidying up the useless parts of large files left behind as a result of the failure. This useless data consumes S3 storage.
Unused AMIs	Over time, unused AMIs can accumulate and should be eliminated to avoid increasing monthly costs to store them.
Unused Workspaces	AWS workspace instances that have 0 user connections registered are considered unused and should be removed.
Unused Accounts and Services	Unused accounts and services, whether from old projects, legacy environments, or past users should be removed to avoid unexpected charges. Even with just small charges, account sprawl can accumulate to generate large unnecessary expenses.

Detecting and Removing Idle Resources

Idle resources can be harder to identify because they are often purposed for current or scale-out workloads, but actually are unused. Find them by looking at the utilization metrics for your workloads and then target those that are 0 (or close to 0) over the past 14+ days. If close to 0, review the workload to understand if it should be right-sized (below), refactored, or simply eliminated.

For example, To detect whether an EC2 instance is being utilized, you can use Amazon CloudWatch to monitor various metrics. For example, you can use the CPUUtilization and the NetworkIn and NetworkOut metrics to see if an EC2 instance is actively used. You can also set up alarms to receive email notifications when one or more of these metrics exceed certain thresholds.

.

Fix

Set up Lifecycle Management

Data lifecycle management can be complicated even in small environments. The number of options and the implications, especially for S3, can create circumstances where costs expand exponentially, resulting in some of the largest overspending in public cloud. Let's break down the problem in two areas: snapshot and backup retention; and S3 lifecycle management.

Clean up and Set up Lifecycle Management	
Remove Old Snapshots	If not configured properly, Snapshots will continue to accumulate for a number of services and will remain after the resource is deleted. Delete unnecessary snapshots for any resources that are no longer in use as well as snapshots older than 30 days.
Configure Snapshot Lifecycle for EBS	Configure your EBS snapshot lifecycle to minimize costs by limiting to the necessary number of snapshots that must be retained, auto-archiving snapshots that require long term retention, and limiting the total retention time to what is necessary for recovering or regulatory obligations. Separate production lifecycle policies for critical data from non-critical storage lifecycle policies (e.g. dev, test, etc.)

Clean up and Set up Lifecycle Management (cont.)

Set up S3 Version expiration with Lifecycle Rules	When S3 versioning is enabled, AWS will retain all non-current versions of an object, forever, even when an object is deleted. A lifecycle rule must be created to clean up non-current versions after a set number of days or a number of version copies. Another lifecycle rule can be configured to remove objects that have been deleted.
Set up S3 Replication with S3 Version Lifecycle Rules	S3 Replication requires that S3 versioning be enabled on both the sources and destination. By default, S3 versioning will retain all non-current versions of an object, even if the object is deleted. As with the above, for S3 replication, S3 versioning lifecycle rules must be configured to remove non-current versions and deleted files.
Configure S3 object transition lifecycle	S3 objects and non-current versions can be configured to transition to lower-cost storage classes a number of days after creation.
Configure S3 object auto-tiering	Data with unknown or changing access patterns can be transitioned to S3 Intelligent tiering storage class. This will allow AWS to automatically transition objects to the appropriate storage class based on access patterns for that object.
Clean up Deregistered AMI snapshots	A deregistered AMI is an AMI that can no longer be used to create new EC2 instances, but is retained for current EC2 instances still running that AMI. Any AMI snapshots will incur storage charges and should be cleaned up.
Review AWS Backup configuration	AWS backups for non-continuous backups retain backups indefinitely by default. These costs can accumulate rapidly. Review your automated and on-demand backup rules to ensure that backups are set at an appropriate frequency and retention period for you requirements and have a transition policy to migrate aged backups to cold storage.

Manage Part-time Utilization

There are two primary use cases for part-time resources: schedule based and workers. Schedule-based part-time resources are infrastructure needed only during certain time windows (as the name implies). Examples include development environments that only need to run while the dev team is working (e.g. 9a-6p weekdays). Workers are part-time resources that are necessary only to do a specific task (e.g. scan and translate a document, produce a report, extract and load a piece of data, etc.). Workers can have intensive or non-intensive workloads.

Recommendations for Part-time Utilization	
Scheduling Instances	You do not incur any charges for stopped Amazon EC2 instances. Scheduled-based workloads can be shutdown or terminated while not used. Running a dev team schedule-based workload only 50 hours a week, as compared to all 168 hours, can save around 70%.
Workers / Flex-based	Flex-based part-time EC2 workloads can benefit from a number of different cost management and optimization strategies. Auto-scaling EC2 Spot Instances are a good way for fault tolerant workloads to run as much as 90% cheaper than persistent EC2 instances running the same workloads part-time. For smaller workloads, refactoring to leverage AWS Lambda is a good option with significant cost reduction as covered in Cost Optimization below.

Over-Provisioned vs. Under-Utilized resources

Over-provisioning is one of the most common cloud mistakes. This is a hold-over from the hardware era where technology sizing decisions were made based on prospective needs in the coming years, knowing that systems would initially be underutilized but that undersizing hardware could be catastrophic from a performance and capital perspective. The capabilities of the public cloud world have eliminated this issue. Workloads, specifically in terms of compute, storage, and databases should be sized for the needs of the workload today and adjusted as the demands of the workload change over time. Up-sizing workloads for even near-term speculative needs can become very expensive.

Under-utilization is the other side of the equation, where appropriate sized resources have experienced a change in demand and can be right-sized to meet the new usage pattern. This is common in the months and years post-transformation as new-to-cloud organizations become increasingly cloud-native organizations and learn to adopt and adapt to the power of the cloud. Continuing to monitor for falling demand and responding with ongoing right-sizing (and eventual elimination) of the resources will have a dramatic impact on overall savings.

Right-sizing workloads based on a 14 day window of use can save as much as 70% and should include EC2, RDS, and EBS for the largest impact on realized savings.

Upgrade Superseded Resource Generations

Once unnecessary resources have been eliminated, we are left with the required infrastructure and it is time to upgrade it to the latest generation. Upgrading the generation (if available) can both improve performance and often reduces the cost of the resources.

Recommendations for Part-time Utilization	
EC2 instance family upgrade	Upgrading from a legacy instance type to the latest generation can represent both a step up in performance as well as a significant cost reduction.
Update AMIs	In some cases, any AMI that you use may need to be updated to work with the newest generation of the instance type.
EBS volume upgrade	Upgrading to the latest generation EBS volume (e.g. from gp2 to gp3) can save 20% on storage costs.

Migrate Regions

Cost by region varies, with us-east-2, us-east-1, and us-west-2 generally being the cheapest regions. The region selected can impact certain cloud spend by up to 45% savings from most expensive to least. For example, AMD powered Amazon EC2 Instances can be up to 45% cheaper in the ap-south-1 region.

Save

Commitment-based Discount Strategy

There is an over-sized perception of risk in using commitment based discounts. This leads decision makers to opt for small discount, short term, low coverage options. But when taking these strategies apart, the core issue is less about the just-in-case need to reduce utilization of certain services and has much more to do with confusion around the benefits of a solid discount strategy even with periodic under-utilization.

In short, a plan that saves you 40%, but for which you end up only utilizing 85% at scale for a brief period of time is still significantly less expensive and less risk than paying On-Demand rates for the perceived benefit of flexibility during the entire period. We will cover how to optimize plans, think about the risk and mitigate issues with the right plan options.

No plan is perfect, but understanding what different strategies offer opens up significant opportunities to improve (reduce) cloud spending.

Committed Use vs. Committed Spend

AWS commitments come in two types: a commitment to use a certain amount of resources (committed use) and a commitment to spend at a certain dollar level per hour (committed spend). Reserved Instances are a commitment to use. Savings Plans are a commitment to spend. Both result in a discount on usage rates for eligible or matching AWS services. How that discount is applied is what differs.

For committed use (Reserved Instances), the discount is applied to a running instance that matches one of the Reserved Instances purchased. It's important to note that Reserved Instances are not physical instances, but rather a billing discount that is applied to the running On-Demand Instances in your account.

For committed spend (Savings Plans), the discount is applied to eligible usage supported by the Savings Plan. Savings Plans apply to your usage after any committed use (Reserved Instance) discounts are applied.

In both cases, any remaining usage beyond either commitment type is charged at the On-Demand rate.

Reserved Instances are ideal for steady and predictable usage. Whereas Savings Plans are Ideal for environments with less predicability and usage of other AWS compute services, namely AWS Fargate and AWS Lambda.

Committed Use Example

Committed use discounts (Reserved Instances) apply to any existing or new On-Demand Instances matching the selected attributes (i.e. as instance type, platform, tenancy, Region, or Availability Zone) are charged at the discounted Reserved Instance price. Reserved Instances act as an automatic discount on new or existing On-Demand Instances in your account.

In this example, let's assume you have acquired a reserved instance plan for two (2) m5.large EC2 instances with a 1 year commitment. At the time this was written, this RI commitment would give you a 38% discount. Instead of being charged the On-Demand rate for the use of any m5.large instances, you would pay the discounted rate per hour and it would be applied to any two m5.large instances in use.

If you own three Reserved Instances with the same instance attributes and region (or Availability Zone if applicable), the billing system checks each hour to see how many total instances you have running that match those parameters. If it is three or less, you will be charged the Reserved Instance rate for each matching instance running that hour. If more than three are running, you will be charged the On-Demand rate for the additional instances.

Committed Spend Example

AWS calculates the potential savings percentages of each combination of eligible usage for Savings Plans. This percentage compares the Savings Plans rates with your current On-Demand rates. Savings Plans are applied to the highest savings percentage usage type first, which is the r5.4xlarge Linux Instance example in the table below. Savings Plans continue to apply until there are no more remaining usages, or your commitment is exhausted. Any remaining usage is charged at the On-Demand rates. Note: EC2 Instance Savings Plans are applied before Compute Savings Plans.

Eligible Savings Plan Usage type	On-Demand rate	Compute Savings Plans rate	EC2 Instance Savings Plans rate
r5.4xlarge Linux	$1.00	$0.70 (30%)	$0.60 (40%)
Fargate vCPU	$0.04	$0.03 (25%)	N/A
Fargate GB	$0.004	$0.003 (25%)	N/A
m5.24xlarge Windows	$10.00	$8.20 (18%)	$7.80 (22%)
Lambda duration (per GB/sec)	$0.000015	$0.00001275 (15%)	N/A

Plan Benefits and Limitations

Plan Type	Benefits	Limitations
Standard Reserved Instances	• Highest discount levels • Buy/Sell on RI Marketplace • Zonal RI can reserve capacity • Size flexible	• Limited to instance family • Limited to region (or zone) • Limited to platform • Limited to tenancy
Convertible Reserved Instances	• Convertible to equivalent cost or more expensive plans • Convert to latest generation	• Limited to instance family • Limited to region (or zone) • Limited to platform • Limited to tenancy • Lower discount levels
EC2 Instance Savings Plans	• Highest discount levels • Size flexible • Platform not required	• Limited to EC2 • Not sellable • Not convertible
Compute Savings Plans	• Covers EC2, Fargate, Lambda • EC2 Size and Family flexible • Platform not required	• Not sellable • Lower discount levels

Reserved Instances Benefits

AWS Reserved Instances have several upsides, including:

- **Cost savings:** Reserved Instances provide a significant discount (up to 75%) compared to On-Demand pricing, allowing customers to save money across various AWS services, including Amazon EC2 and Amazon RDS.
- **Predictable costs:** By committing to a certain amount of usage for a period of time, customers can better predict and manage their AWS costs.
- **Flexibility:** Reserved Instances can be exchanged or returned, allowing customers to easily adjust their usage as needed.
- **Capacity reservation:** Reserved Instances can be used to reserve capacity in a specific availability zone, ensuring that instances are available when needed.
- **Better performance:** Reserved Instances can be used to launch instances in a specific instance type and availability zone, which can improve performance and reduce latencies.
- **Multiple usage models:** Reserved Instances can be used for various usage models such as steady-state, predictable traffic, and scalable burst workloads.
- **Payment options :** AWS provides different payment options like All Upfront, Partial Upfront and No Upfront for Reserved Instances.

Reserved Instances Limitations

AWS Reserved Instances have have some limitations to consider, including:

- **Use it or Lose it:** Once you've committed to a Reserved Instance, you can't cancel It without a penalty. In addition, you are agreeing to pay for It whether you use It or not.
- **Limited flexibility:** Reserved Instances are tied to a specific instance type and availability zone, so if your needs change, you may not be able to use the reserved instances.
- **Limited ability to scale:** Reserved Instances are limited to the number you purchase, so you may not be able to scale up or down as needed.
- **Limited ability to sell:** If you no longer need the reserved instances, you may not be able to sell them to recoup some of your costs.
- **Requires active management:** To take full advantage of Reserved Instances, you need to actively manage your instances and ensure that they are running on the reserved instances.
- **Not all instance families are available:** Not all families of instances are available for reservations, like GPU instances.

Savings Plans Benefits

AWS Savings Plans have several upsides, including:

- **Cost savings:** Savings Plans provide a significant discount (up to 40%) on usage, allowing customers to save money on their Amazon EC2, AWS Fargate, and AWS Lambda usage.
- **Flexibility:** You purchase Savings Plans on a dollar-per-hour basis, rather than a per-instance basis.
- **Predictable costs:** By committing to a certain amount of usage for a period of time, customers can better predict and manage their AWS costs.
- **Less Ongoing Management:** Savings Plans automatically apply to usage within the scope of the plan, without the need to manually modify instance reservations.
- **Multiple payment options:** Savings plans provide different payment options like All Upfront, Partial Upfront and No Upfront.
- **Combination of usage:** Savings plans combine usage across multiple accounts and regions, which can be useful for organizations with multiple teams or projects.
- **Maximized Value:** EC2 Instance Savings Plans apply to all instance types and sizes within a particular instance family and region, maximizing the value of the plan you purchase.

Savings Plans Limitations

AWS Savings Plans have have some limitations to consider, including:

- **Limited flexibility:** Savings Plans can be tied to a specific instance type and availability zone, similar to Reserved Instances, so if your needs change, you may not be able to use the savings plan.
- **Limited ability to scale:** Savings Plans are limited to the number you purchase, so you may not be able to scale up or down as needed.
- **Limited ability to change the term:** Once you've committed to a term, you may not be able to change it without penalty.
- **Not available for all services:** Savings plans are currently only available for Amazon EC2, AWS Fargate, and AWS Lambda.
- **May not be cost-effective:** Savings plans may not always be the most cost-effective option for all types of usage scenarios, especially if your usage patterns are unpredictable.

The RI marketplace

The AWS Reserved Instance Marketplace is a service provided by AWS that allows customers to buy and sell Reserved Instances.

The marketplace allows customers who no longer need their Reserved Instances to sell them to other customers who are looking for additional capacity. This feature allows customers to optimize their usage and maximize their cost savings.

Customers can buy and sell either Standard or Convertible Reserved Instances through the marketplace. Standard Reserved Instances are specific to a particular instance type, availability zone, and term, while Convertible Reserved Instances can be exchanged for instances of different sizes, families, or tenancies.

The AWS RI Marketplace offers a number of benefits, including:

- **Increased flexibility:** Customers can buy or sell Reserved Instances as their needs change, allowing them to optimize their usage and minimize waste.
- **Better cost management:** Customers can take advantage of the cost savings provided by Reserved Instances, even if they don't need them for the entire term.
- **Increased availability:** Customers can purchase Reserved Instances that are available in specific availability zones, ensuring that they have the capacity they need when they need it.
- **Easy to use:** The marketplace is integrated into the AWS Management Console, making it easy for customers to buy and sell Reserved Instances.

Enterprise Discount Plans

When AWS cloud spend exceeds $1,000,000 annually, customers may be eligible to take advantage of enterprise discount programs. These plans are a 3 year commitment to spend over $3,000,000 and give a fixed discount on all monthly cloud spend. This commitment cannot be sold or terminated.

It is best to take advantage of this discount program when spending is well above $1,000,000 annually to avoid being locked into a commitment for lower than the spending commitment.

Building a commitment-based discount strategy hinges on a few of factors:

- How much flexibility you anticipate needing.
- How much risk you are willing to assume.
- How much capital you are willing to deploy up front.
- How frequently you are willing to optimize the program.

Managing Risk for Cost Efficiency

Finally we will conclude with several considerations when optimizing cloud spending for cost efficiency, the first of which is to determine if the infrastructure is highly dynamic or mostly consistent. Dynamic environments will benefit from more flexible or sellable commitment based discount plans while consistent environments will be able to take advantage of the highest discounts while maintaining a high plan utilization.

For dynamic environments, use a combination of Compute Savings Plans for the baseline spending and sellable Standard Reserved Instances for dynamic needs of the environment. This will allow an organization to remove Reserved Instances obligations even if utilization is still high in favor of better Compute Savings Plan utilization. This strategy has the advantage of covering additional compute services beyond Amazon EC2.

For static or more consistent spending, Reserved Instances use can be expanded to achieve deeper discounts with a low commit Compute Savings Plan engineered to cover AWS Fargate and AWS Lambda usage, where applicable.

Managing Risk for Budget Stability

When optimizing cloud spending for stability and minimal variance, the focus will be on coverage instead of efficiency and plan utilization. Meaning, that commitment based discounts should be selected based on total coverage of the environment to fix the cost even if plan utilization may fluctuate.

When managing for predictable outcomes, the best commitment based discount vehicle is a Compute Savings Plan. This will maximize coverage and allow your organization to manage utilization through carefully increasing the Savings Plan to smooth any predicted variance as cloud budgets expand through the year.

Conclusion

Cloud computing has become essential for modern businesses, but as organizations adopt more cloud services, they often find that the costs associated with these services become increasingly inefficient and unpredictable.

There are several key drivers that contribute to the costs associated with cloud computing, including usage-based pricing, multi-cloud environments, dynamic scaling, lack of visibility into costs ,and lack of standardization among providers.

Managing all of these is complex and time consuming but can be done with the right attention, skill and management.

NEED A CUSTOMIZED PLAN FOR YOUR BUSINESS NEEDS?

SIGN UP FOR A FREE CLOUD SAVINGS ASSESSMENT →