Cost optimization e-book

01/2018 | Amazon Web Services

Table of contents

Introduction

Migrating to the cloud enhances customers' ability to scale and flex to the demands of their business workloads. Historically, computing costs were tied to a quarterly or yearly hardware procurement investment, but with cloud technology, customers now have the flexibility to initialize resources and services at any time—and they pay only for what they use. This has led to a pivot in the way that costs are understood, managed, and optimized.

In the past, hardware costs were large, upfront expenses, which led to predictable resource procurement and cost patterns. However, this often resulted in waste, since companies had to purchase enough servers to support their most highly trafficked day. Unfortunately, many of these servers would be idle for much of the year. Because the cloud lets companies scale on demand, they pay only for the resources they use, which minimizes waste but can result in variable cost patterns.

The ability to scale up and down on demand has allowed resource procurement to transition from being solely owned by the Finance team to being the responsibility of stakeholders across IT, Engineering, Finance, and beyond. This democratization of resource procurement has initiated an ever-growing group of cost-conscious stakeholders who are now responsible for understanding, managing, and, ultimately, optimizing costs.

This e-book provides an overview of the most important tenets of cost optimization, including:

- Foundations of cost optimization
- Right sizing
- Reserved Instances
- Elasticity
- Spot Instances
- Storage optimization
- Cost management tools
- Creating a lean cost culture

Using these processes, tools, and approaches, customers can achieve cost optimization that enables them to save money they can instead invest in greater innovation.

1. Foundations of Cost Optimization

CORE PILLARS OF OPTIMIZATION

- **Right sizing:** Provisioning the right services for given workloads.
- **Elasticity:** Growing and shrinking consumption based on volume- and time-based needs.
- **Pricing models:** Using On Demand, Reserved, and Spot Instances effectively.
- **Storage optimization:** Choosing the right storage tier for the workload.

BEST PRACTICES FOR OPTIMIZATION

- **Define and enforce cost allocation tagging.** Tagging enables the assignment of custom metadata to instances, images, and other resources. For example, resources can be categorized by owner, purpose, or environment, helping to organize them and assign cost accountability. Tagging taxonomy and enforcement should be determined as early as possible in the adoption for cloud services.
- **Use effective account structures.** For many organizations, a consolidated billing strategy where all AWS accounts are paid through one master account facilitates simplified payments, maximizes volume discounts, and makes it possible to share Reserved Instance benefits across linked accounts. AWS Organizations enables the creation of groups of AWS accounts, with central policy management and consolidated billing. For more information on effectively structuring accounts, see the AWS Multiple Account Billing Strategy article.
- **Define and use metrics.** Set targets and review progress against them at a set cadence.
- **Enable teams to architect for cost.** Employ training, visualization of progress goals, and balance of incentives.
- **Assign optimization responsibility to cloud center of excellence (CCoE).** A CCoE is charged with overseeing the quality and cost-effectiveness of cloud transformation efforts. It can start small and evolve with the organization’s needs. Having one locus of responsibility for cloud excellence can drive greater cost and value optimization.

Resources

- AWS Trusted Advisor
- Consolidated Billing
- AWS Organizations

2. Right Sizing

Right sizing is the process of matching instance types and sizes to performance and capacity requirements at the lowest possible cost. Right sizing prior to migration can significantly reduce infrastructure costs.

TOOLS FOR RIGHT SIZING

The first step in right sizing is to monitor and analyze usage metrics including vCPU utilization, memory utilization, network utilization, and ephemeral disk use.

- **AWS Cost Explorer:** Reflects the cost and usage of Amazon Elastic Compute Cloud (Amazon EC2) instances over the most recent 13 months.
- **AWS Trusted Advisor:** Provides real-time insight into service usage.
- **EC2 Right Sizing:** Takes two weeks of Amazon EC2 utilization data and provides detailed right-sizing. Recommendations using an automated reference deployment.

CHARACTERISTICS OF INSTANCE TYPES

Customers should use the following principles to correctly analyze specific instance types:

- **Compute-optimized instances:** Focus on recent data and instances that have run for at least half the period being considered. Ignore burstable instance families (T2), which are designed to run at low CPU utilization.
- **Storage-optimized instances (I2 and D2 instance types):** Start with the most commonly used instance type. Identify peaks in I/O and CPU utilization and right size to match.
- **Amazon RDS database instances:** Focus on average and maximum CPU utilization, minimum RAM, and average bytes read/written per second.

TYPICAL USAGE PATTERNS

Usage patterns can help determine which instance size to choose. For example:

- **Steady state:** The load remains constant over time, making forecasting simple. Consider using Reserved Instances.
- **Variable, but predictable:** The load changes on a predictable schedule. Consider using Auto Scaling.
- **Dev/test/production:** Development, testing, and production environments can usually be turned off outside of work hours.
- **Temporary:** Temporary workloads that have flexible start times and can be interrupted are good candidates for Spot Instances.

MIGRATION COMPATIBILITY

When migrating to a different instance family, the current instance type and the new instance type must be compatible:

- **Virtualization type:** The instances must have the same Linux AMI virtualization type (PV AMI versus HVM) and platform (EC2-Classic versus EC2-VPC).
- **Network:** Instances unsupported in EC2-Classic must be launched in a virtual private cloud (VPC).
- **Platform:** If the current instance type supports 32-bit AMIs, make sure to select a new instance type that also supports 32-bit AMIs (not all EC2 instance types do).

Resources

- Linux AMI Virtualization Types
- Instance Types Available Only in a VPC

3. Reserved Instances

Reserved Instances (RIs) allow customers to commit to usage parameters and a one-year or three-year duration at the time of purchase to unlock an hourly rate that is up to 75 percent lower than On Demand pricing. Although some provide a capacity reservation, RIs are not physical instances. They are billing discounts applied when customers use an On Demand Instance that matches the attributes specified in a contract.

Except for Amazon DynamoDB reservations, which are billed based on throughput, reservations are billed for every clock-hour during the term that customers select, regardless of whether an instance is running or not.

By standardizing instance types, customers can ensure that deployments match the characteristics of RIs to maximize discounts. Additionally, the pricing benefits of RIs are shared when the purchasing account is billed under a consolidated billing payer account.

RI PAYMENT OPTIONS

There are three payment options for RIs.

- **No upfront:** No upfront payment is required, and RIs are billed monthly. This requires a good payment history with AWS.
- **Partial upfront:** A portion of the cost is paid upfront, and the remaining hours in the term are billed at a discounted hourly rate, regardless of whether the RI is being used.
- **All upfront:** Full payment is made at the start of the term, with no other costs or additional hourly charges incurred for the remainder of the term, regardless of hours used.

STANDARD VS. CONVERTIBLE RIs

When customers purchase an Amazon EC2 RI, customers can choose between a Standard or Convertible class.

- **Standard:** Allows customers to modify Availability Zone, scope, network platform, and instance size for Linux RIs.

- **Convertible:** Allows customers to exchange the Convertible RI for another one with new attributes, including instance family, instance type, platform, scope, and tenancy.

REGIONAL VS. ZONAL RIs

Standard and Convertible Amazon EC2 RIs can be purchased to apply to instances in a specific Availability Zone (Zonal RI) or to instances in a region (Regional RI). Zonal RIs can provide a capacity reservation.

AWS services with reservation models

- Amazon EC2 RIs
- Amazon RDS RIs
- Amazon DynamoDB Reserved Capacity
- Amazon ElastiCache Reserved Nodes
- Amazon Redshift Reserved Nodes

Resources

- Modifying Reserved Instances
- Exchanging Convertible RIs
- Understanding Consolidated Bills
- Turning Off Reserved Instance Sharing

4. Elasticity

Elasticity means matching resources to needs. There are two basic types of elasticity:

- **Time-based.** Turns off resources when they are not being used.
- **Volume-based.** Matches scale to the intensity of demand.

Automating elasticity enables customers to gain the greatest benefits, especially as environments scale up.

AUTOMATING TIME-BASED ELASTICITY

Customers can use the following tools to automatically start, stop, or terminate unused or underused instances:

- **Amazon EC2 Scheduler:** Creates automatic start and stop schedules for Amazon EC2 instances in all AWS Regions of an account.
- **Amazon EC2 API tools:** Terminates instances programmatically using the TerminateInstances and StopInstances actions.
- **AWS Lambda:** Starts and stops instances when triggered by Amazon CloudWatch Events, such as a specific time or utilization threshold.
- **AWS DataPipeline:** Stops and starts Amazon EC2 instances by running AWS Command Line Interface (CLI) commands on a set schedule.
- **Amazon CloudWatch:** Stops or terminates unused or underutilized Amazon EC2 instances.

AUTOMATING VOLUME-BASED ELASTICITY

These tools enable customers to match resources to demand automatically:

- **Auto Scaling:** Automatically increases the number of Amazon EC2 instances during demand spikes to maintain performance and decrease capacity during lulls.
- **Application Auto Scaling:** Automatically scales resources for other AWS services, including:

- **Amazon ECS**: Can scale services automatically in response to CloudWatch alarms.
- **Amazon EC2 Spot Fleets**: Launches or terminates instances according to scaling policies.
- **Amazon EMR clusters**: Scales out and scales in core and task nodes in a cluster.
- **AppStream 2.0 fleets**: Adjusts the size of a fleet automatically based on utilization metrics, and optimizes the number of running instances to match demand.
- **Amazon DynamoDB:** Dynamically adjusts provisioned throughput capacity in response to actual traffic patterns.

Resources

- Amazon CloudWatch
- TerminateInstances and StopInstances
- Automating elasticity with AWS Lambda
- Automating elasticity with AWS DataPipeline
- Application Auto Scaling
- Spot Fleet auto scaling
- Amazon EMR auto scaling
- AppStream 2.0 fleets auto scaling
- Amazon DynamoDB auto scaling

5. Spot Instances

Spot Instances allow customers to use spare Amazon EC2 computing capacity at discounts of up to 90 percent compared to on-demand pricing. If the Spot price exceeds a customer's maximum stated price or if capacity becomes unavailable, a Spot Instance is terminated or stopped. The Spot price is determined by long-term trends in supply and demand for EC2 spare capacity. AWS publishes the current Spot price and historical prices for Spot Instances through the describe-spot-price-history command as well as the AWS Management Console. Spot works well with workloads that can be interrupted and replaced by an On Demand Instance without the need to back up and restore data, or that constantly save data to persistent storage.

REQUESTING SPOT INSTANCES

Spot Instances can be requested using:

- AWS Management Console
- Amazon EC2 RunInstances API
- Amazon Elastic MapReduce (Amazon EMR)
- AWS Data Pipeline
- AWS CloudFormation
- Amazon Elastic Container Service (Amazon ECS)

TOOLS FOR MANAGING INSTANCE TERMINATION

Customers can use the following approaches to manage potential disruptions due to Spot Instance termination:

- **Termination Notices:** Issued 2 minutes prior to interruption.
- **Persistent Requests:** Set a Spot request to remain open so that a new instance will be launched in its place when the instance is interrupted. Choose **stop** instead of **terminate** for a persistent Spot request to persist data to EBS root device and attached EBS volumes.
- **Block Durations:** Specify a 1- to 6-hour duration requirement.

ADDITIONAL SPOT FEATURES AND OPTIONS

Spot Instances provide additional flexibility through:

- **Launch Groups:** Deploy multiple Spot Instances at once.
- **Spot Fleets:** Automatically request Spot Instances with the lowest price per unit of capacity.

Resources

- Spot Instances overview
- Describe-spot-price-history command
- Launching Spot Instances in Your Auto Scaling Group
- Amazon EC2 API Reference
- Managing Spot interruptions
- Termination notices
- Spot Instance Requests
- Launch Groups
- Spot Fleets

6. Storage Optimization

The key to keeping storage costs low without sacrificing required functionality is to maximize the use of appropriate Amazon Simple Storage Service (Amazon S3) storage tiers when possible, including low-cost Amazon Glacier for archiving, and to use more expensive Amazon Elastic Block Storage (Amazon EBS) volumes with provisioned I/O only when necessary. Organizations should optimize storage periodically. When evaluating storage requirements, customers should segment data by how available and durable it needs to be, the size of data sets, throughput and IOPS thresholds, and regulatory requirements.

OPTIMIZING AMAZON S3 STORAGE

Amazon S3 lets customers configure lifecycle policies to automatically migrate data objects to cheaper tiers of S3 storage as objects are accessed less frequently or delete objects after an expiration date. Amazon S3 object tags categorize data for lifecycle policy application.

OPTIMIZING AMAZON EBS STORAGE

When Amazon EC2 instances are stopped or terminated, attached Amazon EBS volumes are not automatically deleted and will continue to accrue charges. Customers should identify unattached and under-or over-utilized Amazon EBS volumes using tools such as Amazon CloudWatch, AWS Trusted Advisor, or third-party solutions such as Cloudability. Customers can automate the process of deleting unattached volumes by using AWS Lambda functions with Amazon CloudWatch.

- **Resizing or changing volume type:** If customers have a current-generation Amazon EBS volume attached to a current-generation Amazon EC2 instance type, they can use the Elastic Volumes feature to change the size or volume type, or in the case of an SSD (io1) volume, they can adjust IOPS performance without detaching the volume.
- **Deleting stale Amazon EBS snapshots:** If customers have a backup policy that takes EBS volume snapshots daily or weekly, customers will quickly accumulate snapshots. Check for "stale" snapshots—that is, ones that are more than 30 days old—and delete them to reduce storage costs. Deleting a snapshot has no effect on the volume. Customers can use AWS Console or AWS Command Line Interface (CLI) for this purpose, or third-party tools such as Skeddly.

Resources

- Amazon S3 Analytics
- Monitoring the Status of Your Volumes
- Detaching an Amazon EBS Volume from an Instance
- Amazon CloudWatch Events for Amazon EBS
- Deleting an Amazon EBS Snapshot

7. Cost Management Tools

AWS provides a wide range of tools for analyzing, tracking, and managing costs. This section covers a selection of the most generally useful options. Customers can also use their own tools or ones provided by third-party developers.

MONTHLY AWS BILL

Accessible via the AWS Billing and Cost Management console, the AWS bill breaks down costs by service, region, and linked account. It can be further explored via the Bills page.

AWS COST EXPLORER

This tool enables customers to visualize, understand, and manage AWS costs and usage over time using an intuitive interface. Users can filter and group information and quickly create custom reports that include charts and tabular data. AWS Cost Explorer can display up to 12 months of historical data, data for the current month, and the costs that are forecast for the next three months. Default reports include:

- Monthly Costs by AWS Service
- Amazon EC2 Monthly Cost and Usage
- Monthly Costs by Linked Account
- Monthly Running Costs
- Reserved Instance (RI) Reports

AWS COST & USAGE REPORT

This tool tracks AWS usage and estimates by the hour or day. The Cost & Usage Report is updated at least once a day until the end of the billing period. The report provides the most comprehensive insight into costs and usage, and it is the source of truth for the billing pipeline. It can also be used to develop advanced custom metrics. Delivered automatically to an Amazon S3 bucket that the customer specifies, it can be downloaded directly from there (standard S3 storage rates apply).

AWS BUDGETS

Enables users to set custom cost and usage budgets and to receive alerts when approaching or exceeding budgeted amounts. Budget can be created from the AWS Budgets Dashboard or programmatically via the AWS Budgets API. Budgets can be configured to track cost or usage by month, quarter, or year.

Resources

- Getting started with AWS Cost Management tools
- AWS Billing and Cost Management console
- AWS Cost Explorer
- AWS Cost & Usage Report
- AWS Budgets

8. Creating a Lean Cost Culture

CULTURE IS CRITICAL TO COST OPTIMIZATION

Because cloud resources are easier to deploy and they incur usage-based costs, organizations must rely on good governance and user behavior to manage costs. Cost optimization, which should be considered from the outset as part of system design and architecture, is one of the pillars of the Well-Architected Framework.

CHARACTERISTICS OF A LEAN COST CULTURE

- Employees view change as normal and welcome responsibility in the interest of following best practices and adapting to new technology.
- Costs are clearly allocated using linked accounts and tags.
- Key performance indicators affected by cloud adoption are known, shared, and measured, with a regular review cadence.
- There is a mindset shift from cloud cost to cloud ROI.
- Cost optimization is considered in planning, design, and development.
- Tradeoffs between speed-to-market and upfront cost optimization are consciously chosen.

COST OPTIMIZATION IS EVERYONE'S RESPONSIBILITY

All teams can help manage cloud costs, and cost optimization is everyone's responsibility. Few of these teams have previously been given responsibility for understanding, let alone managing, IT costs. Stakeholders need training, policies, and tools to do it effectively. For example:

- Engineering needs to know the cost of deploying resources and how to architect for cost optimization.
- Finance needs cost data for accounting, reporting, and decision making.
- Operations makes large-scale decisions that affect IT costs.
- Business decision makers must track costs against budgets and understand return on investment.
- Executives need to understand the impact of cloud spending to help with divestitures, acquisitions, and organizational strategy.

Lean cost culture resources

- AWS Organizations
- AWS Cost Explorer
- Alerts and Notifications

Conclusion

The cloud offers tremendous opportunity for increased agility, faster innovation, and lower total cost of ownership. The organizations that are most successful in moving from on-premises environments to the cloud are those that establish a well-defined strategy for approaching this new IT operating model early in their journey. Moving from a model of large upfront investment in data centers to the consumption-based model of AWS requires changes to tools, processes, and mindsets to make sure that costs are effectively managed.

The most important key to effectively optimizing costs with AWS is to start early. Although many cost-optimization practices are relatively easy to implement in small environments, new operational best practices, automation, and organizational incentives are needed to be successful at scale across large environments and enterprises. Establishing these best practices early in the journey can help define the right processes and behaviors to ensure success when hitting scale.

Amazon Web Services is dedicated to supporting customers' cloud journeys and empowering them to maximize value from their investments, improve forecasting accuracy and cost predictability, create a culture of ownership and cost transparency, and continuously measure optimization status.

The potential of cost optimization in the cloud is ultimately unlocked through a cultural commitment to the involvement of cross-functional teams. AWS provides powerful tools to simplify these efforts, but organizational recognition and commitment to the process are critical to success.

Learn more at

amazon.com/aws