AWS Well-Architected Framework

Cost Optimization Pillar

Cost Optimization Pillar: AWS Well-Architected Framework

Table of Contents

Cost Optimization Pillar - AWS Well-Architected Framework

Publication date: **June 27, 2024** (*Document revisions*)

Abstract

This whitepaper focuses on the cost optimization pillar of the Amazon Web Services (AWS) Well-Architected Framework. It provides guidance to help customers apply best practices in the design, delivery, and maintenance of AWS environments.

A cost-optimized workload fully utilizes all resources, achieves an outcome at the lowest possible price point, and meets your functional requirements. This whitepaper provides in-depth guidance for building capability within your organization, designing your workload, selecting your services, configuring and operating the services, and applying cost optimization techniques.

Introduction

The AWS Well-Architected Framework helps you understand the decisions you make while building workloads on AWS. The Framework provides architectural best practices for designing and operating reliable, secure, efficient, cost-effective, and sustainable workloads in the cloud. It demonstrates a way to consistently measure your architectures against best practices and identify areas for improvement. We believe that having well-architected workloads greatly increases the likelihood of business success.

The framework is based on six pillars:

- Operational Excellence
- Security
- Reliability
- Performance Efficiency
- Cost Optimization
- Sustainability

This paper focuses on the cost optimization pillar, and how to architect workloads with the most effective use of services and resources, to achieve business outcomes at the lowest price point.

You'll learn how to apply the best practices of the cost optimization pillar within your organization. Cost optimization can be challenging in traditional on-premises solutions because you must predict future capacity and business needs while navigating complex procurement processes. Adopting the practices in this paper will help your organization achieve the following goals:

- Practice Cloud Financial Management
- Expenditure and usage awareness
- Cost effective resources
- Manage demand and supply resources
- Optimize over time

This paper is intended for those in technology and finance roles, such as chief technology officers (CTOs), chief financial officers (CFOs), architects, developers, financial controllers, financial planners, business analysts, and operations team members. This paper does not provide implementation details or architectural patterns, however, it does include references to appropriate resources.

Cost optimization

Cost optimization is a continual process of refinement and improvement over the span of a workload's lifecycle. The practices in this paper help you build and operate cost-aware workloads that achieve business outcomes while minimizing costs and allowing your organization to maximize its return on investment.

Topics

Design principles

Consider the following design principles for cost optimization:

Implement cloud financial management: To achieve financial success and accelerate business value realization in the cloud, you must invest in Cloud Financial Management. Your organization must dedicate the necessary time and resources for building capability in this new domain of technology and usage management. Similar to your Security or Operations capability, you need to build capability through knowledge building, programs, resources, and processes to help you become a cost efficient organization.

Adopt a consumption model: Pay only for the computing resources you consume, and increase or decrease usage depending on business requirements. For example, development and test environments are typically only used for eight hours a day during the work week. You can stop these resources when they're not in use for a potential cost savings of 75% (40 hours versus 168 hours).

Measure overall efficiency: Measure the business output of the workload and the costs associated with delivery. Use this data to understand the gains you make from increasing output, increasing functionality, and reducing cost.

Stop spending money on undifferentiated heavy lifting: AWS does the heavy lifting of data center operations like racking, stacking, and powering servers. It also removes the operational burden of managing operating systems and applications with managed services. This allows you to focus on your customers and business projects rather than on IT infrastructure.

Analyze and attribute expenditure: The cloud makes it easier to accurately identify the cost and usage of workloads, which then allows transparent attribution of IT costs to revenue streams and individual workload owners. This helps measure return on investment (ROI) and gives workload owners an opportunity to optimize their resources and reduce costs.

Definition

There are five focus areas for cost optimization in the cloud:

- Practice Cloud Financial Management
- Expenditure and usage awareness
- Cost-effective resources
- Manage demand and supplying resources
- Optimize over time

Similar to the other pillars within the Well-Architected Framework, there are trade-offs to consider for cost optimization. For example, whether to optimize for speed-to-market, or for cost. In some cases, it's best to optimize for speed—going to market quickly, shipping new features, or meeting a deadline—rather than investing in upfront cost optimization.

Design decisions are sometimes directed by haste rather than data, and the temptation always exists to overcompensate, rather than spend time benchmarking for the most cost-optimal deployment. Overcompensation can lead to over-provisioned and under-optimized deployments. However, it may be a reasonable choice if you must "lift and shift" resources from your on-premises environment to the cloud and then optimize afterwards.

Investing the right amount of effort in a cost optimization strategy up front allows you to realize the economic benefits of the cloud more readily by ensuring a consistent adherence to best practices and avoiding unnecessary over provisioning. The following sections provide techniques and best practices for the initial and ongoing implementation of Cloud Financial Management and cost optimization for your workloads.

Practice Cloud Financial Management

Managing cloud finance requires evolving your existing finance processes to establish and operate with cost transparency, control, planning, and optimization for your AWS environments.

Applying traditional, static waterfall planning, IT budgeting, and cost assessment models to dynamic cloud usage can create risks, lead to inaccurate planning, and result in less visibility. Ultimately, this results in a lost opportunity to effectively optimize and control costs and realize long-term business value. To avoid these pitfalls, actively manage costs throughout the cloud journey, whether you are building applications natively in the cloud, migrating your workloads to the cloud, or expanding your adoption of cloud services.

Cloud Financial Management (CFM) allows finance, product, technology, and business organizations to manage, optimize, and plan costs as they grow their usage and scale on AWS. The primary goal of CFM is to allow customers to achieve their business outcomes in the most cost-efficient manner and accelerate economic and business value creation while finding the right balance between agility and control.

CFM solutions help transform your business through cost transparency, control, forecasting, and optimization. These solutions can also create a cost-conscious culture that drives accountability across all teams and functions. Finance teams can see where costs are coming from, run operations with minimal unexpected expenses, plan for dynamic cloud usage, and save on cloud expenses while teams scale their adoptions in the cloud. Sharing this with engineering teams can provide necessary financial context for their resource selection, use, and optimization.

AWS CFM offers a set of capabilities to manage, optimize, and plan for cloud costs while maintaining business agility. CFM is paramount not only to effectively manage costs, but also to verify that investments are driving expected business outcomes. These are the four pillars of the Cloud Financial Management Framework in the AWS Cloud: *see*, *save*, *plan*, and *run*. Each of these pillars has a set of activities and capabilities.

The four pillars of Cloud Financial Management.

- **See:** How are you currently measuring, monitoring and creating accountability for your cloud spend? If you are new to AWS or planning on using AWS, do you have a plan to establish cost and usage visibility?

  To understand your AWS costs and optimize spending, you need to know where those costs are coming from. This requires a deliberate structure for your accounts and resources, helping your finance organization track spending flows and hold teams accountable for their portion of the bottom line.

  AWS Services: AWS Control Tower, AWS Organizations, Cost allocations tags, Tag policies, AWS Resource Groups, AWS Cost Categories, AWS Cost Explorer, AWS Cost and Usage Report, RIs and SPs

  Resources: AWS Tagging Best Practices, AWS Cost Categories

- **Save:** What cost optimization levers are you currently using to optimize your spend? If you are not using AWS, are you familiar with common usage-based and pricing model-based optimizations?

  In the save tenet, we optimize costs with pricing and resource recommendations. Optimizing costs begins with having a well-defined strategy for your new cloud operating model. Ideally, this should start as early as possible in your cloud journey, setting the stage for a cost-conscious culture reinforced by the right processes and behaviors.

There are many different ways you can optimize cloud costs. One of them is selecting the right purchase model (RIs and SPs) or whether your workload is immutable and containerized so that you can adopt Amazon EC2 Spot Instances. In addition, scale your workload using Amazon EC2 Auto Scaling Groups.

AWS Services: RIs and SPs, Amazon EC2 Auto Scaling Groups, Spot Instances

Resources: Reserved Instances, Savings Plans, Best practices for handling Amazon EC2

- **Plan:** How do you currently plan for future cloud usage and spend? Do you have a methodology to quantify value generation for a new migration? Have you evolved your current budgeting and forecasting processes to adopt variable usage of the cloud?

  The plan tenet means improving your planning with flexible budgeting and forecasting. Once you've established visibility and cost controls, you will likely want to plan and set expectations for spending on cloud projects. AWS gives you the flexibility to build dynamic forecasting and budgeting processes so you can stay informed on whether costs adhere to, or exceed, budgetary limits.

  AWS Services: AWS Cost Explorer, AWS Cost and Usage Report, AWS Budgets

  Resources: Usage-Based Forecasting, AWS Budget Reports and Alerts

- **Run:** What are some of the operational processes and tools you are currently using to manage your cloud expenditures, and who is leading those efforts? Have you put any thought into how things will work from a daily operations perspective once you start using AWS?

  The run tenet is actually managing billing and cost control. You can establish guardrails and set governance to ensure expenses stay in line with budgets. AWS provides several tools to help you get started.

  AWS Services: AWS Billing and Cost Management Console, AWS Identity and Access Management, Service Control Policies (SCP), AWS Service Catalog, AWS Cost Anomaly Detection, AWS Budgets

  Resources: Getting Started with AWS Billing Console

The following are Cloud Financial Management best practices:

Best practices

COST01-BP01 Establish ownership of cost optimization

Create a team (Cloud Business Office, Cloud Center of Excellence, or FinOps team) that is responsible for establishing and maintaining cost awareness across your organization. The owner of cost optimization can be an individual or a team (requires people from finance, technology, and business teams) that understands the entire organization and cloud finance.

Level of risk exposed if this best practice is not established: High

Implementation guidance

This is the introduction of a Cloud Business Office (CBO) or Cloud Center of Excellence (CCOE) function or team that is responsible for establishing and maintaining a culture of cost awareness in cloud computing. This function can be an existing individual, a team within your organization, or a new team of key finance, technology, and organization stakeholders from across the organization.

The function (individual or team) prioritizes and spends the required percentage of their time on cost management and cost optimization activities. For a small organization, the function might spend a smaller percentage of time compared to a full-time function for a larger enterprise.

The function requires a multi-disciplinary approach, with capabilities in project management, data science, financial analysis, and software or infrastructure development. They can improve workload efficiency by running cost optimizations within three different ownerships:

- **Centralized:** Through designated teams such as FinOps team, Cloud Financial Management (CFM) team, Cloud Business Office (CBO), or Cloud Center of Excellence (CCoE), customers can design and implement governance mechanisms and drive best practices company-wide.

- **Decentralized:** Influencing technology teams to run cost optimizations.
- **Hybrid:** Combination of both centralized and decentralized teams can work together to run cost optimizations.

The function may be measured against their ability to run and deliver against cost optimization goals (for example, workload efficiency metrics).

You must secure executive sponsorship for this function, which is a key success factor. The sponsor is regarded as a champion for cost efficient cloud consumption, and provides escalation support for the team to ensure that cost optimization activities are treated with the level of priority defined by the organization. Otherwise, guidance can be ignored and cost saving opportunities will not be prioritized. Together, the sponsor and team help your organization consume the cloud efficiently and deliver business value.

If you have the Business, Enterprise-On-Ramp or Enterprise support plan and need help building this team or function, reach out to your Cloud Financial Management (CFM) experts through your account team.

Implementation steps

- **Define key members:** All relevant parts of your organization must contribute and be interested in cost management. Common teams within organizations typically include: finance, application or product owners, management, and technical teams (DevOps). Some are engaged full time (finance or technical), while others are engaged periodically as required. Individuals or teams performing CFM need the following set of skills:
  - **Software development:** in the case where scripts and automation are being built out.
  - **Infrastructure engineering:** to deploy scripts, automate processes, and understand how services or resources are provisioned.
  - **Operations acumen:** CFM is about operating on the cloud efficiently by measuring, monitoring, modifying, planning, and scaling efficient use of the cloud.
- **Define goals and metrics:** The function needs to deliver value to the organization in different ways. These goals are defined and continually evolve as the organization evolves. Common activities include: creating and running education programs on cost optimization across the organization, developing organization-wide standards, such as monitoring and reporting for cost optimization, and setting workload goals on optimization. This function also needs to regularly report to the organization on their cost optimization capability.

You can define value- or cost-based key performance indicators (KPIs). When you define the KPIs, you can calculate expected cost in terms of efficiency and expected business outcome. Value-based KPIs tie cost and usage metrics to business value drivers and help rationalize changes in AWS spend. The first step to deriving value-based KPIs is working together, cross-organizationally, to select and agree upon a standard set of KPIs.

- **Establish regular cadence:** The group (finance, technology and business teams) should come together regularly to review their goals and metrics. A typical cadence involves reviewing the state of the organization, reviewing any programs currently running, and reviewing overall financial and optimization metrics. Afterwards, key workloads are reported on in greater detail.

  During these regular reviews, you can review workload efficiency (cost) and business outcome. For example, a 20% cost increase for a workload may align with increased customer usage. In this case, this 20% cost increase can be interpreted as an investment. These regular cadence calls can help teams to identify value KPIs that provide meaning to the entire organization.

Resources

Related documents:

- AWS CCOE Blog
- Creating Cloud Business Office
- CCOE - Cloud Center of Excellence

Related videos:

- Vanguard CCOE Success Story

Related examples:

- Using a Cloud Center of Excellence (CCOE) to Transform the Entire Enterprise
- Building a CCOE to transform the entire enterprise
- 7 Pitfalls to Avoid When Building CCOE

COST01-BP02 Establish a partnership between finance and technology

Involve finance and technology teams in cost and usage discussions at all stages of your cloud journey. Teams regularly meet and discuss topics such as organizational goals and targets, current state of cost and usage, and financial and accounting practices.

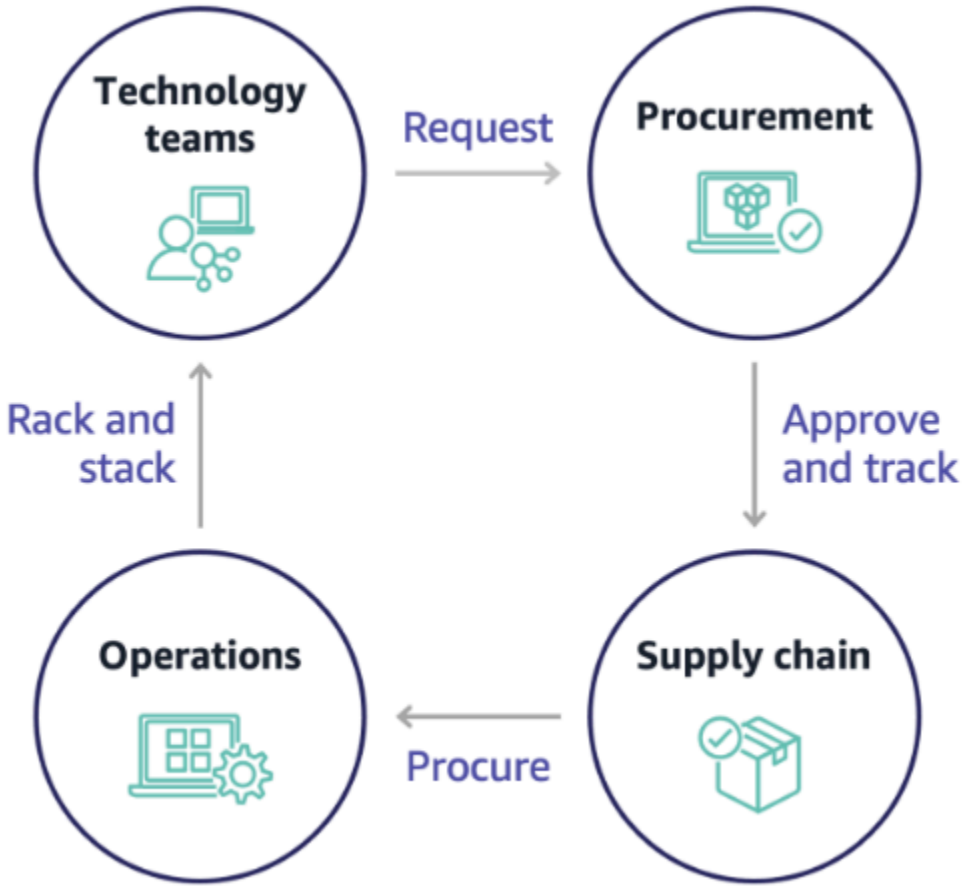
Level of risk exposed if this best practice is not established: High

Implementation guidance

Technology teams innovate faster in the cloud due to shortened approval, procurement, and infrastructure deployment cycles. This can be an adjustment for finance organizations previously used to running time-consuming and resource-intensive processes for procuring and deploying capital in data center and on-premises environments, and cost allocation only at project approval.

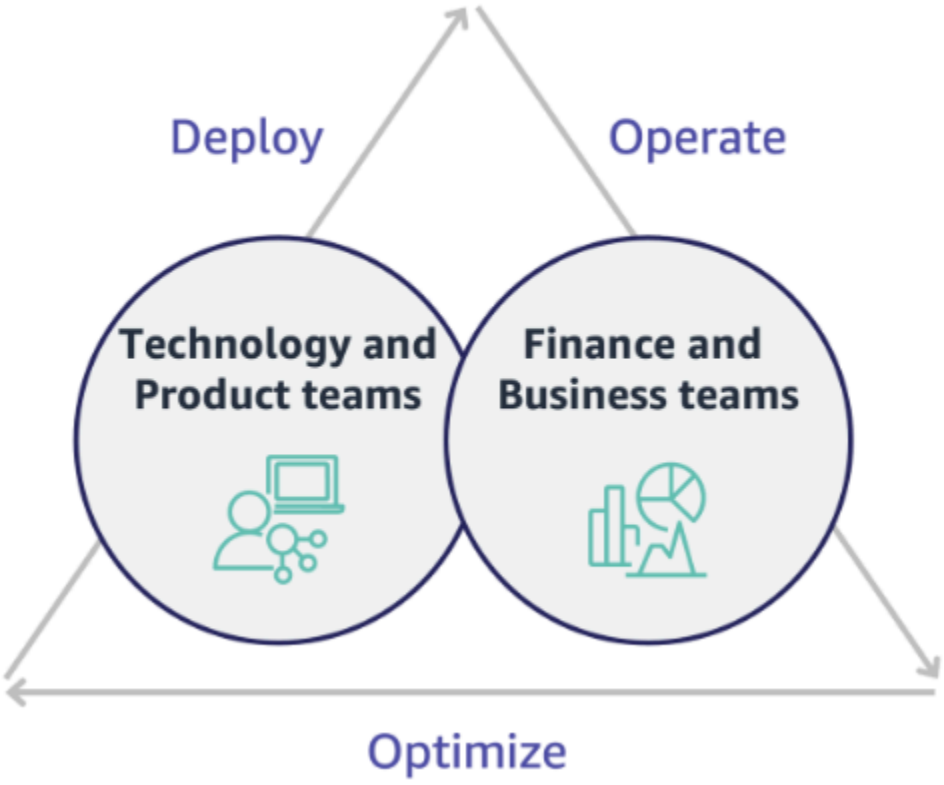
From a finance and procurement organization perspective, the process for capital budgeting, capital requests, approvals, procurement, and installing physical infrastructure is one that has been learned and standardized over decades:

- Engineering or IT teams are typically the requesters
- Various finance teams act as approvers and procurers
- Operations teams rack, stack, and hand off ready-to-use infrastructure



With the adoption of cloud, infrastructure procurement and consumption are no longer beholden to a chain of dependencies. In the cloud model, technology and product teams are no longer just builders, but operators and owners of their products, responsible for most of the activities historically associated with finance and operations teams, including procurement and deployment.

All it really takes to provision cloud resources is an account, and the right set of permissions. This is also what reduces IT and finance risk; which means teams are always a just few clicks or API calls away from terminating idle or unnecessary cloud resources. This is also what allows technology teams to innovate faster – the agility and ability to spin up and then tear down experiments. While the variable nature of cloud consumption may impact predictability from a capital budgeting and forecasting perspective, cloud provides organizations with the ability to reduce the cost of over-provisioning, as well as reduce the opportunity cost associated with conservative under-provisioning.



Establish a partnership between key finance and technology stakeholders to create a shared understanding of organizational goals and develop mechanisms to succeed financially in the variable spend model of cloud computing. Relevant teams within your organization must be involved in cost and usage discussions at all stages of your cloud journey, including:

- **Financial leads:** CFOs, financial controllers, financial planners, business analysts, procurement, sourcing, and accounts payable must understand the cloud model of consumption, purchasing options, and the monthly invoicing process. Finance needs to partner with technology teams to create and socialize an IT value story, helping business teams understand how technology spend is linked to business outcomes. This way, technology expenditures are viewed not as costs, but rather as investments. Due to the fundamental differences between the cloud (such as the rate of change in usage, pay as you go pricing, tiered pricing, pricing models, and detailed billing and usage information) compared to on-premises operation, it is essential that the finance organization understands how cloud usage can impact business aspects including procurement processes, incentive tracking, cost allocation and financial statements.
- **Technology leads:** Technology leads (including product and application owners) must be aware of the financial requirements (for example, budget constraints) as well as business requirements

(for example, service level agreements). This allows the workload to be implemented to achieve the desired goals of the organization.

The partnership of finance and technology provides the following benefits:

- Finance and technology teams have near real-time visibility into cost and usage.
- Finance and technology teams establish a standard operating procedure to handle cloud spend variance.
- Finance stakeholders act as strategic advisors with respect to how capital is used to purchase commitment discounts (for example, Reserved Instances or AWS Savings Plans), and how the cloud is used to grow the organization.
- Existing accounts payable and procurement processes are used with the cloud.
- Finance and technology teams collaborate on forecasting future AWS cost and usage to align and build organizational budgets.
- Better cross-organizational communication through a shared language, and common understanding of financial concepts.

Additional stakeholders within your organization that should be involved in cost and usage discussions include:

- **Business unit owners:** Business unit owners must understand the cloud business model so that they can provide direction to both the business units and the entire company. This cloud knowledge is critical when there is a need to forecast growth and workload usage, and when assessing longer-term purchasing options, such as Reserved Instances or Savings Plans.
- **Engineering team:** Establishing a partnership between finance and technology teams is essential for building a cost-aware culture that encourages engineers to take action on Cloud Financial Management (CFM). One of the common problems of CFM or finance operations practitioners and finance teams is getting engineers to understand the whole business on cloud, follow best practices, and take recommended actions.
- **Third parties:** If your organization uses third parties (for example, consultants or tools), ensure that they are aligned to your financial goals and can demonstrate both alignment through their engagement models and a return on investment (ROI). Typically, third parties will contribute to reporting and analysis of any workloads that they manage, and they will provide cost analysis of any workloads that they design.

Implementing CFM and achieving success requires collaboration across finance, technology, and business teams, and a shift in how cloud spend is communicated and evaluated across the organization. Include engineering teams so that they can be part of these cost and usage discussions at all stages, and encourage them to follow best practices and take agreed-upon actions accordingly.

Implementation steps

- **Define key members:** Verify that all relevant members of your finance and technology teams participate in the partnership. Relevant finance members will be those having interaction with the cloud bill. This will typically be CFOs, financial controllers, financial planners, business analysts, procurement, and sourcing. Technology members will typically be product and application owners, technical managers and representatives from all teams that build on the cloud. Other members may include business unit owners, such as marketing, that will influence usage of products, and third parties such as consultants, to achieve alignment to your goals and mechanisms, and to assist with reporting.
- **Define topics for discussion:** Define the topics that are common across the teams, or will need a shared understanding. Follow cost from that time it is created, until the bill is paid. Note any members involved, and organizational processes that are required to be applied. Understand each step or process it goes through and the associated information, such as pricing models available, tiered pricing, discount models, budgeting, and financial requirements.
- **Establish regular cadence:** To create a finance and technology partnership, establish a regular communication cadence to create and maintain alignment. The group needs to come together regularly against their goals and metrics. A typical cadence involves reviewing the state of the organization, reviewing any programs currently running, and reviewing overall financial and optimization metrics. Then key workloads are reported on in greater detail.

Resources

Related documents:

- AWS News Blog

COST01-BP03 Establish cloud budgets and forecasts

Adjust existing organizational budgeting and forecasting processes to be compatible with the highly variable nature of cloud costs and usage. Processes must be dynamic, using trend-based or business driver-based algorithms or a combination of both.

Level of risk exposed if this best practice is not established: High

Implementation guidance

In traditional on-premises IT setups, customers often face the challenge of planning for fixed costs that change only occasionally, typically with the purchase of new IT hardware and services to meet peak demand. In contrast, AWS Cloud adopts a different approach, where customers pay for the resources they use as dictated by their actual IT and business needs. In the cloud environment, demand can fluctuate on a monthly, daily, or even hourly basis.

Using the cloud brings efficiency, speed, and agility, which results in a highly-variable cost and usage pattern. Costs can decrease or sometimes increase in response to greater workload efficiency or the deployment of new workloads and features. As workloads scale to serve an expanding customer base, cloud usage and costs correspondingly rise due to the increased accessibility of resources. This flexibility in cloud services extends to the costs and forecasts, which creates a degree of elasticity.

It's essential to align closely with these shifting business needs and demand drivers, and aim for the most accurate planning possible. Traditional organizational budget processes need to adapt to accommodate this variability.

Consider cost modelling while you forecast the cost for new workloads. Cost modelling creates a baseline understanding of expected cloud costs, which helps you perform total cost of ownership (TCO), return on investment (ROI), and other financial analysis, set targets and expectations with stakeholders, and identify opportunities for cost optimization.

Your organization should understand the cost definitions and accepted groupings. The level of detail at which you forecast can vary based on your organization's structure and internal workflows. Select a granularity that suits your specific requirements and organizational setup. It is important to understand at what level the forecast is performed:

- **Management account or AWS Organizations level:** The management account is the account that you use to create AWS Organizations. Organizations have one management account by default.

- **Linked or member account:** An account in Organizations is a standard AWS account that contains your AWS resources and the identities that can access those resources.
- **Environment:** An environment is a collection of AWS resources that runs an application version. An environment can be made with multiple linked or member accounts.
- **Project:** A project is a combination of set objectives or tasks to be accomplished within a fixed period. It is important to consider the project lifecycle during your forecast.
- **AWS services:** Groups or categories such as compute or storage services where you can group AWS services for your forecast.
- **Custom grouping:** You can create custom groups based on your organization's needs, such as business units, cost centers, teams, cost allocation tags, cost categories, linked accounts, or combination of these.

Identify the business drivers that can impact your usage cost, and forecast for each of them separately to calculate expected usage in advance. Some of the drivers might be linked to IT and product teams within the organization. Other business drivers, such as marketing events, promotions, geographic expansions, mergers, and acquisitions, are known by your sales, marketing, and business leaders, and it's important to collaborate and account for all those demand drivers as well.

You can use AWS Cost Explorer for trend-based forecasting in a defined future time range based on your past spend. AWS Cost Explorer's forecasting engine segments your historical data based on charge types (for example, Reserved Instances) and uses a combination of machine learning and rule-based models to predict spend across all charge types individually.

Once you've established your forecast process and built models, you can use AWS Budgets to set custom budgets at a granular level by specifying the time period, recurrence, or amount (fixed or variable) and add filters such as service, AWS Region, and tags. The budget is usually prepared for a single year and remains fixed, which requires strict adherence from everyone involved. In contrast, forecasts are more flexible, which allows for readjustments throughout the year and provides dynamic projections over a period of one, two, or three years. Both budgets and forecasts play a crucial role when you establish financial expectations among various technology and business stakeholders. Accurate forecasts and implementation also provides accountability to stakeholders who are directly responsible for provisioning cost in the first place, and it can also raise their overall cost awareness.

To stay informed on the performance of your existing budgets, you can create and schedule AWS Budgets reports to email you and your stakeholders on a regular cadence. You can also create AWS

Budgets alerts based on actual costs, which are reactive in nature, or on forecasted costs, which provides time to implement mitigations against potential cost overruns. You can be alerted when your cost or usage actually exceeds a certain level or if they are forecasted to exceed your budgeted amount.

Adjust existing budget and forecast processes to be more dynamic using trend-based algorithms (with historical costs as inputs) and driver-based algorithms (for example, new product launches, Regional expansion, or new environments for workloads), which are ideal for a dynamic and variable spending environment. Once you've determined your trend-based forecast using Cost Explorer or any other tools, use the AWS Pricing Calculator to estimate your AWS use case and future costs based on the expected usage (traffic, requests-per-second, or required Amazon EC2 instances).

Track the accuracy of that forecast, as budgets should be set based on these forecast calculations and estimations. Monitor the accuracy and effectiveness of the integrated cloud cost forecasts. Regularly review actual spend compared to your forecast, and adjust as needed to improve forecast precision. Track forecast variance, and perform root cause analysis on reported variance to act and adjust forecasts.

As mentioned in the COST01-BP02 Establish a partnership between finance and technology, it is important to foster a partnership and cadence between IT, finance, and other stakeholders to verify that they are all using the same tools or processes for consistency. In cases where budgets may need to change, increase cadence touch points to react to those changes more quickly.

Implementation steps

- **Define the cost language within the organization:** Create a common AWS cost language within the Organization with multiple dimension and grouping. Make sure stakeholders understand forecast granularity, pricing models, and the level of your cost forecasts.
- **Analyze trend-based forecasts:** Use trend-based forecast tools such as AWS Cost Explorer and Amazon Forecast. Analyze your usage cost on multiple dimensions like service, account, tags, and cost categories.
- **Analyze driver-based forecasts:** Identify the impact of business drivers on your cloud usage, and forecast for each of them separately to calculate expected usage cost in advance. Work closely with business unit owners and stakeholders to understand the impact on new drivers, and calculate expected cost changes to define accurate budgets.
- **Update existing forecast and budget processes:** Based on adopted forecast methods such as trend-based, business driver-based, or a combination of both forecasting methods, define

your forecast and budget processes. Budgets should be calculated, realistic, and based on your forecasts.

- **Configure alerts and notifications:** Use AWS Budgets alerts and cost anomaly detection to get alerts and notifications.
- **Perform regular reviews with key stakeholders:** For example, align on changes in business direction and usage with stakeholders in IT, finance, platform teams, and other areas of the business.

Resources

Related documents:

- AWS Cost Explorer
- AWS Cost and Usage Report
- Forecasting with Cost Explorer
- QuickSight Forecasting
- AWS Budgets

Related videos:

- How can I use AWS Budgets to track my spending and usage
- AWS Cost Optimization Series: AWS Budgets

Related examples:

- Understand and build driver-based forecasting
- How to establish and drive a forecasting culture
- How to improve your cloud cost forecasting
- Using the right tools for your cloud cost forecasting

COST01-BP04 Implement cost awareness in your organizational processes

Implement cost awareness, create transparency, and accountability of costs into new or existing processes that impact usage, and leverage existing processes for cost awareness. Implement cost awareness into employee training.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Cost awareness must be implemented in new and existing organizational processes. It is one of the foundational, prerequisite capabilities for other best practices. It is recommended to reuse and modify existing processes where possible — this minimizes the impact to agility and velocity. Report cloud costs to the technology teams and the decision makers in the business and finance teams to raise cost awareness, and establish efficiency key performance indicators (KPIs) for finance and business stakeholders. The following recommendations will help implement cost awareness in your workload:

- Verify that change management includes a cost measurement to quantify the financial impact of your changes. This helps proactively address cost-related concerns and highlight cost savings.
- Verify that cost optimization is a core component of your operating capabilities. For example, you can leverage existing incident management processes to investigate and identify root causes for cost and usage anomalies or cost overruns.
- Accelerate cost savings and business value realization through automation or tooling. When thinking about the cost of implementing, frame the conversation to include an return on investment (ROI) component to justify the investment of time or money.
- Allocate cloud costs by implementing showbacks or chargebacks for cloud spend, including spend on commitment-based purchase options, shared services and marketplace purchases to drive most cost-aware cloud consumption.
- Extend existing training and development programs to include cost-awareness training throughout your organization. It is recommended that this includes continuous training and certification. This will build an organization that is capable of self-managing cost and usage.
- Take advantage of free AWS native tools such as AWS Cost Anomaly Detection, AWS Budgets, and AWS Budgets Reports.

When organizations consistently adopt Cloud Financial Management (CFM) practices, those behaviours become ingrained in the way of working and decision-making. The result is a culture that is more cost-aware, from developers architecting a new born-in-the-cloud application, to finance managers analyzing the ROI on these new cloud investments.

Implementation steps

- **Identify relevant organizational processes:** Each organizational unit reviews their processes and identifies processes that impact cost and usage. Any processes that result in the creation or termination of a resource need to be included for review. Look for processes that can support cost awareness in your business, such as incident management and training.
- **Establish self-sustaining cost-aware culture:** Make sure all the relevant stakeholders align with cause-of-change and impact as a cost so that they understand cloud cost. This will allow your organization to establish a self-sustaining cost-aware culture of innovation.
- **Update processes with cost awareness:** Each process is modified to be made cost aware. The process may require additional pre-checks, such as assessing the impact of cost, or post-checks validating that the expected changes in cost and usage occurred. Supporting processes such as training and incident management can be extended to include items for cost and usage.

To get help, reach out to CFM experts through your Account team, or explore the resources and related documents below.

Resources

Related documents:

- AWS Cloud Financial Management

Related examples:

- Strategy for Efficient Cloud Cost Management
- Cost Control Blog Series #3: How to Handle Cost Shock
- A Beginner's Guide to AWS Cost Management

COST01-BP05 Report and notify on cost optimization

Set up cloud budgets and configure mechanisms to detect anomalies in usage. Configure related tools for cost and usage alerts against pre-defined targets and receive notifications when any usage exceeds those targets. Have regular meetings to analyze the cost-effectiveness of your workloads and promote cost awareness.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

You must regularly report on cost and usage optimization within your organization. You can implement dedicated sessions to discuss cost performance, or include cost optimization in your regular operational reporting cycles for your workloads. Use services and tools to monitor your cost performances regularly and implement cost savings opportunities.

View your cost and usage with multiple filters and granularity by using AWS Cost Explorer, which provides dashboards and reports such as costs by service or by account, daily costs, or marketplace costs. Track your progress of cost and usage against configured budgets with AWS Budgets Reports.

Use AWS Budgets to set custom budgets to track your costs and usage and respond quickly to alerts received from email or Amazon Simple Notification Service (Amazon SNS) notifications if you exceed your threshold. Set your preferred budget period to daily, monthly, quarterly, or annually, and create specific budget limits to stay informed on how actual or forecasted costs and usage progress toward your budget threshold. You can also set up alerts and actions against those alerts to run automatically or through an approval process when a budget target is exceeded.

Implement notifications on cost and usage to ensure that changes in cost and usage can be acted upon quickly if they are unexpected. AWS Cost Anomaly Detection allows you to reduce cost surprises and enhance control without slowing innovation. AWS Cost Anomaly Detection identifies anomalous spend and root causes, which helps to reduce the risk of billing surprises. With three simple steps, you can create your own contextualized monitor and receive alerts when any anomalous spend is detected.

You can also use QuickSight with AWS Cost and Usage Report (CUR) data, to provide highly customized reporting with more granular data. QuickSight allows you to schedule reports and receive periodic Cost Report emails for historical cost and usage or cost-saving opportunities.

Check our Cost Intelligence Dashboard (CID) solution built on QuickSight, which gives you advanced visibility.

Use AWS Trusted Advisor, which provides guidance to verify whether provisioned resources are aligned with AWS best practices for cost optimization.

Check your Savings Plans recommendations through visual graphs against your granular cost and usage. Hourly graphs show On-Demand spend alongside the recommended Savings Plans commitment, providing insight into estimated savings, Savings Plans coverage, and Savings Plans utilization. This helps organizations to understand how their Savings Plans apply to each hour of spend without having to invest time and resources into building models to analyze their spend.

Periodically create reports containing a highlight of Savings Plans, Reserved Instances, and Amazon EC2 rightsizing recommendations from AWS Cost Explorer to start reducing the cost associated with steady-state workloads, idle, and underutilized resources. Identify and recoup spend associated with cloud waste for resources that are deployed. Cloud waste occurs when incorrectly-sized resources are created or different usage patterns are observed instead what is expected. Follow AWS best practices to reduce your waste or ask your account team and partner to help you to optimize and save your cloud costs.

Generate reports regularly for better purchasing options for your resources to drive down unit costs for your workloads. Purchasing options such as Savings Plans, Reserved Instances, or Amazon EC2 Spot Instances offer the deepest cost savings for fault-tolerant workloads and allow stakeholders (business owners, finance, and tech teams) to be part of these commitment discussions.

Share the reports that contain opportunities or new release announcements that may help you to reduce total cost of ownership (TCO) of the cloud. Adopt new services, Regions, features, solutions, or new ways to achieve further cost reductions.

Implementation steps

- **Configure AWS Budgets:** Configure AWS Budgets on all accounts for your workload. Set a budget for the overall account spend, and a budget for the workload by using tags.
  - Well-Architected Labs: Cost and Governance Usage
- **Report on cost optimization:** Set up a regular cycle to discuss and analyze the efficiency of the workload. Using the metrics established, report on the metrics achieved and the cost of achieving them. Identify and fix any negative trends, as well as positive trends that you can

promote across your organization. Reporting should involve representatives from the application teams and owners, finance, and key decision makers with respect to cloud expenditure.

Resources

Related documents:

- AWS Cost Explorer
- AWS Trusted Advisor
- AWS Budgets
- AWS Cost and Usage Report
- AWS Budgets Best Practices
- Amazon S3 Analytics

Related examples:

- Key ways to start optimizing your AWS cloud costs

COST01-BP06 Monitor cost proactively

Implement tooling and dashboards to monitor cost proactively for the workload. Regularly review the costs with configured tools or out of the box tools, do not just look at costs and categories when you receive notifications. Monitoring and analyzing costs proactively helps to identify positive trends and allows you to promote them throughout your organization.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

It is recommended to monitor cost and usage proactively within your organization, not just when there are exceptions or anomalies. Highly visible dashboards throughout your office or work environment ensure that key people have access to the information they need, and indicate the organization's focus on cost optimization. Visible dashboards allow you to actively promote successful outcomes and implement them throughout your organization.

Create a daily or frequent routine to use AWS Cost Explorer or any other dashboard such as Amazon QuickSight to see the costs and analyze proactively. Analyze AWS service usage and costs

at the AWS account-level, workload-level, or specific AWS service-level with grouping and filtering, and validate whether they are expected or not. Use the hourly- and resource-level granularity and tags to filter and identify incurring costs for the top resources. You can also build your own reports with the Cost Intelligence Dashboard, an Amazon QuickSight solution built by AWS Solutions Architects, and compare your budgets with the actual cost and usage.

Implementation steps

- **Report on cost optimization:** Set up a regular cycle to discuss and analyze the efficiency of the workload. Using the metrics established, report on the metrics achieved and the cost of achieving them. Identify and fix any negative trends, and identify positive trends to promote across your organization. Reporting should involve representatives from the application teams and owners, finance, and management.
- **Create and activate daily granularity AWS Budgets for the cost and usage to take timely actions to prevent any potential cost overruns:** AWS Budgets allow you to configure alert notifications, so you stay informed if any of your budget types fall out of your pre-configured thresholds. The best way to leverage AWS Budgets is to set your expected cost and usage as your limits, so that anything above your budgets can be considered overspend.
- **Create AWS Cost Anomaly Detection for cost monitor:** AWS Cost Anomaly Detection uses advanced Machine Learning technology to identify anomalous spend and root causes, so you can quickly take action. It allows you to configure cost monitors that define spend segments you want to evaluate (for example, individual AWS services, member accounts, cost allocation tags, and cost categories), and lets you set when, where, and how you receive your alert notifications. For each monitor, attach multiple alert subscriptions for business owners and technology teams, including a name, a cost impact threshold, and alerting frequency (individual alerts, daily summary, weekly summary) for each subscription.
- **Use AWS Cost Explorer or integrate your AWS Cost and Usage Report (CUR) data with Amazon QuickSight dashboards to visualize your organization's costs:** AWS Cost Explorer has an easy-to-use interface that lets you visualize, understand, and manage your AWS costs and usage over time. The Cost Intelligence Dashboard is a customizable and accessible dashboard to help create the foundation of your own cost management and optimization tool.

Resources

Related documents:

- AWS Budgets

- AWS Cost Explorer
- Daily Cost and Usage Budgets
- AWS Cost Anomaly Detection

Related examples:

- AWS Cost Anomaly Detection Alert with Slack

COST01-BP07 Keep up-to-date with new service releases

Consult regularly with experts or AWS Partners to consider which services and features provide lower cost. Review AWS blogs and other information sources.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

AWS is constantly adding new capabilities so you can leverage the latest technologies to experiment and innovate more quickly. You may be able to implement new AWS services and features to increase cost efficiency in your workload. Regularly review AWS Cost Management, the AWS News Blog, the AWS Cost Management blog, and What's New with AWS for information on new service and feature releases. What's New posts provide a brief overview of all AWS service, feature, and Region expansion announcements as they are released.

Implementation steps

- **Subscribe to blogs:** Go to the AWS blogs pages and subscribe to the What's New Blog and other relevant blogs. You can sign up on the communication preference page with your email address.
- **Subscribe to AWS News:** Regularly review the AWS News Blog and What's New with AWS for information on new service and feature releases. Subscribe to the RSS feed, or with your email to follow announcements and releases.
- **Follow AWS Price Reductions:** Regular price cuts on all our services has been a standard way for AWS to pass on the economic efficiencies to our customers gained from our scale. As of September 20, 2023, AWS has reduced prices 134 times since 2006. If you have any pending business decisions due to price concerns, you can review them again after price reductions and new service integrations. You can learn about the previous price reductions efforts, including

Amazon Elastic Compute Cloud (Amazon EC2) instances, in the price-reduction category of the AWS News Blog.

- **AWS events and meetups:** Attend your local AWS summit, and any local meetups with other organizations from your local area. If you cannot attend in person, try to attend virtual events to hear more from AWS experts and other customers' business cases.
- **Meet with your account team:** Schedule a regular cadence with your account team, meet with them and discuss industry trends and AWS services. Speak with your account manager, Solutions Architect, and support team.

Resources

Related documents:

- AWS Cost Management
- What's New with AWS
- AWS News Blog

Related examples:

- Amazon EC2 – 15 Years of Optimizing and Saving Your IT Costs
- AWS News Blog - Price Reduction

COST01-BP08 Create a cost-aware culture

Implement changes or programs across your organization to create a cost-aware culture. It is recommended to start small, then as your capabilities increase and your organization's use of the cloud increases, implement large and wide ranging programs.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

A cost-aware culture allows you to scale cost optimization and Cloud Financial Management (financial operations, cloud center of excellence, cloud operations teams, and so on) through best practices that are performed in an organic and decentralized manner across your organization.

Cost awareness allows you to create high levels of capability across your organization with minimal effort, compared to a strict top-down, centralized approach.

Creating cost awareness in cloud computing, especially for primary cost drivers in cloud computing, allows teams to understand expected outcomes of any changes in cost perspective. Teams who access the cloud environments should be aware of pricing models and the difference between traditional on-premesis datacenters and cloud computing.

The main benefit of a cost-aware culture is that technology teams optimize costs proactively and continually (for example, they are considered a non-functional requirement when architecting new workloads, or making changes to existing workloads) rather than performing reactive cost optimizations as needed.

Small changes in culture can have large impacts on the efficiency of your current and future workloads. Examples of this include:

- Giving visibility and creating awareness in engineering teams to understand what they do, and what they impact in terms of cost.
- Gamifying cost and usage across your organization. This can be done through a publicly visible dashboard, or a report that compares normalized costs and usage across teams (for example, cost-per-workload and cost-per-transaction).
- Recognizing cost efficiency. Reward voluntary or unsolicited cost optimization accomplishments publicly or privately, and learn from mistakes to avoid repeating them in the future.
- Creating top-down organizational requirements for workloads to run at pre-defined budgets.
- Questioning business requirements of changes, and the cost impact of requested changes to the architecture infrastructure or workload configuration to make sure you pay only what you need.
- Making sure the change planner is aware of expected changes that have a cost impact, and that they are confirmed by the stakeholders to deliver business outcomes cost-effectively.

Implementation steps

- **Report cloud costs to technology teams:** To raise cost awareness, and establish efficiency KPIs for finance and business stakeholders.
- **Inform stakeholders or team members about planned changes:** Create an agenda item to discuss planned changes and the cost-benefit impact on the workload during weekly change meetings.

- **Meet with your account team:** Establish a regular meeting cadence with your account team, and discuss industry trends and AWS services. Speak with your account manager, architect, and support team.
- **Share success stories:** Share success stories about cost reduction for any workload, AWS account, or organization to create a positive attitude and encouragement around cost optimization.
- **Training:** Ensure technical teams or team members are trained for awareness of resource costs on AWS Cloud.
- **AWS events and meetups:** Attend local AWS summits, and any local meetups with other organizations from your local area.
- **Subscribe to blogs:** Go to the AWS blogs pages and subscribe to the What's New Blog and other relevant blogs to follow new releases, implementations, examples, and changes shared by AWS.

Resources

Related documents:

- AWS Blog
- AWS Cost Management
- AWS News Blog

Related examples:

- AWS Cloud Financial Management

COST01-BP09 Quantify business value from cost optimization

Quantifying business value from cost optimization allows you to understand the entire set of benefits to your organization. Because cost optimization is a necessary investment, quantifying business value allows you to explain the return on investment to stakeholders. Quantifying business value can help you gain more buy-in from stakeholders on future cost optimization investments, and provides a framework to measure the outcomes for your organization's cost optimization activities.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

Quantifying the business value means measuring the benefits that businesses gain from the actions and decisions they take. Business value can be tangible (like reduced expenses or increased profits) or intangible (like improved brand reputation or increased customer satisfaction).

To quantify business value from cost optimization means determining how much value or benefit you're getting from your efforts to spend more efficiently. For example, if a company spends $100,000 to deploy a workload on AWS and later optimizes it, the new cost becomes only $80,000 without sacrificing the quality or output. In this scenario, the quantified business value from cost optimization would be a savings of $20,000. But beyond just savings, the business might also quantify value in terms of faster delivery times, improved customer satisfaction, or other metrics that result from the cost optimization efforts. Stakeholders need to make decisions about the potential value of cost optimization, the cost of optimizing the workload, and return value.

In addition to reporting savings from cost optimization, it is recommended that you quantify the additional value delivered. Cost optimization benefits are typically quantified in terms of lower costs per business outcome. For example, you can quantify Amazon Elastic Compute Cloud(Amazon EC2) cost savings when you purchase Savings Plans, which reduce cost and maintain workload output levels. You can quantify cost reductions in AWS spending when idle Amazon EC2 instances are removed, or unattached Amazon Elastic Block Store (Amazon EBS) volumes are deleted.

The benefits from cost optimization, however, go above and beyond cost reduction or avoidance. Consider capturing additional data to measure efficiency improvements and business value.

Implementation steps

- **Evaluate business benefits:** This is the process of analyzing and adjusting AWS Cloud cost in ways that maximize the benefit received from each dollar spent. Instead of focusing on cost reduction without business value, consider business benefits and return on investments for cost optimization, which may bring more value out of the money you spend. It's about spending wisely and making investments and expenditures in areas that yield the best return.
- **Analyze forecasting AWS costs:** Forecasting helps finance stakeholders set expectations with other internal and external organization stakeholders, and can improve your organization's financial predictability. AWS Cost Explorer can be used to perform forecasting for your cost and usage.

Resources

Related documents:

- AWS Cloud Economics
- AWS Blog
- AWS Cost Management
- AWS News Blog
- Well-Architected Reliability Pillar whitepaper
- AWS Cost Explorer

Related videos:

- Unlock Business Value with Windows on AWS

Related examples:

- Measuring and Maximizing the Business Value of Customer 360
- The Business Value of Adopting Amazon Web Services Managed Databases
- The Business Value of Amazon Web Services for Independent Software Vendors
- Business Value of Cloud Modernization
- The Business Value of Migration to Amazon Web Services

Expenditure and usage awareness

Understanding your organization's costs and drivers is critical for managing your cost and usage effectively, and identifying cost-reduction opportunities. Organizations typically operate multiple workloads run by multiple teams. These teams can be in different organization units, each with its own revenue stream. The capability to attribute resource costs to the workloads, individual organization, or product owners drives efficient usage behavior and helps reduce waste. Accurate cost and usage monitoring allows you to understand how profitable organization units and products are, and allows you to make more informed decisions about where to allocate resources within your organization. Awareness of usage at all levels in the organization is key to driving change, as change in usage drives changes in cost.

Consider taking a multi-faceted approach to becoming aware of your usage and expenditures. Your team must gather data, analyze, and then report. Key factors to consider include:

Topics

- Governance
- Monitor cost and usage
- Decommission resources

Governance

To manage your costs in the cloud, you must manage your usage through the following governance areas:

Best practices

- COST02-BP01 Develop policies based on your organization requirements
- COST02-BP02 Implement goals and targets
- COST02-BP03 Implement an account structure
- COST02-BP04 Implement groups and roles
- COST02-BP05 Implement cost controls
- COST02-BP06 Track project lifecycle

COST02-BP01 Develop policies based on your organization requirements

Develop policies that define how resources are managed by your organization and inspect them periodically. Policies should cover the cost aspects of resources and workloads, including creation, modification, and decommissioning over a resource's lifetime.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Understanding your organization's costs and drivers is critical for managing your cost and usage effectively and identifying cost reduction opportunities. Organizations typically operate multiple workloads run by multiple teams. These teams can be in different organization units, each with its own revenue stream. The capability to attribute resource costs to the workloads, individual organization, or product owners drives efficient usage behaviour and helps reduce waste. Accurate cost and usage monitoring helps you understand how optimized a workload is, as well as how profitable organization units and products are. This knowledge allows for more informed decision making about where to allocate resources within your organization. Awareness of usage at all levels in the organization is key to driving change, as change in usage drives changes in cost. Consider taking a multi-faceted approach to becoming aware of your usage and expenditures.

The first step in performing governance is to use your organization's requirements to develop policies for your cloud usage. These policies define how your organization uses the cloud and how resources are managed. Policies should cover all aspects of resources and workloads that relate to cost or usage, including creation, modification, and decommissioning over a resource's lifetime. Verify that policies and procedures are followed and implemented for any change in a cloud environment. During your IT change management meetings, raise questions to find out the cost impact of planned changes, whether increasing or decreasing, the business justification, and the expected outcome.

Policies should be simple so that they are easily understood and can be implemented effectively throughout the organization. Policies also need to be easy to follow and interpret (so they are used) and specific (no misinterpretation between teams). Moreover, they need to be inspected periodically (like our mechanisms) and updated as customers business conditions or priorities change, which would make the policy outdated.

Start with broad, high-level policies, such as which geographic Region to use or times of the day that resources should be running. Gradually refine the policies for the various organizational units

and workloads. Common policies include which services and features can be used (for example, lower performance storage in test and development environments), which types of resources can be used by different groups (for example, the largest size of resource in a development account is medium) and how long these resources will be in use (whether temporary, short term, or for a specific period of time).

Policy example

The following is a sample policy you can review to create your own cloud governance policies, which focus on cost optimization. Make sure you adjust policy based on your organization's requirements and your stakeholders' requests.

- **Policy name:** Define a clear policy name, such as Resource Optimization and Cost Reduction Policy.
- **Purpose:** Explain why this policy should be used and what is the expected outcome. The objective of this policy is to verify that there is a minimum cost required to deploy and run the desired workload to meet business requirements.
- **Scope:** Clearly define who should use this policy and when it should be used, such as DevOps X Team to use this policy in us-east customers for X environment (production or non-production).

Policy statement

1. Select us-east-1or multiple us-east regions based on your workload's environment and business requirement (development, user acceptance testing, pre-production, or production).
2. Schedule Amazon EC2 and Amazon RDS instances to run between six in the morning and eight at night (Eastern Standard Time (EST)).
3. Stop all unused Amazon EC2 instances after eight hours and unused Amazon RDS instances after 24 hours of inactivity.
4. Terminate all unused Amazon EC2 instances after 24 hours of inactivity in non-production environments. Remind Amazon EC2 instance owner (based on tags) to review their stopped Amazon EC2 instances in production and inform them that their Amazon EC2 instances will be terminated within 72 hours if they are not in use.
5. Use generic instance family and size such as m5.large and then resize the instance based on CPU and memory utilization using AWS Compute Optimizer.
6. Prioritize using auto scaling to dynamically adjust the number of running instances based on traffic.

7. Use spot instances for non-critical workloads.
8. Review capacity requirements to commit saving plans or reserved instances for predictable workloads and inform Cloud Financial Management Team.
9. Use Amazon S3 lifecycle policies to move infrequently accessed data to cheaper storage tiers. If no retention policy defined, use Amazon S3 Intelligent Tiering to move objects to archived tier automatically.
10. Monitor resource utilization and set alarms to trigger scaling events using Amazon CloudWatch.
11. For each AWS account, use AWS Budgets to set cost and usage budgets for your account based on cost center and business units.
12. Using AWS Budgets to set cost and usage budgets for your account can help you stay on top of your spending and avoid unexpected bills, allowing you to better control your costs.

Procedure: Provide detailed procedures for implementing this policy or refer to other documents that describe how to implement each policy statement. This section should provide step-by-step instructions for carrying out the policy requirements.

To implement this policy, you can use various third-party tools or AWS Config rules to check for compliance with the policy statement and trigger automated remediation actions using AWS Lambda functions. You can also use AWS Organizations to enforce the policy. Additionally, you should regularly review your resource usage and adjust the policy as necessary to verify that it continues to meet your business needs.

Implementation steps

- **Meet with stakeholders:** To develop policies, ask stakeholders (cloud business office, engineers, or functional decision makers for policy enforcement) within your organization to specify their requirements and document them. Take an iterative approach by starting broadly and continually refine down to the smallest units at each step. Team members include those with direct interest in the workload, such as organization units or application owners, as well as supporting groups, such as security and finance teams.
- **Get confirmation:** Make sure teams agree on policies who can access and deploy to the AWS Cloud. Make sure they follow your organization's policies and confirm that their resource creations align with the agreed policies and procedures.
- **Create onboarding training sessions:** Ask new organization members to complete onboarding training courses to create cost awareness and organization requirements. They may assume different policies from their previous experience or not think of them at all.

- **Define locations for your workload:** Define where your workload operates, including the country and the area within the country. This information is used for mapping to AWS Regions and Availability Zones.
- **Define and group services and resources:** Define the services that the workloads require. For each service, specify the types, the size, and the number of resources required. Define groups for the resources by function, such as application servers or database storage. Resources can belong to multiple groups.
- **Define and group the users by function:** Define the users that interact with the workload, focusing on what they do and how they use the workload, not on who they are or their position in the organization. Group similar users or functions together. You can use the AWS managed policies as a guide.
- **Define the actions:** Using the locations, resources, and users identified previously, define the actions that are required by each to achieve the workload outcomes over its life time (development, operation, and decommission). Identify the actions based on the groups, not the individual elements in the groups, in each location. Start broadly with read or write, then refine down to specific actions to each service.
- **Define the review period:** Workloads and organizational requirements can change over time. Define the workload review schedule to ensure it remains aligned with organizational priorities.
- **Document the policies:** Verify the policies that have been defined are accessible as required by your organization. These policies are used to implement, maintain, and audit access of your environments.

Resources

Related documents:

- Change Management in the Cloud
- AWS Managed Policies for Job Functions
- AWS multiple account billing strategy
- Actions, Resources, and Condition Keys for AWS Services
- AWS Management and Governance
- Control access to AWS Regions using IAM policies
- Global Infrastructures Regions and AZs

Related videos:

- AWS Management and Governance at Scale

COST02-BP02 Implement goals and targets

Implement both cost and usage goals and targets for your workload. Goals provide direction to your organization on expected outcomes, and targets provide specific measurable outcomes to be achieved for your workloads.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Develop cost and usage goals and targets for your organization. As a growing organization on AWS, it is important to set and track goals for cost optimization. These goals or key performance indicators (KPIs) can include things like percent of spend on-demand or adoption of certain optimized services such as AWS Graviton instances or gp3 EBS volume types. Set measurable and achievable goals to help you measure efficiency improvements, which is important for your business operations. Goals provide guidance and direction to your organization on expected outcomes.

Targets provide specific, measurable outcomes to be achieved. In short, a goal is the direction you want to go, and a target is how far in that direction and when that goal should be achieved (use guidance of specific, measurable, assignable, realistic and timely, or SMART). An example of a goal is that platform usage should increase significantly, with only a minor (non-linear) increase in cost. An example target is a 20% increase in platform usage, with less than a five percent increase in costs. Another common goal is that workloads need to be more efficient every six months. The accompanying target would be that the cost per business metrics needs to decrease by five percent every six months. Use the right metrics, and set calculated KPIs for your organization. You can start with basic KPIs and evolve later based on business needs.

A goal for cost optimization is to increase workload efficiency, which corresponds to a decrease in the cost per business outcome of the workload over time. Implement this goal for all workloads, and set a target like a five percent increase in efficiency every six months to a year. In the cloud, you can achieve this through the establishment of capability in cost optimization, as well as new service and feature releases.

Targets are the quantifiable benchmarks you want to reach to meet your goals and benchmarks compare your actual results against a target. Establish benchmarks with KPIs for the cost per unit of compute services (such as Spot adoption, Graviton adoption, latest instance types, and On-Demands coverage), storage services (such as EBS GP3 adoption, obsolete EBS snapshots, and Amazon S3 standard storage), or database service usage (such as RDS open-source engines, Graviton adoption, and On-demand coverage). These benchmarks and KPIs can help you verify that you use AWS services in the most cost-effective manner.

The following table provides a list of standard AWS metrics for reference. Each organization can have different target values for these KPIs.

Category	KPI (%)	Description
Compute	EC2 usage Coverage	EC2 instances (in cost or hours) using SP+RI+Spot compared to total (in cost or hours) of EC2 instances
Compute	Compute SP/RI utilization	Utilized SP or RI hours compared to total available SP or RI hours
Compute	EC2/Hour cost	EC2 cost divided by the number of EC2 instances running in that hour
Compute	vCPU cost	Cost per vCPU for all instances
Compute	Latest Instance Generation	Percentage of instances on Graviton (or other modern generation instance types)
Database	RDS coverage	RDS instances (in cost or hours) using RI compared to total (in cost or hours) of RDS instances

Category	KPI (%)	Description
Database	RDS utilization	Utilized RI hours compared to total available RI hours
Database	RDS uptime	RDS cost divided by the number of RDS instances running in that hour
Database	Latest Instance Generation	Percentage of instances on Graviton (or other modern instance types)
Storage	Storage utilization	Optimized storage cost (for example Glacier, deep archive, or Infrequent Access) divided by total storage cost
Tagging	Untagged resources	Cost Explorer: 1. Filter out credits, discounts , taxes, refunds, marketplace, and copy the latest monthly cost 2. Select **Show only untagged resources** in Cost Explorer 3. Divide the amount in **untagged resources** with your monthly cost.

Using this table, include target or benchmark values, which should be calculated based on your organizational goals. You need to measure certain metrics for your business and understand business outcome for that workload to define accurate and realistic KPIs. When you evaluate performance metrics within an organization, distinguish between different types of metrics that serve distinct purposes. These metrics primarily measure the performance and efficiency of the

technical infrastructure rather than directly the overall business impact. For instance, they might track server response times, network latency, or system uptime. These metrics are crucial to assess how well the infrastructure supports the organization's technical operations. However, they don't provide direct insight into broader business objectives like customer satisfaction, revenue growth, or market share. To gain a comprehensive understanding of business performance, complement these efficiency metrics with strategic business metrics that directly correlate with business outcomes.

Establish near real-time visibility over your KPIs and related savings opportunities and track your progress over time. To get started with the definition and tracking of KPI goals, we recommend the KPI dashboard from Cloud Intelligence Dashboards (CID). Based on the data from Cost and Usage Report (CUR), the KPI dashboard provides a series of recommended cost optimization KPIs, with the ability to set custom goals and track progress over time.

If you have other solutions to set and track KPI goals, make sure these methods are adopted by all cloud financial management stakeholders in your organization.

Implementation steps

- **Define expected usage levels:** To begin, focus on usage levels. Engage with the application owners, marketing, and greater business teams to understand what the expected usage levels are for the workload. How might customer demand change over time, and what can change due to seasonal increases or marketing campaigns?
- **Define workload resourcing and costs:** With usage levels defined, quantify the changes in workload resources required to meet those usage levels. You may need to increase the size or number of resources for a workload component, increase data transfer, or change workload components to a different service at a specific level. Specify the costs at each of these major points, and predict the change in cost when there is a change in usage.
- **Define business goals:** Take the output from the expected changes in usage and cost, combine this with expected changes in technology, or any programs that you are running, and develop goals for the workload. Goals must address usage and cost, as well as the relationship between the two. Goals must be simple, high-level, and help people understand what the business expects in terms of outcomes (such as making sure unused resources are kept below certain cost level). You don't need to define goals for each unused resource type or define costs that can cause losses in goals and targets. Verify that there are organizational programs (for example, capability building like training and education) if there are expected changes in cost without changes in usage.

- **Define targets:** For each of the defined goals, specify a measurable target. If the goal is to increase efficiency in the workload, the target should quantify the amount of improvement (typically in business outputs for each dollar spent) and when it should be delivered. For example, you could set a goal to minimize waste due to over-provisioning. With this goal, your target can be that waste due to compute over-provisioning in the first tier of production workloads should not exceed ten percent of tier compute cost. Additionally, a second target could be that waste due to compute over-provisioning in the second tier of production workloads should not exceed five percent of tier compute cost.

Resources

Related documents:

- AWS managed policies for job functions
- AWS multiple account billing strategy
- Control access to AWS Regions using IAM policies
- S.M.A.R.T. Goals
- How to track your cost optimization KPIs with the CID KPI Dashboard

Related videos:

- Well-Architected Labs: Goals and Targets (Level 100)

Related examples:

- What is a unit metric?
- Selecting a unit metric to support your business
- Unit metrics in practice – lessons learned
- How unit metrics help create alignment between business functions

COST02-BP03 Implement an account structure

Implement a structure of accounts that maps to your organization. This assists in allocating and managing costs throughout your organization.

Level of risk exposed if this best practice is not established: High

Implementation guidance

AWS Organizations allows you to create multiple AWS accounts which can help you centrally govern your environment as you scale your workloads on AWS. You can model your organizational hierarchy by grouping AWS accounts in organizational unit (OU) structure and creating multiple AWS accounts under each OU. To create an account structure, you need to decide which of your AWS accounts will be the management account first. After that, you can create new AWS accounts or select existing accounts as member accounts based on your designed account structure by following management account best practices and member account best practices.

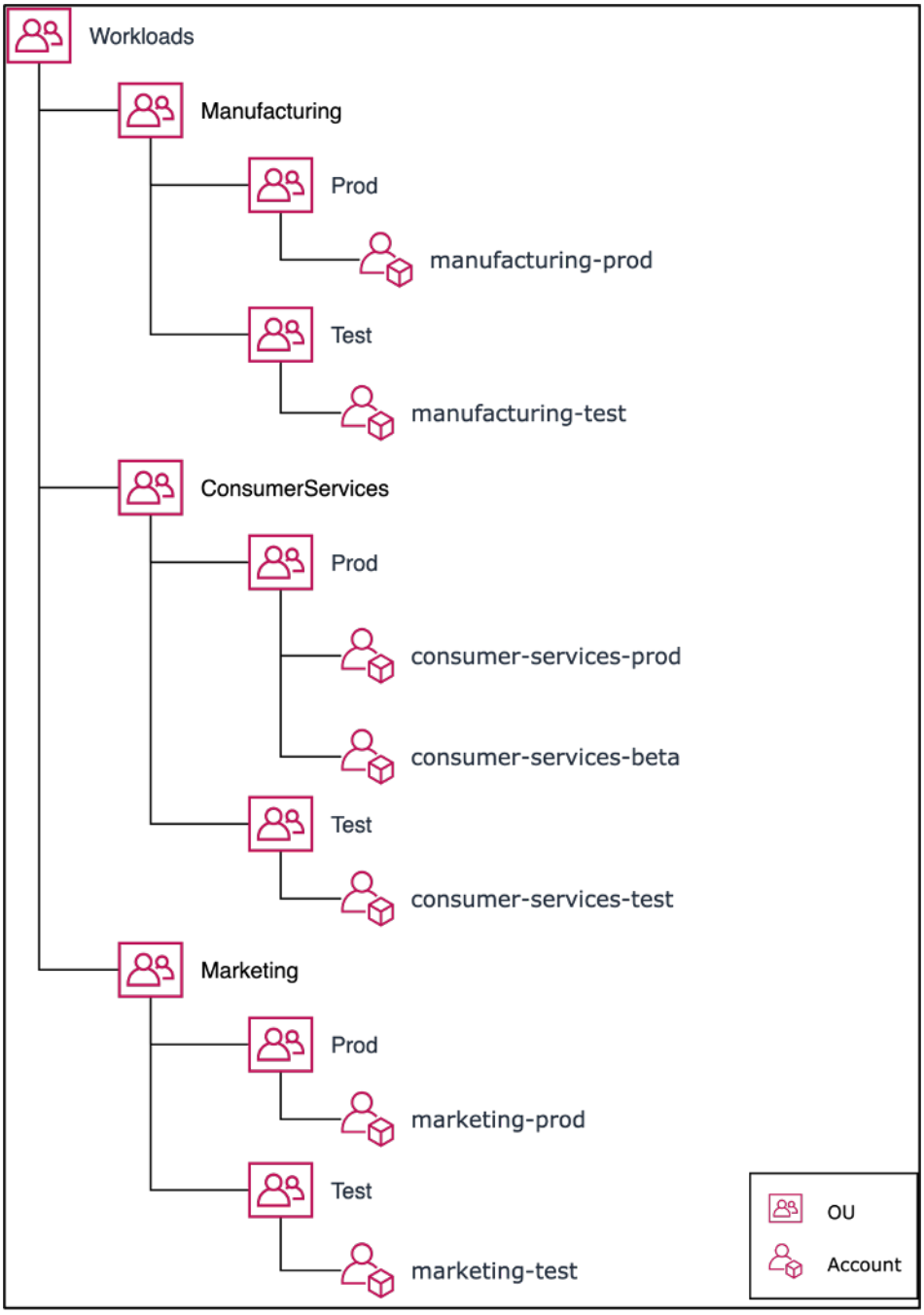
It is advised to always have at least one management account with one member account linked to it, regardless of your organization size or usage. All workload resources should reside only within member accounts and no resource should be created in management account. There is no one size fits all answer for how many AWS accounts you should have. Assess your current and future operational and cost models to ensure that the structure of your AWS accounts reflects your organization's goals. Some companies create multiple AWS accounts for business reasons, for example:

- Administrative or fiscal and billing isolation is required between organization units, cost centers, or specific workloads.
- AWS service limits are set to be specific to particular workloads.
- There is a requirement for isolation and separation between workloads and resources.

Within AWS Organizations, consolidated billing creates the construct between one or more member accounts and the management account. Member accounts allow you to isolate and distinguish your cost and usage by groups. A common practice is to have separate member accounts for each organization unit (such as finance, marketing, and sales), or for each environment lifecycle (such as development, testing and production), or for each workload (workload a, b, and c), and then aggregate these linked accounts using consolidated billing.

Consolidated billing allows you to consolidate payment for multiple member AWS accounts under a single management account, while still providing visibility for each linked account's activity. As costs and usage are aggregated in the management account, this allows you to maximize your service volume discounts, and maximize the use of your commitment discounts (Savings Plans and Reserved Instances) to achieve the highest discounts.

The following diagram shows how you can use AWS Organizations with organizational units (OU) to group multiple accounts, and place multiple AWS accounts under each OU. It is recommended to use OUs for various use cases and workloads which provides patterns for organizing accounts.



Example of grouping multiple AWS accounts under organizational units.

AWS Control Tower can quickly set up and configure multiple AWS accounts, ensuring that governance is aligned with your organization's requirements.

Implementation steps

- **Define separation requirements:** Requirements for separation are a combination of multiple factors, including security, reliability, and financial constructs. Work through each factor in order and specify whether the workload or workload environment should be separate from other workloads. Security promotes adhesion to access and data requirements. Reliability manages limits so that environments and workloads do not impact others. Review the security and reliability pillars of the Well-Architected Framework periodically and follow the provided best practices. Financial constructs create strict financial separation (different cost center, workload ownerships and accountability). Common examples of separation are production and test workloads being run in separate accounts, or using a separate account so that the invoice and billing data can be provided to the individual business units or departments in the organization or stakeholder who owns the account.
- **Define grouping requirements:** Requirements for grouping do not override the separation requirements, but are used to assist management. Group together similar environments or workloads that do not require separation. An example of this is grouping multiple test or development environments from one or more workloads together.
- **Define account structure:** Using these separations and groupings, specify an account for each group and maintain separation requirements. These accounts are your member or linked accounts. By grouping these member accounts under a single management or payer account, you combine usage, which allows for greater volume discounts across all accounts, which provides a single bill for all accounts. It's possible to separate billing data and provide each member account with an individual view of their billing data. If a member account must not have its usage or billing data visible to any other account, or if a separate bill from AWS is required, define multiple management or payer accounts. In this case, each member account has its own management or payer account. Resources should always be placed in member or linked accounts. The management or payer accounts should only be used for management.

Resources

Related documents:

- Using Cost Allocation Tags
- AWS managed policies for job functions
- AWS multiple account billing strategy
- Control access to AWS Regions using IAM policies

- AWS Control Tower
- AWS Organizations
- Best practices for management accounts and member accounts
- Organizing Your AWS Environment Using Multiple Accounts
- Turning on shared reserved instances and Savings Plans discounts
- Consolidated billing
- Consolidated billing

Related examples:

- Splitting the CUR and Sharing Access

Related videos:

- Introducing AWS Organizations
- Set Up a Multi-Account AWS Environment that Uses Best Practices for AWS Organizations

Related examples:

- Defining an AWS Multi-Account Strategy for telecommunications companies
- Best Practices for Optimizing AWS accounts
- Best Practices for Organizational Units with AWS Organizations

COST02-BP04 Implement groups and roles

Implement groups and roles that align to your policies and control who can create, modify, or decommission instances and resources in each group. For example, implement development, test, and production groups. This applies to AWS services and third-party solutions.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

User roles and groups are fundamental building blocks in the design and implementation of secure and efficient systems. Roles and groups help organizations balance the need for control with the requirement for flexibility and productivity, ultimately supporting organizational objectives

and user needs. As recommended in Identity and access management section of AWS Well-Architected Framework Security Pillar, you need robust identity management and permissions in place to provide access to the right resources for the right people under the right conditions. Users receive only the access necessary to complete their tasks. This minimizes the risk associated with unauthorized access or misuse.

After you develop policies, you can create logical groups and user roles within your organization. This allows you to assign permissions, control usage, and help implement robust access control mechanisms, preventing unauthorized access to sensitive information. Begin with high-level groupings of people. Typically, this aligns with organizational units and job roles (for example, a systems administrator in the IT Department, financial controller, or business analysts). The groups categorize people that do similar tasks and need similar access. Roles define what a group must do. It is easier to manage permissions for groups and roles than for individual users. Roles and groups assign permissions consistently and systematically across all users, preventing errors and inconsistencies.

When a user's role changes, administrators can adjust access at the role or group level, rather than reconfiguring individual user accounts. For example, a systems administrator in IT requires access to create all resources, but an analytics team member only needs to create analytics resources.

Implementation steps

- **Implement groups:** Using the groups of users defined in your organizational policies, implement the corresponding groups, if necessary. For best practices on users, groups and authentication, see the Security Pillar of the AWS Well-Architected Framework.
- **Implement roles and policies:** Using the actions defined in your organizational policies, create the required roles and access policies. For best practices on roles and policies, see the Security Pillar of the AWS Well-Architected Framework.

Resources

Related documents:

- AWS managed policies for job functions
- AWS multiple account billing strategy
- AWS Well-Architected Framework Security Pillar
- AWS Identity and Access Management (IAM)
- AWS Identity and Access Management policies

Related videos:

- Why use Identity and Access Management

Related examples:

- Control access to AWS Regions using IAM policies
- Starting your Cloud Financial Management journey: Cloud cost operations

COST02-BP05 Implement cost controls

Implement controls based on organization policies and defined groups and roles. These certify that costs are only incurred as defined by organization requirements such as control access to regions or resource types.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

A common first step in implementing cost controls is to set up notifications when cost or usage events occur outside of policies. You can act quickly and verify if corrective action is required without restricting or negatively impacting workloads or new activity. After you know the workload and environment limits, you can enforce governance. AWS Budgets allows you to set notifications and define monthly budgets for your AWS costs, usage, and commitment discounts (Savings Plans and Reserved Instances). You can create budgets at an aggregate cost level (for example, all costs), or at a more granular level where you include only specific dimensions such as linked accounts, services, tags, or Availability Zones.

Once you set up your budget limits with AWS Budgets, use AWS Cost Anomaly Detection to reduce your unexpected cost. AWS Cost Anomaly Detection is a cost management services that uses machine learning to continually monitor your cost and usage to detect unusual spends. It helps you identify anomalous spend and root causes, so you can quickly take action. First, create a cost monitor in AWS Cost Anomaly Detection, then choose your alerting preference by setting up a dollar threshold (such as an alert on anomalies with impact greater than $1,000). Once you receive alerts, you can analyze the root cause behind the anomaly and impact on your costs. You can also monitor and perform your own anomaly analysis in AWS Cost Explorer.

Enforce governance policies in AWS through AWS Identity and Access Management and AWS Organizations Service Control Policies (SCP). IAM allows you to securely manage access to AWS

services and resources. Using IAM, you can control who can create or manage AWS resources, the type of resources that can be created, and where they can be created. This minimizes the possibility of resources being created outside of the defined policy. Use the roles and groups created previously and assign IAM policies to enforce the correct usage. SCP offers central control over the maximum available permissions for all accounts in your organization, keeping your accounts stay within your access control guidelines. SCPs are available only in an organization that has all features turned on, and you can configure the SCPs to either deny or allow actions for member accounts by default. For more details on implementing access management, see the Well-Architected Security Pillar whitepaper.

Governance can also be implemented through management of AWS service quotas. By ensuring service quotas are set with minimum overhead and accurately maintained, you can minimize resource creation outside of your organization's requirements. To achieve this, you must understand how quickly your requirements can change, understand projects in progress (both creation and decommission of resources), and factor in how fast quota changes can be implemented. Service quotas can be used to increase your quotas when required.

Implementation steps

- **Implement notifications on spend:** Using your defined organization policies, create AWS Budgets to notify you when spending is outside of your policies. Configure multiple cost budgets, one for each account, which notify you about overall account spending. Configure additional cost budgets within each account for smaller units within the account. These units vary depending on your account structure. Some common examples are AWS Regions, workloads (using tags), or AWS services. Configure an email distribution list as the recipient for notifications, and not an individual's email account. You can configure an actual budget for when an amount is exceeded, or use a forecasted budget for notifying on forecasted usage. You can also preconfigure AWS Budget Actions that can enforce specific IAM or SCP policies, or stop target Amazon EC2 or Amazon RDS instances. Budget Actions can be started automatically or require workflow approval.
- **Implement notifications on anomalous spend:** Use AWS Cost Anomaly Detection to reduce your surprise costs in your organization and analyze root cause of potential anomalous spend. Once you create cost monitor to identify unusual spend at your specified granularity and configure notifications in AWS Cost Anomaly Detection, it sends you alert when unusual spend is detected. This will allow you to analyze root case behind the anomaly and understand the impact on your cost. Use AWS Cost Categories while configuring AWS Cost Anomaly Detection to identify which project team or business unit team can analyze the root cause of the unexpected cost and take timely necessary actions.

- **Implement controls on usage:** Using your defined organization policies, implement IAM policies and roles to specify which actions users can perform and which actions they cannot. Multiple organizational policies may be included in an AWS policy. In the same way that you defined policies, start broadly and then apply more granular controls at each step. Service limits are also an effective control on usage. Implement the correct service limits on all your accounts.

Resources

Related documents:

- AWS managed policies for job functions
- AWS multiple account billing strategy
- Control access to AWS Regions using IAM policies
- AWS Budgets
- AWS Cost Anomaly Detection
- Control Your AWS Costs

Related videos:

- How can I use AWS Budgets to track my spending and usage

Related examples:

- Example IAM access management policies
- Example service control policies
- AWS Budgets Actions
- Create IAM Policy to control access to Amazon EC2 resources using Tags
- Restrict the access of IAM Identity to specific Amazon EC2 resources
- Slack integrations for Cost Anomaly Detection using Amazon Q Developer in chat applications

COST02-BP06 Track project lifecycle

Track, measure, and audit the lifecycle of projects, teams, and environments to avoid using and paying for unnecessary resources.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

By effectively tracking the project lifecycle, organizations can achieve better cost control through enhanced planning, management, and resource optimization. The insights gained through tracking are invaluable for making informed decisions that contribute to the cost-effectiveness and overall success of the project.

Tracking the entire lifecycle of the workload helps you understand when workloads or workload components are no longer required. The existing workloads and components may appear to be in use, but when AWS releases new services or features, they can be decommissioned or adopted. Check the previous stages of workloads. After a workload is in production, previous environments can be decommissioned or greatly reduced in capacity until they are required again.

You can tag resources with a timeframe or reminder to pin the time that the workload was reviewed. For example, if the development environment was last reviewed months ago, it could be a good time to review it again to explore if new services can be adopted or if the environment is in use. You can group and tag your applications with myApplications on AWS to manage and track metadata such as criticality, environment, last reviewed, and cost center. You can both track your workload's lifecycle and monitor and manage the cost, health, security posture, and performance of your applications.

AWS provides various management and governance services you can use for entity lifecycle tracking. You can use AWS Config or AWS Systems Manager to provide a detailed inventory of your AWS resources and configuration. It is recommended that you integrate with your existing project or asset management systems to keep track of active projects and products within your organization. Combining your current system with the rich set of events and metrics provided by AWS allows you to build a view of significant lifecycle events and proactively manage resources to reduce unnecessary costs.

Similar to Application Lifecycle Management (ALM), tracking project lifecycle should involve multiple processes, tools, and teams working together, such as design and development, testing, production, support, and workload redundancy.

By carefully monitoring each phase of a project's lifecycle, organizations gain crucial insights and enhanced control, facilitating successful project planning, implementation, and completion. This careful oversight verifies that projects not only meet quality standards, but are delivered on time and within budget, promoting overall cost efficiency.

For more details on implementing entity lifecycle tracking, see AWS Well-Architected Operational Excellence Pillar whitepaper.

Implementation steps

- **Establish project lifecycle monitoring process:** The Cloud Center of Excellence team must establish project lifecycle monitoring process. Establish a structured and systematic approach to monitoring workloads in order to improve control, visibility, and performance of the projects. Make the monitoring process transparent, collaborative, and focused on continuous improvement to maximize its effectiveness and value.
- **Perform workload reviews:** As defined by your organizational policies, set up a regular cadence to audit your existing projects and perform workload reviews. The amount of effort spent in the audit should be proportional to the approximate risk, value, or cost to the organization. Key areas to include in the audit would be risk to the organization of an incident or outage, value, or contribution to the organization (measured in revenue or brand reputation), cost of the workload (measured as total cost of resources and operational costs), and usage of the workload (measured in number of organization outcomes per unit of time). If these areas change over the lifecycle, adjustments to the workload are required, such as full or partial decommissioning.

Resources

Related documents:

- Guidance for Tagging on AWS
- What Is ALM (Application Lifecycle Management)?
- AWS managed policies for job functions

Related examples:

- Control access to AWS Regions using IAM policies

Related Tools

- AWS Config
- AWS Systems Manager
- AWS Budgets

- AWS Organizations
- AWS CloudFormation

Monitor cost and usage

Enable teams to take action on their cost and usage through detailed visibility into the workload. Cost optimization begins with a granular understanding of the breakdown in cost and usage, the ability to model and forecast future spend, usage, and features, and the implementation of sufficient mechanisms to align cost and usage to your organization's objectives. The following are required areas for monitoring your cost and usage:

Best practices

- COST03-BP01 Configure detailed information sources
- COST03-BP02 Add organization information to cost and usage
- COST03-BP03 Identify cost attribution categories
- COST03-BP04 Establish organization metrics
- COST03-BP05 Configure billing and cost management tools
- COST03-BP06 Allocate costs based on workload metrics

COST03-BP01 Configure detailed information sources

Set up cost management and reporting tools for enhanced analysis and transparency of cost and usage data. Configure your workload to create log entries that facilitate the tracking and segregation of costs and usage.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Detailed billing information such as hourly granularity in cost management tools allow organizations to track their consumptions with further details and help them to identify some of the cost increase reasons. These data sources provide the most accurate view of cost and usage across your entire organization.

You can use AWS Data Exports to create exports of the AWS Cost and Usage Report (CUR) 2.0. This is the new and recommended way to receive your detailed cost and usage data from AWS. It

provides daily or hourly usage granularity, rates, costs, and usage attributes for all chargeable AWS services (the same information as CUR), along with some improvements. All possible dimensions are in the CUR such as tagging, location, resource attributes, and account IDs.

There are three export types based on the type of export you want to create: a standard data export, an export to a cost and usage dashboard with QuickSight integration, or a legacy data export.

- **Standard data export:** A customized export of a table that delivers to Amazon S3 on a recurring basis.
- **Cost and usage dashboard:** An export and integration to QuickSight to deploy a pre-built cost and usage dashboard.
- **Legacy data export:** An export of the legacy AWS Cost and Usage Report (CUR).

You can create data exports with the following customizations:

- Include resource IDs
- Split cost allocation data
- Hourly granularity
- Versioning
- Compression type and file format

For your workloads that run containers on Amazon ECS or Amazon EKS, enable split cost allocation data so that you can allocate your container costs to individual business units and teams, based on how your container workloads consume shared compute and memory resources. Split cost allocation data introduces cost and usage data for new container-level resources to AWS Cost and Usage Report. Split cost allocation data is calculated by computing the cost of individual ECS services and tasks running on the cluster.

A cost and usage dashboard exports the cost and usage dashboard table to an S3 bucket on a recurring basis and deploys a prebuilt cost and usage dashboard to QuickSight. Use this option if you want to quickly deploy a dashboard of your cost and usage data without the ability for customization.

If desired, you can still export CUR in legacy mode, where you can integrate other processing services such as AWS Glue to prepare the data for analysis and perform data analysis with Amazon Athena using SQL to query the data.

Implementation steps

- **Create data exports:** Create customized exports with the data you want and control the schema of your exports. Create billing and cost management data exports using basic SQL, and visualize your billing and cost management data by integrating with QuickSight. You can also export your data in standard mode to analyze your data with other processing tools like Amazon Athena.
- **Configure the cost and usage report:** Using the billing console, configure at least one cost and usage report. Configure a report with hourly granularity that includes all identifiers and resource IDs. You can also create other reports with different granularities to provide higher-level summary information.
- **Configure hourly granularity in Cost Explorer:** To access cost and usage data with hourly granularity for the past 14 days, consider enabling hourly and resource level data in the billing console.
- **Configure application logging:** Verify that your application logs each business outcome that it delivers so it can be tracked and measured. Ensure that the granularity of this data is at least hourly so it matches with the cost and usage data. For more details on logging and monitoring, see Well-Architected Operational Excellence Pillar.

Resources

Related documents:

- AWS Data Exports
- AWS Glue
- QuickSight
- AWS Cost Management Pricing
- Tagging AWS resources
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports
- Well-Architected Operational Excellence Pillar

Related examples:

- AWS Account Setup
- Data Exports for AWS Billing and Cost Management

- AWS Cost Explorer Common Use Cases

COST03-BP02 Add organization information to cost and usage

Define a tagging schema based on your organization, workload attributes, and cost allocation categories so that you can filter and search for resources or monitor cost and usage in cost management tools. Implement consistent tagging across all resources where possible by purpose, team, environment, or other criteria relevant to your business.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

Implement tagging in AWS to add organization information to your resources, which will then be added to your cost and usage information. A tag is a key-value pair — the key is defined and must be unique across your organization, and the value is unique to a group of resources. An example of a key-value pair is the key is `Environment`, with a value of `Production`. All resources in the production environment will have this key-value pair. Tagging allows you categorize and track your costs with meaningful, relevant organization information. You can apply tags that represent organization categories (such as cost centers, application names, projects, or owners), and identify workloads and characteristics of workloads (such as test or production) to attribute your costs and usage throughout your organization.

When you apply tags to your AWS resources (such as Amazon Elastic Compute Cloud instances or Amazon Simple Storage Service buckets) and activate the tags, AWS adds this information to your Cost and Usage Reports. You can run reports and perform analysis on tagged and untagged resources to allow greater compliance with internal cost management policies and ensure accurate attribution.

Creating and implementing an AWS tagging standard across your organization's accounts helps you manage and govern your AWS environments in a consistent and uniform manner. Use Tag Policies in AWS Organizations to define rules for how tags can be used on AWS resources in your accounts in AWS Organizations. Tag Policies allow you to easily adopt a standardized approach for tagging AWS resources

AWS Tag Editor allows you to add, delete, and manage tags of multiple resources. With Tag Editor, you search for the resources that you want to tag, and then manage tags for the resources in your search results.

AWS Cost Categories allows you to assign organization meaning to your costs, without requiring tags on resources. You can map your cost and usage information to unique internal organization structures. You define category rules to map and categorize costs using billing dimensions, such as accounts and tags. This provides another level of management capability in addition to tagging. You can also map specific accounts and tags to multiple projects.

Implementation steps

- **Define a tagging schema:** Gather all stakeholders from across your business to define a schema. This typically includes people in technical, financial, and management roles. Define a list of tags that all resources must have, as well as a list of tags that resources should have. Verify that the tag names and values are consistent across your organization.
- **Tag resources:** Using your defined cost attribution categories, place tags on all resources in your workloads according to the categories. Use tools such as the CLI, Tag Editor, or AWS Systems Manager to increase efficiency.
- **Implement AWS Cost Categories:** You can create Cost Categories without implementing tagging. Cost categories use the existing cost and usage dimensions. Create category rules from your schema and implement them into cost categories.
- **Automate tagging:** To verify that you maintain high levels of tagging across all resources, automate tagging so that resources are automatically tagged when they are created. Use services such as AWS CloudFormation to verify that resources are tagged when created. You can also create a custom solution to tag automatically using Lambda functions or use a microservice that scans the workload periodically and removes any resources that are not tagged, which is ideal for test and development environments.
- **Monitor and report on tagging:** To verify that you maintain high levels of tagging across your organization, report and monitor the tags across your workloads. You can use AWS Cost Explorer to view the cost of tagged and untagged resources, or use services such as Tag Editor. Regularly review the number of untagged resources and take action to add tags until you reach the desired level of tagging.

Resources

Related documents:

- Tagging Best Practices
- AWS CloudFormation Resource Tag

- AWS Cost Categories
- Tagging AWS resources
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

Related videos:

- How can I tag my AWS resources to divide up my bill by cost center or project
- Tagging AWS Resources

COST03-BP03 Identify cost attribution categories

Identify organization categories such as business units, departments or projects that could be used to allocate cost within your organization to the internal consuming entities. Use those categories to enforce spend accountability, create cost awareness and drive effective consumption behaviors.

Level of risk exposed if this best practice is not established: High

Implementation guidance

The process of categorizing costs is crucial in budgeting, accounting, financial reporting, decision making, benchmarking, and project management. By classifying and categorizing expenses, teams can gain a better understanding of the types of costs they incur throughout their cloud journey helping teams make informed decisions and manage budgets effectively.

Cloud spend accountability establishes a strong incentive for disciplined demand and cost management. The result is significantly greater cloud cost savings for organizations that allocate most of their cloud spend to consuming business units or teams. Moreover, allocating cloud spend helps organizations adopt more best practices of centralized cloud governance.

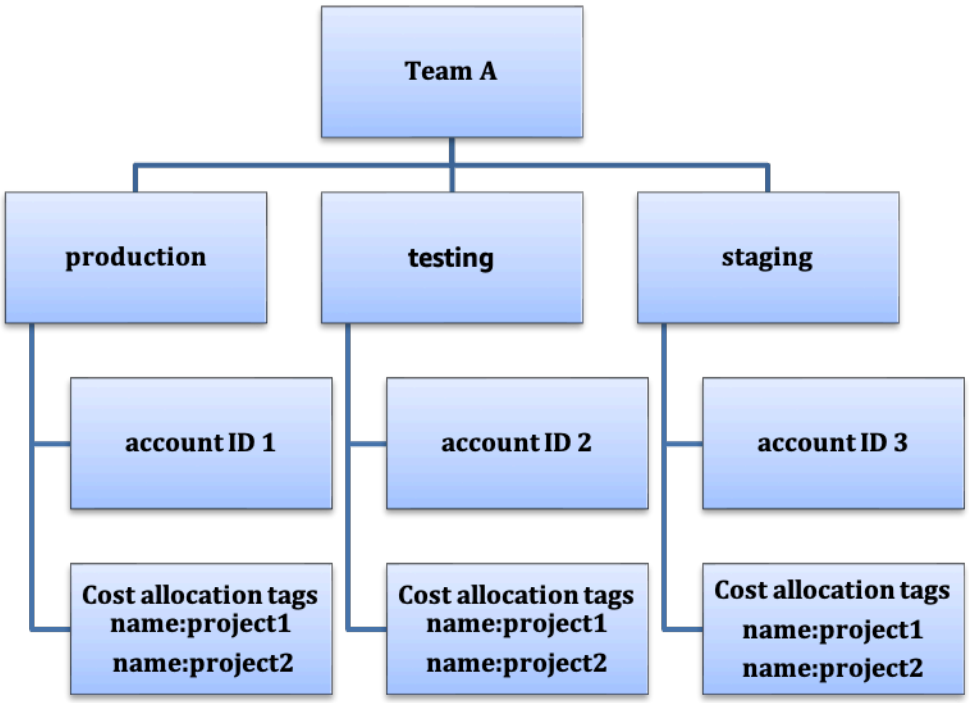
Work with your finance team and other relevant stakeholders to understand the requirements of how costs must be allocated within your organization during your regular cadence calls. Workload costs must be allocated throughout the entire lifecycle, including development, testing, production, and decommissioning. Understand how the costs incurred for learning, staff development, and idea creation are attributed in the organization. This can be helpful to correctly

allocate accounts used for this purpose to training and development budgets instead of generic IT cost budgets.

After defining your cost attribution categories with stakeholders in your organization, use AWS Cost Categories to group your cost and usage information into meaningful categories in the AWS Cloud, such as cost for a specific project, or AWS accounts for departments or business units. You can create custom categories and map your cost and usage information into these categories based on rules you define using various dimensions such as account, tag, service, or charge type. Once cost categories are set up, you can view your cost and usage information by these categories, which allows your organization to make better strategic and purchasing decisions. These categories are visible in AWS Cost Explorer, AWS Budgets, and AWS Cost and Usage Report as well.

For example, create cost categories for your business units (DevOps team), and under each category create multiple rules (rules for each sub category) with multiple dimensions (AWS accounts, cost allocation tags, services or charge type) based on your defined groupings. With cost categories, you can organize your costs using a rule-based engine. The rules that you configure organize your costs into categories. Within these rules, you can filter with using multiple dimensions for each category such as specific AWS accounts, AWS services, or charge types. You can then use these categories across multiple products in the AWS Billing and Cost Management and Cost Management console. This includes AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report, and AWS Cost Anomaly Detection.

As an example, the following diagram displays how to group your costs and usage information in your organization by having multiple teams (cost category), multiple environments (rules), and each environment having multiple resources or assets (dimensions).



Cost and usage organization chart

You can create groupings of costs using cost categories as well. After you create the cost categories (allowing up to 24 hours after creating a cost category for your usage records to be updated with values), they appear in AWS Cost Explorer, AWS Budgets, AWS Cost and Usage Report, and AWS Cost Anomaly Detection. In AWS Cost Explorer and AWS Budgets, a cost category appears as an additional billing dimension. You can use this to filter for the specific cost category value, or group by the cost category.

Implementation steps

- **Define your organization categories:** Meet with internal stakeholders and business units to define categories that reflect your organization's structure and requirements. These categories should directly map to the structure of existing financial categories, such as business unit, budget, cost center, or department. Look at the outcomes the cloud delivers for your business such as training or education, as these are also organization categories.
- **Define your functional categories:** Meet with internal stakeholders and business units to define categories that reflect the functions that you have within your business. This may be the workload or application names, and the type of environment, such as production, testing, or development.

- **Define AWS Cost Categories:** Create cost categories to organize your cost and usage information with using AWS Cost Categories and map your AWS cost and usage into meaningful categories. Multiple categories can be assigned to a resource, and a resource can be in multiple different categories, so define as many categories as needed so that you can manage your costs within the categorized structure using AWS Cost Categories.

Resources

Related documents:

- Tagging AWS resources
- Using Cost Allocation Tags
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports
- AWS Cost Categories
- Managing your costs with AWS Cost Categories
- Creating cost categories
- Tagging cost categories
- Splitting charges within cost categories
- AWS Cost Categories Features

Related examples:

- Organize your cost and usage data with AWS Cost Categories
- Managing your costs with AWS Cost Categories

COST03-BP04 Establish organization metrics

Establish the organization metrics that are required for this workload. Example metrics of a workload are customer reports produced, or web pages served to customers.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Understand how your workload's output is measured against business success. Each workload typically has a small set of major outputs that indicate performance. If you have a complex workload with many components, then you can prioritize the list, or define and track metrics for each component. Work with your teams to understand which metrics to use. This unit will be used to understand the efficiency of the workload, or the cost for each business output.

Implementation steps

- **Define workload outcomes:** Meet with the stakeholders in the business and define the outcomes for the workload. These are a primary measure of customer usage and must be business metrics and not technical metrics. There should be a small number of high-level metrics (less than five) per workload. If the workload produces multiple outcomes for different use cases, then group them into a single metric.
- **Define workload component outcomes:** Optionally, if you have a large and complex workload, or can easily break your workload into components (such as microservices) with well-defined inputs and outputs, define metrics for each component. The effort should reflect the value and cost of the component. Start with the largest components and work towards the smaller components.

Resources

Related documents:

- Tagging AWS resources
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

COST03-BP05 Configure billing and cost management tools

Configure cost management tools that meet your organization's policies to manage and optimize cloud spending. This includes services, tools, and resources to organize and track cost and usage data, enhance control through consolidated billing and access permission, improve planning through budgeting and forecasts, receive notifications or alerts, and lower cost with resources and pricing optimizations.

Level of risk exposed if this best practice is not established: High

Implementation guidance

To establish strong accountability, consider your account strategy first as part of your cost allocation strategy. Get this right, and you may not need to go any further. Otherwise, there can be unawareness and further pain points.

To encourage accountability of cloud spend, grant users access to tools that provide visibility into their costs and usage. AWS recommends that you configure all workloads and teams for the following purposes:

- **Organize:** Establish your cost allocation and governance baseline with your own tagging strategy and taxonomy. Create multiple AWS Accounts with tools such as AWS Control Tower or AWS Organization. Tag the supported AWS resources and categorize them meaningfully based on your organization structure (business units, departments, or projects). Tag account names for specific cost centers and map them with AWS Cost Categories to group accounts for business units to their cost centers so that business unit owner can see multiple accounts' consumption in one place.
- **Access:** Track organization-wide billing information in consolidated billing. Verify the right stakeholders and business owners have access.
- **Control:** Build effective governance mechanisms with the right guardrails to prevent unexpected scenarios when using Service Control Policies (SCP), tag policies, IAM policies and budget alerts. For example, you can allow teams to create specific resources in preferred regions only by using effective control mechanisms and prevent resource creations without specific tag (such as cost-center).
- **Current state:** Configure a dashboard that shows current levels of cost and usage. The dashboard should be available in a highly visible place within the work environment like an operations dashboard. You can export data and use the Cost and Usage Dashboard from the AWS Cost Optimization Hub or any supported product to create this visibility. You may need to create different dashboards for different personas. For example, manager dashboard may differ from an engineering dashboard.
- **Notifications:** Provide notifications when cost or usage exceeds defined limits and anomalies occur with AWS Budgets or AWS Cost Anomaly Detection.
- **Reports:** Summarize all cost and usage information. Raise awareness and accountability of your cloud spend with detailed, attributable cost data. Create reports that are relevant to the team consuming them and contain recommendations.

- **Tracking:** Show the current cost and usage against configured goals or targets.
- **Analysis:** Allow team members to perform custom and deep analysis down to the hourly, daily or monthly granularity with different filters (resource, account, tag, etc.).
- **Inspect:** Stay up to date with your resource deployment and cost optimization opportunities. Get notifications using Amazon CloudWatch, Amazon SNS, or Amazon SES for resource deployments at the organization level. Review cost optimization recommendations with AWS Trusted Advisor or AWS Compute Optimizer.
- **Trend reports:** Display the variability in cost and usage over the required period with the required granularity.
- **Forecasts:** Show estimated future costs, estimate your resource usage, and spend with forecast dashboards you create.

You can use AWS Cost Optimization Hub to understand potential cost-saving opportunities consolidated from a centralized location and create data exports for integration with Amazon Athena. You can also use the AWS Cost Optimization Hub to deploy the Cost and Usage Dashboard, which utilizes QuickSight for interactive cost analysis and secure cost insight sharing.

If you don't have essential skills or bandwidth in your organization, you can work with AWS ProServ, AWS Managed Services (AMS), or AWS Partners. You can also use third-party tools but ensure you validate the value proposition.

Implementation steps

- **Allow team-based access to tools:** Configure your accounts and create groups that have access to the required cost and usage reports for their consumptions and use AWS Identity and Access Management to control access to the tools such as AWS Cost Explorer. These groups must include representatives from all teams that own or manage an application. This certifies that every team has access to their cost and usage information to track their consumption.
- **Organize Costs Tags and Categories:** organize your costs across teams, business units, applications, environments, and projects. Use resource tags to organize costs, by cost allocation tags. Create Cost Categories based on the dimensions with using tags, accounts, services, etc. to map your costs.
- **Configure AWS Budgets:** Configure AWS Budgets on all accounts for your workloads. Set budgets for the overall account spend, and budgets for the workloads by using tags and cost categories. Configure notifications in AWS Budgets to receive alerts for when you exceed your budgeted amounts, or when your estimated costs exceed your budgets.

- **Configure AWS Cost Anomaly Detection:** Use AWS Cost Anomaly Detection for your accounts, core services or cost categories you created to monitor your cost and usage and detect unusual spends. You can receive alerts individually in aggregated reports and receive alerts in an email or an Amazon SNS topic which allows you to analyze and determine the root cause of the anomaly and identify the factor that is driving the cost increase.
- **Use cost analysis tools:** Configure AWS Cost Explorer for your workload and accounts to visualize your cost data for further analysis. Create a dashboard for the workload that tracks overall spend, key usage metrics for the workload, and forecast of future costs based on your historical cost data.
- **Use cost-saving analysis tools:** Use AWS Cost Optimization Hub to identify savings opportunities with tailored recommendations including deleting unused resources, rightsizing, savings Plans, reservations and compute optimizer recommendations.
- **Configure advanced tools:** You can optionally create visuals to facilitate interactive analysis and sharing of cost insights. With Data Exports on AWS Cost Optimization Hub, you can create cost and usage dashboard powered by QuickSight for your organization that provides additional detail and granularity. You can also implement advanced analysis capability with using data exports in Amazon Athena for advanced queries, and create dashboards on QuickSight. Work with AWS Partners to adopt cloud management solutions for consolidated cloud bill monitoring and optimization.

Resources

Related documents:

- What is AWS Billing and Cost Management and Cost Management?
- Establishing your best practice AWS environment
- Best Practices for Tagging AWS Resources
- Tagging your AWS resources
- AWS Cost Categories
- Analyzing your costs with AWS Budgets
- Analyzing your costs with AWS Cost Explorer
- What is AWS Data Exports?

Related videos:

- Deploying Cloud Intelligence Dashboards
- Get Alerts on any FinOps or Cost Optimization Metric or KPI

Related examples:

- Cost and Usage Dashboard powered by QuickSight
- AWS Cost and Usage Governance Workshop

COST03-BP06 Allocate costs based on workload metrics

Allocate the workload's costs based on usage metrics or business outcomes to measure workload cost efficiency. Implement a process to analyze the cost and usage data with analytics services, which can provide insight and charge back capability.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

Cost optimization means delivering business outcomes at the lowest price point, which can only be achieved by allocating workload costs based on workload metrics (measured by workload efficiency). Monitor the defined workload metrics through log files or other application monitoring. Combine this data with the workload's costs, which can be obtained by looking at costs with a specific tag value or account ID. Perform this analysis at the hourly level. Your efficiency typically changes if you have static cost components (for example, a backend database running permanently) with a varying request rate (for example, usage peaks at nine in the morning to five in the evening, with few requests at night). Understanding the relationship between the static and variable costs helps you focus your optimization activities.

Creating workload metrics for shared resources may be challenging compared to resources like containerized applications on Amazon Elastic Container Service (Amazon ECS) and Amazon API Gateway. However, there are certain ways you can categorize usage and track cost. If you need to track Amazon ECS and AWS Batch shared resources, you can enable split cost allocation data in AWS Cost Explorer. With split cost allocation data, you can understand and optimize the cost and usage of your containerized applications and allocate application costs back to individual business entities based on how shared compute and memory resources are consumed.

Implementation steps

- **Allocate costs to workload metrics:** Using the defined metrics and configured tags, create a metric that combines the workload output and workload cost. Use analytics services such as Amazon Athena and Amazon QuickSight to create an efficiency dashboard for the overall workload and any components.

Resources

Related documents:

- Tagging AWS resources
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

Related examples:

- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

Decommission resources

After you manage a list of projects, employees, and technology resources over time you will be able to identify which resources are no longer being used, and which projects that no longer have an owner.

Best practices

- COST04-BP01 Track resources over their lifetime
- COST04-BP02 Implement a decommissioning process
- COST04-BP03 Decommission resources
- COST04-BP04 Decommission resources automatically
- COST04-BP05 Enforce data retention policies

COST04-BP01 Track resources over their lifetime

Define and implement a method to track resources and their associations with systems over their lifetime. You can use tagging to identify the workload or function of the resource.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Decommission workload resources that are no longer required. A common example is resources used for testing: after testing has been completed, the resources can be removed. Tracking resources with tags (and running reports on those tags) can help you identify assets for decommission, as they will not be in use or the license on them will expire. Using tags is an effective way to track resources, by labeling the resource with its function, or a known date when it can be decommissioned. Reporting can then be run on these tags. Example values for feature tagging are `feature-X testing` to identify the purpose of the resource in terms of the workload lifecycle. Another example is using `LifeSpan` or `TTL` for the resources, such as to-be-deleted tag key name and value to define the time period or specific time for decommissioning.

Implementation steps

- **Implement a tagging scheme:** Implement a tagging scheme that identifies the workload the resource belongs to, verifying that all resources within the workload are tagged accordingly. Tagging helps you categorize resources by purpose, team, environment, or other criteria relevant to your business. For more detail on tagging uses cases, strategies, and techniques, see AWS Tagging Best Practices.
- **Implement workload throughput or output monitoring:** Implement workload throughput monitoring or alarming, initiating on either input requests or output completions. Configure it to provide notifications when workload requests or outputs drop to zero, indicating the workload resources are no longer used. Incorporate a time factor if the workload periodically drops to zero under normal conditions. For more detail on unused or underutilized resources, see AWS Trusted Advisor Cost Optimization checks.
- **Group AWS resources:** Create groups for AWS resources. You can use AWS Resource Groups to organize and manage your AWS resources that are in the same AWS Region. You can add tags to most of your resources to help identify and sort your resources within your organization. Use Tag Editor add tags to supported resources in bulk. Consider using AWS Service Catalog to create, manage, and distribute portfolios of approved products to end users and manage the product lifecycle.

Resources

Related documents:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS Trusted Advisor Cost Optimization Checks
- Tagging AWS resources
- Publishing Custom Metrics

Related videos:

- How to optimize costs using AWS Trusted Advisor

Related examples:

- Organize AWS resources
- Optimize cost using AWS Trusted Advisor

COST04-BP02 Implement a decommissioning process

Implement a process to identify and decommission unused resources.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Implement a standardized process across your organization to identify and remove unused resources. The process should define the frequency searches are performed and the processes to remove the resource to verify that all organization requirements are met.

Implementation steps

- **Create and implement a decommissioning process:** Work with the workload developers and owners to build a decommissioning process for the workload and its resources. The process should cover the method to verify if the workload is in use, and also if each of the workload resources are in use. Detail the steps necessary to decommission the resource, removing them from service while ensuring compliance with any regulatory requirements. Any associated

resources should be included, such as licenses or attached storage. Notify the workload owners that the decommissioning process has been started.

Use the following decommission steps to guide you on what should be checked as part of your process:

- **Identify resources to be decommissioned:** Identify resources that are eligible for decommissioning in your AWS Cloud. Record all necessary information and schedule the decommission. In your timeline, be sure to account for if (and when) unexpected issues arise during the process.
- **Coordinate and communicate:** Work with workload owners to confirm the resource to be decommissioned
- **Record metadata and create backups:** Record metadata (such as public IPs, Region, AZ, VPC, Subnet, and Security Groups) and create backups (such as Amazon Elastic Block Store snapshots or taking AMI, keys export, and Certificate export) if it is required for the resources in the production environment or if they are critical resources.
- **Validate infrastructure-as-code:** Determine whether resources were deployed with AWS CloudFormation, Terraform, AWS Cloud Development Kit (AWS CDK), or any other infrastructure-as-code deployment tool so they can be re-deployed if necessary.
- **Prevent access:** Apply restrictive controls for a period of time, to prevent the use of resources while you determine if the resource is required. Verify that the resource environment can be reverted to its original state if required.
- **Follow your internal decommissioning process:** Follow the administrative tasks and decommissioning process of your organization, like removing the resource from your organization domain, removing the DNS record, and removing the resource from your configuration management tool, monitoring tool, automation tool and security tools.

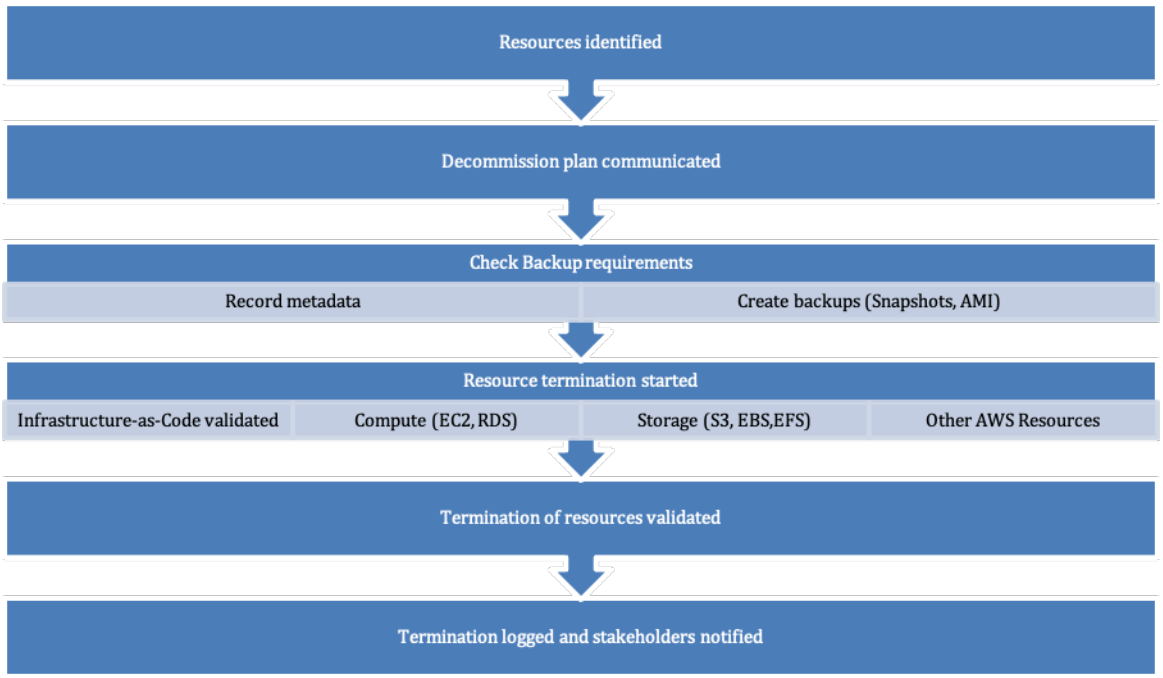
If the resource is an Amazon EC2 instance, consult the following list. For more detail, see How do I delete or terminate my Amazon EC2 resources?

- Stop or terminate all your Amazon EC2 instances and load balancers. Amazon EC2 instances are visible in the console for a short time after they're terminated. You aren't billed for any instances that aren't in the running state
- Delete your Auto Scaling infrastructure.
- Release all Dedicated Hosts.
- Delete all Amazon EBS volumes and Amazon EBS snapshots.
- Release all Elastic IP addresses.

- Deregister all Amazon Machine Images (AMIs).
- Terminate all AWS Elastic Beanstalk environments.

If the resource is an object in Amazon S3 Glacier storage and if you delete an archive before meeting the minimum storage duration, you will be charged a prorated early deletion fee. Amazon S3 Glacier minimum storage duration depends on the storage class used. For a summary of minimum storage duration for each storage class, see Performance across the Amazon S3 storage classes. For detail on how early deletion fees are calculated, see Amazon S3 pricing.

The following simple decommissioning process flowchart outlines the decommissioning steps. Before decommissioning resources, verify that resources you have identified for decommissioning are not being used by the organization.



Resource decommissioning flow.

Resources

Related documents:

- AWS Auto Scaling
- AWS Trusted Advisor
- AWS CloudTrail

Related videos:

- Delete CloudFormation stack but retain some resources
- Find out which user launched Amazon EC2 instance

Related examples:

- Delete or terminate Amazon EC2 resources
- Find out which user launched an Amazon EC2 instance

COST04-BP03 Decommission resources

Decommission resources initiated by events such as periodic audits, or changes in usage. Decommissioning is typically performed periodically and can be manual or automated.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

The frequency and effort to search for unused resources should reflect the potential savings, so an account with a small cost should be analyzed less frequently than an account with larger costs. Searches and decommission events can be initiated by state changes in the workload, such as a product going end of life or being replaced. Searches and decommission events may also be initiated by external events, such as changes in market conditions or product termination.

Implementation steps

- **Decommission resources:** This is the depreciation stage of AWS resources that are no longer needed or ending of a licensing agreement. Complete all final checks completed before moving to the disposal stage and decommissioning resources to prevent any unwanted disruptions like taking snapshots or backups. Using the decommissioning process, decommission each of the resources that have been identified as unused.

Resources

Related documents:

- AWS Auto Scaling

- AWS Trusted Advisor

COST04-BP04 Decommission resources automatically

Design your workload to gracefully handle resource termination as you identify and decommission non-critical resources, resources that are not required, or resources with low utilization.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

Use automation to reduce or remove the associated costs of the decommissioning process. Designing your workload to perform automated decommissioning will reduce the overall workload costs during its lifetime. You can use Amazon EC2 Auto Scaling or Application Auto Scaling to perform the decommissioning process. You can also implement custom code using the API or SDK to decommission workload resources automatically.

Modern applications are built serverless-first, a strategy that prioritizes the adoption of serverless services. AWS developed serverless services for all three layers of your stack: compute, integration, and data stores. Using serverless architecture will allow you to save costs during low-traffic periods with scaling up and down automatically.

Implementation steps

- **Implement Amazon EC2 Auto Scaling or Application Auto Scaling:** For resources that are supported, configure them with Amazon EC2 Auto Scaling or Application Auto Scaling. These services can help you optimize your utilization and cost efficiencies when consuming AWS services. When demand drops, these services will automatically remove any excess resource capacity so you avoid overspending.
- **Configure CloudWatch to terminate instances:** Instances can be configured to terminate using CloudWatch alarms. Using the metrics from the decommissioning process, implement an alarm with an Amazon Elastic Compute Cloud action. Verify the operation in a non-production environment before rolling out.
- **Implement code within the workload:** You can use the AWS SDK or AWS CLI to decommission workload resources. Implement code within the application that integrates with AWS and terminates or removes resources that are no longer used.
- **Use serverless services:** Prioritize building serverless architectures and event-driven architecture on AWS to build and run your applications. AWS offers multiple serverless technology

services that inherently provide automatically optimized resource utilization and automated decommissioning (scale in and scale out). With serverless applications, resource utilization is automatically optimized and you never pay for over-provisioning.

Resources

Related documents:

- Amazon EC2 Auto Scaling
- Getting Started with Amazon EC2 Auto Scaling
- Application Auto Scaling
- AWS Trusted Advisor
- Serverless on AWS
- Create Alarms to Stop, Terminate, Reboot, or Recover an Instance
- Adding terminate actions to Amazon CloudWatch alarms

Related examples:

- Scheduling automatic deletion of AWS CloudFormation stacks

COST04-BP05 Enforce data retention policies

Define data retention policies on supported resources to handle object deletion per your organizations' requirements. Identify and delete unnecessary or orphaned resources and objects that are no longer required.

Level of risk exposed if this best practice is not established: Medium

Use data retention policies and lifecycle policies to reduce the associated costs of the decommissioning process and storage costs for the identified resources. Defining your data retention policies and lifecycle policies to perform automated storage class migration and deletion will reduce the overall storage costs during its lifetime. You can use Amazon Data Lifecycle Manager to automate the creation and deletion of Amazon Elastic Block Store snapshots and Amazon EBS-backed Amazon Machine Images (AMIs), and use Amazon S3 Intelligent-Tiering or an Amazon S3 lifecycle configuration to manage the lifecycle of your Amazon S3 objects. You can also

implement custom code using the API or SDK to create lifecycle policies and policy rules for objects to be deleted automatically.

Implementation steps

- **Use Amazon Data Lifecycle Manager:** Use lifecycle policies on Amazon Data Lifecycle Manager to automate deletion of Amazon EBS snapshots and Amazon EBS-backed AMIs.
- **Set up lifecycle configuration on a bucket:** Use Amazon S3 lifecycle configuration on a bucket to define actions for Amazon S3 to take during an object's lifecycle, as well as deletion at the end of the object's lifecycle, based on your business requirements.

Resources

Related documents:

- AWS Trusted Advisor
- Amazon Data Lifecycle Manager
- How to set lifecycle configuration on Amazon S3 bucket

Related videos:

- Automate Amazon EBS Snapshots with Amazon Data Lifecycle Manager
- Empty an Amazon S3 bucket using a lifecycle configuration rule

Related examples:

- Empty an Amazon S3 bucket using a lifecycle configuration rule

Cost effective resources

Using the appropriate services, resources, and configurations for your workloads is key to cost savings.Consider the following when creating cost-effective resources:

You can use AWS Solutions Architects, AWS Solutions, AWS Reference Architectures, and APN Partners to help you choose an architecture based on what you have learned.

Topics

Evaluate cost when selecting services

Best practices

COST05-BP01 Identify organization requirements for cost

Work with team members to define the balance between cost optimization and other pillars, such as performance and reliability, for this workload.

Level of risk exposed if this best practice is not established: High

Implementation guidance

In most organizations, the information technology (IT) department is comprised of multiple small teams, each with its own agenda and focus area, that reflects the specialisies and skills of its team members. You need to understand your organization's overall objectives, priorities, goals and how each department or project contributes to these objectives. Categorizing all essential resources, including personnel, equipment, technology, materials, and external services, is crucial for achieving organizational objectives and comprehensive budget planning. Adopting this systematic approach to cost identification and understanding is fundamental for establishing a realistic and robust cost plan for the organization.

When selecting services for your workload, it is key that you understand your organization priorities. Create a balance between cost optimization and other AWS Well-Architected Framework pillars, such as performance and reliability. This process should be conducted systematically and regularly to reflect changes in the organization's objectives, market conditions, and operational dynamics. A fully cost-optimized workload is the solution that is most aligned to your organization's requirements, not necessarily the lowest cost. Meet with all teams in your organization, such as product, business, technical, and finance to collect information. Evaluate the impact of tradeoffs between competing interests or alternative approaches to help make informed decisions when determining where to focus efforts or choosing a course of action.

For example, accelerating speed to market for new features may be emphasized over cost optimization, or you may choose a relational database for non-relational data to simplify the effort to migrate a system, rather than migrating to a database optimized for your data type and updating your application.

Implementation steps

- **Identify organization requirements for cost:** Meet with team members from your organization, including those in product management, application owners, development and operational teams, management, and financial roles. Prioritize the Well-Architected pillars for this workload and its components. The output should be a list of the pillars in order. You can also add a weight to each pillar to indicate how much additional focus it has, or how similar the focus is between two pillars.
- **Address the technical debt and document it:** During the workload review, address the technical debt. Document a backlog item to revisit the workload in the future, with the goal of refactoring or re-architecting to optimize it further. It's essential to clearly communicate the trade-offs that were made to other stakeholders.

Resources

Related best practices:

- REL11-BP07 Architect your product to meet availability targets and uptime service level agreements (SLAs)
- OPS01-BP06 Evaluate tradeoffs

Related documents:

- AWS Total Cost of Ownership (TCO) Calculator
- Amazon S3 storage classes
- Cloud products

COST05-BP02 Analyze all components of the workload

Verify every workload component is analyzed, regardless of current size or current costs. The review effort should reflect the potential benefit, such as current and projected costs.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Workload components, which are designed to deliver business value to the organization, may encompass various services. For each component, one might choose specific AWS Cloud services to address business needs. This selection could be influenced by factors such as familiarity with or prior experience using these services.

After identifying your organization's requirements as mentioned in COST05-BP01 Identify organization requirements for cost, perform a thorough analysis on all components in your workload. Analyze each component considering current and projected costs and sizes. Consider the cost of analysis against any potential workload savings over its lifecycle. The effort expended on the analysis of all components of this workload should correspond to the potential savings or improvements anticipated from optimization of that specific component. For example, if the cost of the proposed resource is $10 per month, and under forecasted loads would not exceed $15 per month, spending a day of effort to reduce costs by 50% (five dollars per month) could exceed the potential benefit over the life of the system. Use a faster and more efficient data-based estimation to create the best overall outcome for this component.

Workloads can change over time, and the right set of services may not be optimal if the workload architecture or usage changes. Analysis for selection of services must incorporate current and future workload states and usage levels. Implementing a service for future workload state or usage may reduce overall costs by reducing or removing the effort required to make future changes. For example, using EMR Serverless might be the appropriate choice initially. However, as consumption for that service increases, transitioning to EMR on EC2 could reduce costs for that component of the workload.

AWS Cost Explorer and the AWS Cost and Usage Reports (CUR) can analyze the cost of a proof of concept (PoC) or running environment. You can also use AWS Pricing Calculator to estimate workload costs.

Write a workflow to be followed by technical teams to review their workloads. Keep this workflow simple, but also cover all the necessary steps to make sure the teams understand each component of the workload and its pricing. Your organization can then follow and customize this workflow based on the specific needs of each team.

1. **List each service in use for your workload:** This is a good starting point. Identify all of the services currently in use and where costs are originate from.
2. **Understand how pricing works for those services:** Understand the pricing model of each service. Different AWS services have different pricing models based on factors like usage volume, data transfer, and feature-specific pricing.
3. **Focus on the services that have unexpected workload costs and that do not align with your expected usage and business outcome:** Identify outliers or services where the cost is not proportional to the value or usage with using AWS Cost Explorer or AWS Cost and Usage Reports. It's important to correlate costs with business outcomes to prioritize optimization efforts.
4. **AWS Cost Explorer, CloudWatch Logs, VPC Flow Logs, and Amazon S3 Storage Lens to understand the root cause of those high costs:** These tools are instrumental in the diagnosis of high costs. Each service offers a different lens to view and analyze usage and costs. For instance, Cost Explorer helps determine overall cost trends, CloudWatch Logs provides operational insights, VPC Flow Logs displays IP traffic, and Amazon S3 Storage Lens is useful for storage analytics.
5. **Use AWS Budgets to set budgets for certain amounts for services or accounts:** Setting budgets is a proactive way to manage costs. Use AWS Budgets to set custom budget thresholds and receive alerts when costs exceed those thresholds.

6. **Configure Amazon CloudWatch alarms to send billing and usage alerts:** Set up monitoring and alerts for cost and usage metrics. CloudWatch alarms can notify you when certain thresholds are breached, which improves intervention response time.

Facilitate notable enhancement and financial savings over time through strategic review of all workload components and irrespective of their present attributes. The effort invested in this review process should be deliberate, with careful consideration of the potential advantages that might be realized.

Implementation steps

- **List the workload components:** Build a list of your workload's components. Use this list to verify that each component was analyzed. The effort spent should reflect the criticality to the workload as defined by your organization's priorities. Group together resources functionally to improve efficiency (for example, production database storage, if there are multiple databases).
- **Prioritize the component list:** Take the component list and prioritize it in order of effort. This is typically in order of the cost of the component, from most expensive to least expensive or the criticality as defined by your organization's priorities.
- **Perform the analysis:** For each component on the list, review the options and services available, and choose the option that aligns best with your organizational priorities.

Resources

Related documents:

- AWS Pricing Calculator
- AWS Cost Explorer
- Amazon S3 storage classes
- AWS Cloud products

Related videos:

- AWS Cost Optimization Series: CloudWatch

COST05-BP03 Perform a thorough analysis of each component

Look at overall cost to the organization of each component. Calculate the total cost of ownership by factoring in cost of operations and management, especially when using managed services by cloud provider. The review effort should reflect potential benefit (for example, time spent analyzing is proportional to component cost).

Level of risk exposed if this best practice is not established: High

Implementation guidance

Consider the time savings that will allow your team to focus on retiring technical debt, innovation, value-adding features and building what differentiates the business. For example, you might need to lift and shift (also known as rehost) your databases from your on-premises environment to the cloud as rapidly as possible and optimize later. It is worth exploring the possible savings attained by using managed services on AWS that may remove or reduce license costs. Managed services on AWS remove the operational and administrative burden of maintaining a service, such as patching or upgrading the OS, and allow you to focus on innovation and business.

Since managed services operate at cloud scale, they can offer a lower cost per transaction or service. You can make potential optimizations in order to achieve some tangible benefit, without changing the core architecture of the application. For example, you may be looking to reduce the amount of time you spend managing database instances by migrating to a database-as-a-service platform like Amazon Relational Database Service (Amazon RDS) or migrating your application to a fully managed platform like AWS Elastic Beanstalk.

Usually, managed services have attributes that you can set to ensure sufficient capacity. You must set and monitor these attributes so that your excess capacity is kept to a minimum and performance is maximized. You can modify the attributes of AWS Managed Services using the AWS Management Console or AWS APIs and SDKs to align resource needs with changing demand. For example, you can increase or decrease the number of nodes on an Amazon EMR cluster (or an Amazon Redshift cluster) to scale out or in.

You can also pack multiple instances on an AWS resource to activate higher density usage. For example, you can provision multiple small databases on a single Amazon Relational Database Service (Amazon RDS) database instance. As usage grows, you can migrate one of the databases to a dedicated Amazon RDS database instance using a snapshot and restore process.

When provisioning workloads on managed services, you must understand the requirements of adjusting the service capacity. These requirements are typically time, effort, and any impact to

normal workload operation. The provisioned resource must allow time for any changes to occur, provision the required overhead to allow this. The ongoing effort required to modify services can be reduced to virtually zero by using APIs and SDKs that are integrated with system and monitoring tools, such as Amazon CloudWatch.

Amazon RDS, Amazon Redshift, and Amazon ElastiCache provide a managed database service. Amazon Athena, Amazon EMR, and Amazon OpenSearch Service provide a managed analytics service.

AMS is a service that operates AWS infrastructure on behalf of enterprise customers and partners. It provides a secure and compliant environment that you can deploy your workloads onto. AMS uses enterprise cloud operating models with automation to allow you to meet your organization requirements, move into the cloud faster, and reduce your on-going management costs.

Implementation steps

- **Perform a thorough analysis:** Using the component list, work through each component from the highest priority to the lowest priority. For the higher priority and more costly components, perform additional analysis and assess all available options and their long term impact. For lower priority components, assess if changes in usage would change the priority of the component, and then perform an analysis of appropriate effort.
- **Compare managed and unmanaged resources:** Consider the operational cost for the resources you manage and compare them with AWS managed resources. For example, review your databases running on Amazon EC2 instances and compare with Amazon RDS options (an AWS managed service) or Amazon EMR compared to running Apache Spark on Amazon EC2. When moving from a self-managed workload to a AWS fully managed workload, research your options carefully. The three most important factors to consider are the type of managed service you want to use, the process you will use to migrate your data and understand the AWS shared responsibility model.

Resources

Related documents:

- AWS Total Cost of Ownership (TCO) Calculator
- Amazon S3 storage classes
- AWS Cloud products

- AWS Shared Responsibility Model

Related videos:

- Why move to a managed database?
- What is Amazon EMR and how can I use it for processing data?

Related examples:

- Why to move to a managed database
- Consolidate data from identical SQL Server databases into a single Amazon RDS for SQL Server database using AWS DMS
- Deliver data at scale to Amazon Managed Streaming for Apache Kafka (Amazon MSK)
- Migrate an ASP.NET web application to AWS Elastic Beanstalk

COST05-BP04 Select software with cost-effective licensing

Open-source software eliminates software licensing costs, which can contribute significant costs to workloads. Where licensed software is required, avoid licenses bound to arbitrary attributes such as CPUs, look for licenses that are bound to output or outcomes. The cost of these licenses scales more closely to the benefit they provide.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

Open source originated in the context of software development to indicate that the software complies with certain free distribution criteria. Open source software is composed of source code that anyone can inspect, modify, and enhance. Based on business requirements, skill of engineers, forecasted usage, or other technology dependencies, organizations can consider using open source software on AWS to minimize their license costs. In other words, the cost of software licenses can be reduced through the use of open source software. This can have significant impact on workload costs as the size of the workload scales.

Measure the benefits of licensed software against the total cost to optimize your workload. Model any changes in licensing and how they would impact your workload costs. If a vendor changes the

cost of your database license, investigate how that impacts the overall efficiency of your workload. Consider historical pricing announcements from your vendors for trends of licensing changes across their products. Licensing costs may also scale independently of throughput or usage, such as licenses that scale by hardware (CPU bound licenses). These licenses should be avoided because costs can rapidly increase without corresponding outcomes.

For instance, operating an Amazon EC2 instance in us-east-1 with a Linux operating system allows you to cut costs by approximately 45%, compared to running another Amazon EC2 instance that runs on Windows.

The AWS Pricing Calculator offers a comprehensive way to compare the costs of various resources with different license options, such as Amazon RDS instances and different database engines. Additionally, the AWS Cost Explorer provides an invaluable perspective for the costs of existing workloads, especially those that come with different licenses. For license management, AWS License Manager offers a streamlined method to oversee and handle software licenses. Customers can deploy and operationalize their preferred open source software in the AWS Cloud.

Implementation steps

- **Analyze license options:** Review the licensing terms of available software. Look for open source versions that have the required functionality, and whether the benefits of licensed software outweigh the cost. Favorable terms align the cost of the software to the benefits it provides.
- **Analyze the software provider:** Review any historical pricing or licensing changes from the vendor. Look for any changes that do not align to outcomes, such as punitive terms for running on specific vendors hardware or platforms. Additionally, look for how they perform audits, and penalties that could be imposed.

Resources

Related documents:

- Open Source at AWS
- AWS Total Cost of Ownership (TCO) Calculator
- Amazon S3 storage classes
- Cloud products

Related examples:

- Open Source Blogs
- AWS Open Source Blogs
- Optimization and Licensing Assessment

COST05-BP05 Select components of this workload to optimize cost in line with organization priorities

Factor in cost when selecting all components for your workload. This includes using application-level and managed services or serverless, containers, or event-driven architecture to reduce overall cost. Minimize license costs by using open-source software, software that does not have license fees, or alternatives to reduce spending.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

Consider the cost of services and options when selecting all components. This includes using application level and managed services, such as Amazon Relational Database Service (Amazon RDS), Amazon DynamoDB, Amazon Simple Notification Service (Amazon SNS), and Amazon Simple Email Service (Amazon SES) to reduce overall organization cost.

Use serverless and containers for compute, such as AWS Lambda and Amazon Simple Storage Service (Amazon S3) for static websites. Containerize your application if possible and use AWS Managed Container Services such as Amazon Elastic Container Service (Amazon ECS) or Amazon Elastic Kubernetes Service (Amazon EKS).

Minimize license costs by using open-source software, or software that does not have license fees (for example, Amazon Linux for compute workloads or migrate databases to Amazon Aurora).

You can use serverless or application-level services such as Lambda, Amazon Simple Queue Service (Amazon SQS), Amazon SNS, and Amazon SES. These services remove the need for you to manage a resource and provide the function of code execution, queuing services, and message delivery. The other benefit is that they scale in performance and cost in line with usage, allowing efficient cost allocation and attribution.

Using event-driven architecture is also possible with serverless services. Event-driven architectures are push-based, so everything happens on demand as the event presents itself in the router.

This way, you're not paying for continuous polling to check for an event. This means less network bandwidth consumption, less CPU utilization, less idle fleet capacity, and fewer SSL/TLS handshakes.

For more information on serverless, see Well-Architected Serverless Application lens whitepaper.

Implementation steps

- **Select each service to optimize cost:** Using your prioritized list and analysis, select each option that provides the best match with your organizational priorities. Instead of increasing the capacity to meet the demand, consider other options which may give you better performance with lower cost. For example, if you need to review expected traffic for your databases on AWS, consider either increasing the instance size or using Amazon ElastiCache services (Redis or Memcached) to provide cached mechanisms for your databases.
- **Evaluate event-driven architecture:** Using serverless architecture also allows you to build event-driven architecture for distributed microservice-based applications, which helps you build scalable, resilient, agile and cost-effective solutions.

Resources

Related documents:

- AWS Total Cost of Ownership (TCO) Calculator
- AWS Serverless
- What is Event-Driven Architecture
- Amazon S3 storage classes
- Cloud products
- Amazon ElastiCache (Redis OSS)

Related examples:

- Getting started with event-driven architecture
- Event-driven architecture
- How Statsig runs 100x more cost-effectively using Amazon ElastiCache (Redis OSS)
- Best practices for working with AWS Lambda functions

COST05-BP06 Perform cost analysis for different usage over time

Workloads can change over time. Some services or features are more cost effective at different usage levels. By performing the analysis on each component over time and at projected usage, the workload remains cost-effective over its lifetime.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

As AWS releases new services and features, the optimal services for your workload may change. Effort required should reflect potential benefits. Workload review frequency depends on your organization requirements. If it is a workload of significant cost, implementing new services sooner will maximize cost savings, so more frequent review can be advantageous. Another initiation for review is change in usage patterns. Significant changes in usage can indicate that alternate services would be more optimal.

If you need to move data into AWS Cloud, you can select any wide variety of services AWS offers and partner tools to help you migrate your data sets, whether they are files, databases, machine images, block volumes, or even tape backups. For example, to move a large amount of data to and from AWS or process data at the edge, you can use one of the AWS purpose-built devices to cost effectively move petabytes of data offline. Another example is for higher data transfer rates, a direct connect service may be cheaper than a VPN which provides the required consistent connectivity for your business.

Based on the cost analysis for different usage over time, review your scaling activity. Analyze the result to see if the scaling policy can be tuned to add instances with multiple instance types and purchase options. Review your settings to see if the minimum can be reduced to serve user requests but with a smaller fleet size, and add more resources to meet the expected high demand.

Perform cost analysis for different usage over time by discussing with stakeholders in your organization and use AWS Cost Explorer's forecast feature to predict the potential impact of service changes. Monitor usage level launches using AWS Budgets, CloudWatch billing alarms and AWS Cost Anomaly Detection to identify and implement the most cost-effective services sooner.

Implementation steps

- **Define predicted usage patterns:** Working with your organization, such as marketing and product owners, document what the expected and predicted usage patterns will be for the

workload. Discuss with business stakeholders about both historical and forecasted cost and usage increases and make sure increases align with business requirements. Identify calendar days, weeks, or months where you expect more users to use your AWS resources, which indicate that you should increase the capacity of the existing resources or adopt additional services to reduce the cost and increase performance.

- **Perform cost analysis at predicted usage:** Using the usage patterns defined, perform analysis at each of these points. The analysis effort should reflect the potential outcome. For example, if the change in usage is large, a thorough analysis should be performed to verify any costs and changes. In other words, when cost increases, usage should increase for business as well.

Resources

Related documents:

- AWS Total Cost of Ownership (TCO) Calculator
- Amazon S3 storage classes
- Cloud products
- Amazon EC2 Auto Scaling
- Cloud Data Migration
- AWS Snow Family

Related videos:

- AWS OpsHub for Snow Family

Select the correct resource type, size, and number

By selecting the best resource type, size, and number of resources, you meet the technical requirements with the lowest cost resource. Right-sizing activities takes into account all of the resources of a workload, all of the attributes of each individual resource, and the effort involved in the right-sizing operation. Right-sizing can be an iterative process, initiated by changes in usage patterns and external factors, such as AWS price drops or new AWS resource types. Right-sizing can also be one-off if the cost of the effort to right-size, outweighs the potential savings over the life of the workload.

In AWS, there are a number of different approaches:

Best practices

- COST06-BP01 Perform cost modeling
- COST06-BP02 Select resource type, size, and number based on data
- COST06-BP03 Select resource type, size, and number automatically based on metrics
- COST06-BP04 Consider using shared resources

COST06-BP01 Perform cost modeling

Identify organization requirements (such as business needs and existing commitments) and perform cost modeling (overall costs) of the workload and each of its components. Perform benchmark activities for the workload under different predicted loads and compare the costs. The modeling effort should reflect the potential benefit. For example, time spent is proportional to component cost.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Perform cost modelling for your workload and each of its components to understand the balance between resources, and find the correct size for each resource in the workload, given a specific level of performance. Understanding cost considerations can inform your organizational business case and decision-making process when evaluating the value realization outcomes for planned workload deployment.

Perform benchmark activities for the workload under different predicted loads and compare the costs. The modelling effort should reflect potential benefit; for example, time spent is proportional to component cost or predicted saving. For best practices, refer to the Review section of the Performance Efficiency Pillar of the AWS Well-Architected Framework.

As an example, to create cost modeling for a workload consisting of compute resources, AWS Compute Optimizer can assist with cost modelling for running workloads. It provides right-sizing recommendations for compute resources based on historical usage. Make sure CloudWatch Agents are deployed to the Amazon EC2 instances to collect memory metrics which help you with more accurate recommendations within AWS Compute Optimizer. This is the ideal data source for compute resources because it is a free service that uses machine learning to make multiple recommendations depending on levels of risk.

There are multiple services you can use with custom logs as data sources for rightsizing operations for other services and workload components, such as AWS Trusted Advisor, Amazon CloudWatch and Amazon CloudWatch Logs. AWS Trusted Advisor checks resources and flags resources with low utilization which can help you right size your resources and create cost modelling.

The following are recommendations for cost modelling data and metrics:

- The monitoring must accurately reflect the user experience. Select the correct granularity for the time period and thoughtfully choose the maximum or 99th percentile instead of the average.
- Select the correct granularity for the time period of analysis that is required to cover any workload cycles. For example, if a two-week analysis is performed, you might be overlooking a monthly cycle of high utilization, which could lead to under-provisioning.
- Choose the right AWS services for your planned workload by considering your existing commitments, selected pricing models for other workloads, and ability to innovate faster and focus on your core business value.

Implementation steps

- **Perform cost modeling for resources:** Deploy the workload or a proof of concept into a separate account with the specific resource types and sizes to test. Run the workload with the test data and record the output results, along with the cost data for the time the test was run. Afterwards, redeploy the workload or change the resource types and sizes and run the test again. Include license fees of any products you may use with these resources and estimated operations (labor or engineer) costs for deploying and managing these resources while creating cost modeling. Consider cost modeling for a period (hourly, daily, monthly, yearly or three years).

Resources

Related documents:

- AWS Auto Scaling
- Identifying Opportunities to Right Size
- Amazon CloudWatch features
- Cost Optimization: Amazon EC2 Right Sizing
- AWS Compute Optimizer
- AWS Pricing Calculator

Related examples:

- Perform a Data-Driven Cost Modelling
- Estimate the cost of planned AWS resource configurations
- Choose the right AWS tools

COST06-BP02 Select resource type, size, and number based on data

Select resource size or type based on data about the workload and resource characteristics. For example, compute, memory, throughput, or write intensive. This selection is typically made using a previous (on-premises) version of the workload, using documentation, or using other sources of information about the workload.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

Amazon EC2 provides a wide selection of instance types with different levels of CPU, memory, storage, and networking capacity to fit different use cases. These instance types feature different blends of CPU, memory, storage, and networking capabilities, giving you versatility when selecting the right resource combination for your projects. Every instance type comes in multiple sizes, so that you can adjust your resources based on your workload's demands. To determine which instance type you need, gather details about the system requirements of the application or software that you plan to run on your instance. These details should include the following:

- Operating system
- Number of CPU cores
- GPU cores
- Amount of system memory (RAM)
- Storage type and space
- Network bandwidth requirement

Identify the purpose of compute requirements and which instance is needed, and then explore the various Amazon EC2 instance families. Amazon offers the following instance type families:

- General Purpose

- Compute Optimized
- Memory Optimized
- Storage Optimized
- Accelerated Computing
- HPC Optimized

For a deeper understanding of the specific purposes and use cases that a particular Amazon EC2 instance family can fulfill, see AWS Instance types.

System requirements gathering is critical for you to select the specific instance family and instance type that best serves your needs. Instance type names are comprised of the family name and the instance size. For example, the t2.micro instance is from the T2 family and is micro-sized.

Select resource size or type based on workload and resource characteristics (for example, compute, memory, throughput, or write intensive). This selection is typically made using cost modelling, a previous version of the workload (such as an on-premises version), using documentation, or using other sources of information about the workload (whitepapers or published solutions). Using AWS pricing calculators or cost management tools can assist in making informed decisions about instance types, sizes, and configurations.

Implementation steps

- **Select resources based on data:** Use your cost modeling data to select the anticipated workload usage level, and choose the specified resource type and size. Relying on the cost modeling data, determine the number of virtual CPUs, total memory (GiB), the local instance store volume (GB), Amazon EBS volumes, and the network performance level, taking into account the data transfer rate required for the instance. Always make selections based on detailed analysis and accurate data to optimize performance while managing costs effectively.

Resources

Related documents:

- AWS Instance types
- AWS Auto Scaling
- Amazon CloudWatch features
- Cost Optimization: EC2 Right Sizing

Related videos:

- Selecting the right Amazon EC2 instance for your workloads
- Right size your service

Related examples:

- It just got easier to discover and compare Amazon EC2 instance types

COST06-BP03 Select resource type, size, and number automatically based on metrics

Use metrics from the currently running workload to select the right size and type to optimize for cost. Appropriately provision throughput, sizing, and storage for compute, storage, data, and networking services. This can be done with a feedback loop such as automatic scaling or by custom code in the workload.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

Create a feedback loop within the workload that uses active metrics from the running workload to make changes to that workload. You can use a managed service, such as AWS Auto Scaling, which you configure to perform the right sizing operations for you. AWS also provides APIs, SDKs, and features that allow resources to be modified with minimal effort. You can program a workload to stop-and-start an Amazon EC2 instance to allow a change of instance size or instance type. This provides the benefits of right-sizing while removing almost all the operational cost required to make the change.

Some AWS services have built in automatic type or size selection, such as Amazon Simple Storage Service Intelligent-Tiering. Amazon S3 Intelligent-Tiering automatically moves your data between two access tiers, frequent access and infrequent access, based on your usage patterns.

Implementation steps

- **Increase your observability by configuring workload metrics:** Capture key metrics for the workload. These metrics provide an indication of the customer experience, such as workload output, and align to the differences between resource types and sizes, such as CPU and memory

usage. For compute resource, analyze performance data to right size your Amazon EC2 instances. Identify idle instances and ones that are underutilized. Key metrics to look for are CPU usage and memory utilization (for example, 40% CPU utilization at 90% of the time as explained in Rightsizing with AWS Compute Optimizer and Memory Utilization Enabled). Identify instances with a maximum CPU usage and memory utilization of less than 40% over a four-week period. These are the instances to right size to reduce costs. For storage resources such as Amazon S3, you can use Amazon S3 Storage Lens, which allows you to see 28 metrics across various categories at the bucket level, and 14 days of historical data in the dashboard by default. You can filter your Amazon S3 Storage Lens dashboard by summary and cost optimization or events to analyze specific metrics.

- **View rightsizing recommendations:** Use the rightsizing recommendations in AWS Compute Optimizer and the Amazon EC2 rightsizing tool in the Cost Management console, or review AWS Trusted Advisor right-sizing your resources to make adjustments on your workload. It is important to use the right tools when right-sizing different resources and follow right-sizing guidelines whether it is an Amazon EC2 instance, AWS storage classes, or Amazon RDS instance types. For storage resources, you can use Amazon S3 Storage Lens, which gives you visibility into object storage usage, activity trends, and makes actionable recommendations to optimize costs and apply data protection best practices. Using the contextual recommendations that Amazon S3 Storage Lens derives from analysis of metrics across your organization, you can take immediate steps to optimize your storage.
- **Select resource type and size automatically based on metrics:** Using the workload metrics, manually or automatically select your workload resources. For compute resources, configuring AWS Auto Scaling or implementing code within your application can reduce the effort required if frequent changes are needed, and it can potentially implement changes sooner than a manual process. You can launch and automatically scale a fleet of On-Demand Instances and Spot Instances within a single Auto Scaling group. In addition to receiving discounts for using Spot Instances, you can use Reserved Instances or a Savings Plan to receive discounted rates of the regular On-Demand Instance pricing. All of these factors combined help you optimize your cost savings for Amazon EC2 instances and determine the desired scale and performance for your application. You can also use an attribute-based instance type selection (ABS) strategy in Auto Scaling Groups (ASG), which lets you express your instance requirements as a set of attributes, such as vCPU, memory, and storage. You can automatically use newer generation instance types when they are released and access a broader range of capacity with Amazon EC2 Spot Instances. Amazon EC2 Fleet and Amazon EC2 Auto Scaling select and launch instances that fit the specified attributes, removing the need to manually pick instance types. For storage resources, you can use the Amazon S3 Intelligent Tiering and Amazon EFS Infrequent Access

features, which allow you to select storage classes automatically that deliver automatic storage cost savings when data access patterns change, without performance impact or operational overhead.

Resources

Related documents:

- AWS Auto Scaling
- AWS Right-Sizing
- AWS Compute Optimizer
- Amazon CloudWatch features
- CloudWatch Getting Set Up
- CloudWatch Publishing Custom Metrics
- Getting Started with Amazon EC2 Auto Scaling
- Amazon S3 Storage Lens
- Amazon S3 Intelligent-Tiering
- Amazon EFS Infrequent Access
- Launch an Amazon EC2 Instance Using the SDK

Related videos:

- Right Size Your Services

Related examples:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive scaling with Amazon EC2 Auto Scaling
- Optimize Costs and Gain Visibility into Usage with Amazon S3 Storage Lens

COST06-BP04 Consider using shared resources

For already-deployed services at the organization level for multiple business units, consider using shared resources to increase utilization and reduce total cost of ownership (TCO). Using shared resources can be a cost-effective option to centralize the management and costs by using existing solutions, sharing components, or both. Manage common functions like monitoring, backups, and connectivity either within an account boundary or in a dedicated account. You can also reduce cost by implementing standardization, reducing duplication, and reducing complexity.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

Where multiple workloads cause the same function, use existing solutions and shared components to improve management and optimize costs. Consider using existing resources (especially shared ones), such as non-production database servers or directory services, to mitigate cloud costs by following security best practices and organizational regulations. For optimal value realization and efficiency, it is crucial to allocate costs back (using showback and chargeback) to the pertinent areas of the business driving consumption.

Showback refers to reports that break down cloud costs into attributable categories, such as consumers, business units, general ledger accounts, or other responsible entities. The goal of showback is to show teams, business units, or individuals the cost of their consumed cloud resources.

Chargeback means to allocate central service spend to cost units based on a strategy suitable for a specific financial management process. For customers, chargeback charges the cost incurred from one shared services account to different financial cost categories suitable for a customer reporting process. By establishing chargeback mechanisms, you can report costs incurred by different business units, products, and teams.

Workloads can be categorized as critical and non-critical. Based on this classification, use shared resources with general configurations for less critical workloads. To further optimize costs, reserve dedicated servers solely for critical workloads. Share resources or provision them across several accounts to manage them efficiently. Even with distinct development, testing, and production environments, secure sharing is feasible and does not compromise organizational structure.

To improve your understanding and optimize cost and usage for containerized applications, use split cost allocation data which helps you allocate costs to individual business entities based on

how the application consumes shared compute and memory resources. Split cost allocation data helps you achieve task-level showback and chargeback in container workloads running on Amazon Elastic Container Service (Amazon ECS) or Amazon Elastic Kubernetes Service (Amazon EKS).

For distributed architectures, build a shared services VPC, which provides centralized access to shared services required by workloads in each of the VPCs. These shared services can include resources such as directory services or VPC endpoints. To reduce administrative overhead and cost, share resources from a central location instead of building them in each VPC.

When you use shared resources, you can save on operational costs, maximize resource utilization, and improve consistency. In a multi-account design, you can host some AWS services centrally and access them using several applications and accounts in a hub to save cost. You can use AWS Resource Access Manager (AWS RAM) to share other common resources, such as VPC subnets and AWS Transit Gateway attachments, AWS Network Firewall, or Amazon SageMaker AI pipelines. In a multi-account environment, use AWS RAM to create a resource once and share it with other accounts.

Organizations should tag shared costs effectively and verify that they do not have a significant portion of their costs untagged or unallocated. If you do not allocate shared costs effectively and no one takes accountability for shared costs management, shared cloud costs can spiral. You should know where you have incurred costs at the resource, workload, team, or organization level, as this knowledge enhances your understanding of the value delivered at the applicable level when compared to the business outcomes achieved. Ultimately, organizations benefit from cost savings as a result of sharing cloud infrastructure. Encourage cost allocation on shared cloud resources to optimize cloud spend.

Implementation steps

- **Evaluate existing resources:** Review existing workloads that use similar services for your workload. Depending on the workload's components, consider existing platforms if business logic or technical requirement allow.
- **Use resource sharing in AWS RAM and restrict accordingly:** Use AWS RAM to share resources with other AWS accounts within your organization. When you share resources, you don't need to duplicate resources in multiple accounts, which minimizes the operational burden of resource maintenance. This process also helps you securely share the resources that you have created with roles and users in your account, as well as with other AWS accounts.
- **Tag resources:** Tag resources that are candidates for cost reporting and categorize them within cost categories. Activate these cost related resource tags for cost allocation to provide visibility

of AWS resources usage. Focus on creating an appropriate level of granularity with respect to cost and usage visibility, and influence cloud consumption behaviors through cost allocation reporting and KPI tracking.

Resources

Related best practices:

- SEC03-BP08 Share resources securely within your organization

Related documents:

- What is AWS Resource Access Manager?
- AWS services that you can use with AWS Organizations
- Shareable AWS resources
- AWS Cost and Usage (CUR) Queries

Related videos:

- AWS Resource Access Manager - granular access control with managed permissions
- How to design your AWS cost allocation strategy
- AWS Cost Categories

Related examples:

- How-to chargeback shared services: An AWS Transit Gateway example
- How to build a chargeback/showback model for Savings Plans using the CUR
- Using VPC Sharing for a Cost-Effective Multi-Account Microservice Architecture
- Improve cost visibility of Amazon EKS with AWS Split Cost Allocation Data
- Improve cost visibility of Amazon ECS and AWS Batch with AWS Split Cost Allocation Data

Select the best pricing model

Perform workload cost modeling: Consider the requirements of the workload components and understand the potential pricing models. Define the availability requirement of the component.

Determine if there are multiple independent resources that perform the function in the workload, and what the workload requirements are over time. Compare the cost of the resources using the default On-Demand pricing model and other applicable models. Factor in any potential changes in resources or workload components.

Perform regular account level analysis: Performing regular cost modeling ensures that opportunities to optimize across multiple workloads can be implemented. For example, if multiple workloads use On-Demand, at an aggregate level, the risk of change is lower, and implementing a commitment-based discount will achieve a lower overall cost. It is recommended to perform analysis in regular cycles of two weeks to one month. This analysis allows you to make small adjustment purchases, so the coverage of your pricing models continues to evolve with your changing workloads and their components.

Use the AWS Cost Explorer recommendations tool to find opportunities for commitment discounts.

To find opportunities for Spot workloads, use an hourly view of your overall usage, and look for regular periods of changing usage or elasticity.

Pricing models: AWS has multiple pricing models that allow you to pay for your resources in the most cost-effective way that suits your organization's needs. The following section describes each purchasing model:

- On-Demand Instances
- Spot Instances
- Commitment discounts - Savings Plans
- Commitment discounts - Reserved Instances/Capacity
- Geographic selection
- Third-party agreements and pricing

On-Demand Instances: This is the default, pay as you go pricing model. When you use resources (for example, EC2 instances or services such as DynamoDB on demand) you pay a flat rate, and you have no long-term commitments. You can increase or decrease the capacity of your resources or services based on the demands of your application. On-Demand has an hourly rate, but depending on the service, can be billed in increments of one second (for example Amazon RDS, or Linux EC2 instances). On demand is recommended for applications with short-term workloads (for example, a four-month project), that spike periodically, or unpredictable workloads that can't be interrupted. On demand is also suitable for workloads, such as pre-production environments, which require

uninterrupted runtimes, but do not run long enough for a commitment discount (Savings Plans or Reserved Instances).

Spot Instances: A Spot Instance is spare Amazon EC2 compute capacity available at discounts of up to 90% off On-Demand prices with no long-term commitment required. With Spot Instances, you can significantly reduce the cost of running your applications or scale your application's compute capacity for the same budget. Unlike On-Demand, Spot Instances can be interrupted with a 2-minute warning if Amazon EC2 needs the capacity back, or the Spot Instance price exceeds your configured price. On average, Spot Instances are interrupted less than 5% of the time.

Spot Instances are ideal when there is a queue or buffer in place, or where there are multiple resources working independently to process the requests (for example, Hadoop data processing). Typically these workloads are fault-tolerant, stateless, and flexible, such as batch processing, big data and analytics, containerized environments, and high performance computing (HPC). Non-critical workloads such as test and development environments are also candidates for Spot.

Spot Instances are also integrated into multiple AWS services, such as Amazon EC2 Auto Scaling groups, Amazon EMR, Amazon Elastic Container Service (Amazon ECS), and AWS Batch.

When a Spot Instance needs to be reclaimed, Amazon EC2 sends a two-minute warning via a Spot Instance interruption notice delivered through CloudWatch Events, as well as in the instance metadata. During that two-minute period, your application can use the time to save its state, drain running containers, upload final log files, or remove itself from a load balancer. At the end of the two minutes, you have the option to hibernate, stop, or terminate the Spot Instance.

Consider the following best practices when adopting Spot Instances in your workloads:

- **Be flexible across as many instance types as possible:** Be flexible in both the family and size of the instance type, to improve the likelihood of fulfilling your target capacity requirements, obtain the lowest possible cost, and minimize the impact of interruptions.
- **Be flexible about where your workload will run:** Available capacity can vary by Availability Zone. This improves the likelihood of fulfilling your target capacity by tapping into multiple spare capacity pools, and provides the lowest possible cost.
- **Design for continuity:** Design your workloads for statelessness and fault-tolerance, so that if some of your EC2 capacity gets interrupted, it will not have impact on the availability or performance of the workload.
- We recommend using Spot Instances in combination with On-Demand and Savings Plans/ Reserved Instances to maximize workload cost optimization with performance.

Commitment discounts – Savings Plans: AWS provides a number of ways for you to reduce your costs by reserving or committing to use a certain amount of resources, and receiving a discounted rate for your resources. A Savings Plan allows you to make an hourly spend commitment for one or three years, and receive discounted pricing across your resources. Savings Plans provide discounts for AWS Compute services such as Amazon EC2, AWS Fargate, and AWS Lambda. When you make the commitment, you pay that commitment amount every hour, and it is subtracted from your On-Demand usage at the discount rate. For example, you commit to $50 an hour, and have $150 an hour of On-Demand usage. Considering the Savings Plans pricing, your specific usage has a discount rate of 50%. So, your $50 commitment covers $100 of On-Demand usage. You will pay $50 (commitment) and $50 of remaining On-Demand usage.

Compute Savings Plans are the most flexible and provide a discount of up to 66%. They automatically apply across Availability Zones, instance size, instance family, operating system, tenancy, Region, and compute service.

Instance Savings Plans have less flexibility but provide a higher discount rate (up to 72%). They automatically apply across Availability Zones, instance size, operating system, and tenancy.

There are three payment options:

- **No upfront payment:** There is no upfront payment; you then pay a reduced hourly rate each month for the total hours in the month.
- **Partial upfront payment:** Provides a higher discount rate than No upfront. Part of the usage is paid up front; you then pay a smaller reduced hourly rate each month for the total hours in the month.
- **All upfront payment:** Usage for the entire period is paid up front, and no other costs are incurred for the remainder of the term for usage that is covered by the commitment.

You can apply any combination of these three purchasing options across your workloads.

Savings plans apply first to the usage in the account they are purchased in, from the highest discount percentage to the lowest, then they apply to the consolidated usage across all other accounts, from the highest discount percentage to the lowest.

It is recommended to purchase all Savings Plans in an account with no usage or resources, such as the management account. This ensures that the Savings Plan applies to the highest discount rates across all of your usage, maximizing the discount amount.

Workloads and usage typically change over time. It is recommended to continually purchase small amounts of Savings Plans commitment over time. This ensures that you maintain high levels of coverage to maximize your discounts, and your plans closely match your workload and organization requirements at all times.

Do not set a target coverage in your accounts, due to the variability of discount that is possible. Low coverage does not necessarily indicate high potential savings. You may have a low coverage in your account, but if your usage is made up of small instances, with a licensed operating system, the potential saving could be as low as a few percent. Instead, track and monitor the potential savings available in the Savings Plan recommendation tool. Frequently review the Savings Plans recommendations in Cost Explorer (perform regular analysis) and continue to purchase commitments until the estimated savings are below the required discount for the organization. For example, track and monitor that your potential discounts remained below 20%, if it goes above that a purchase must be made.

Monitor the utilization and coverage, but only to detect changes. Do not aim for a specific utilization percent, or coverage percent, as this does not necessarily scale with savings. Ensure that a purchase of Savings Plans results in an increase in coverage, and if there are decreases in coverage or utilization ensure they are quantified and known. For example, you migrate a workload resource to a newer instance type, which reduces utilization of an existing plan, but the performance benefit outweighs the saving reduction.

Commitment discounts – Reserved Instances/Commitment: Similar to Savings Plans, Reserved Instances (RI) offer discounts up to 72% for a commitment to running a minimum amount of resources. Reserved Instances are available for Amazon RDS, Amazon OpenSearch Service, Amazon ElastiCache, Amazon Redshift, and DynamoDB. Amazon CloudFront and AWS Elemental MediaConvert also provide discounts when you make minimum usage commitments. Reserved Instances are currently available for Amazon EC2, however Savings Plans offer the same discount levels with increased flexibility and no management overhead.

Reserved Instances offer the same pricing options of no upfront, partial upfront, and all upfront, and the same terms of one or three years.

Reserved Instances can be purchased in a Region or a specific Availability Zone. They provide a capacity reservation when purchased in an Availability Zone.

Amazon EC2 features convertible RIs, however, Savings Plans should be used for all EC2 instances due to increased flexibility and reduced operational costs.

The same process and metrics should be used to track and make purchases of Reserved Instances. It is recommended to not track coverage of RIs across your accounts. It is also recommended that utilization percentage is not monitored or tracked, instead view the utilization report in Cost Explorer, and use net savings column in the table. If the net savings is a significantly large negative amount, you must take action to remediate the unused RI.

EC2 Fleet: EC2 Fleet is a feature that allows you to define a target compute capacity, and then specify the instance types and the balance of On-Demand and Spot Instances for the fleet. EC2 Fleet will automatically launch the lowest price combination of resources to meet the defined capacity.

Geographic selection: When you architect your solutions, a best practice is to seek to place computing resources closer to users to provide lower latency and strong data sovereignty. For global audiences, you should use multiple locations to meet these needs. You should select the geographic location that minimizes your costs.

The AWS Cloud infrastructure is built around Regions and Availability Zones. A Region is a physical location in the world where we have multiple Availability Zones. Availability Zones consist of one or more discrete data centers, each with redundant power, networking, and connectivity, housed in separate facilities.

Each AWS Region operates within local market conditions, and resource pricing is different in each Region. Choose a specific Region to operate a component of or your entire solution so that you can run at the lowest possible price globally. You can use the AWS Simple Monthly Calculator to estimate the costs of your workload in various Regions.

Third-party agreements and pricing: When you use third-party solutions or services in the cloud, it is important that the pricing structures are aligned to Cost Optimization outcomes. Pricing should scale with the outcomes and value it provides. An example of this is software that takes a percentage of savings it provides, the more you save (outcome) the more it charges. Agreements that scale with your bill are typically not aligned to Cost Optimization, unless they provide outcomes for every part of your specific bill. For example, a solution that provides recommendations for Amazon EC2 and charges a percentage of your entire bill will increase if you use other services for which it provides no benefit. Another example is a managed service that is charged at a percentage of the cost of resources that are managed. A larger instance size may not necessarily require more management effort, but will be charged more. Ensure that these service pricing arrangements include a cost optimization program or features in their service to drive efficiency.

Best practices

- COST07-BP01 Perform pricing model analysis
- COST07-BP02 Choose Regions based on cost
- COST07-BP03 Select third-party agreements with cost-efficient terms
- COST07-BP04 Implement pricing models for all components of this workload
- COST07-BP05 Perform pricing model analysis at the management account level

COST07-BP01 Perform pricing model analysis

Analyze each component of the workload. Determine if the component and resources will be running for extended periods (for commitment discounts) or dynamic and short-running (for spot or on-demand). Perform an analysis on the workload using the recommendations in cost management tools and apply business rules to those recommendations to achieve high returns.

Level of risk exposed if this best practice is not established: High

Implementation guidance

AWS has multiple pricing models that allow you to pay for your resources in the most cost-effective way that suits your organization's needs and depending on product. Work with your teams to determine the most appropriate pricing model. Often your pricing model consists of a combination of multiple options, as determined by your availability

On-Demand Instances allow you pay for compute or database capacity by the hour or by the second (60 seconds minimum) depending on which instances you run, without long-term commitments or upfront payments.

Savings Plans are a flexible pricing model that offers low prices on Amazon EC2, Lambda, and AWS Fargate usage, in exchange for a commitment to a consistent amount of usage (measured in dollars per hour) over one year or three years terms.

Spot Instances are an Amazon EC2 pricing mechanism that allows you request spare compute capacity at discounted hourly rate (up to 90% off the on-demand price) without upfront commitment.

Reserved Instances allow you up to 75 percent discount by prepaying for capacity. For more details, see Optimizing costs with reservations.

You might choose to include a Savings Plan for the resources associated with the production, quality, and development environments. Alternatively, because sandbox resources are only powered on when needed, you might choose an on-demand model for the resources in that environment. Use Amazon Spot Instances to reduce Amazon EC2 costs or use Compute Savings Plans to reduce Amazon EC2, Fargate, and Lambda cost. The AWS Cost Explorer recommendations tool provides opportunities for commitment discounts with Saving plans.

If you have been purchasing Reserved Instances for Amazon EC2 in the past or have established cost allocation practices inside your organization, you can continue using Amazon EC2 Reserved Instances for the time being. However, we recommend working on a strategy to use Savings Plans in the future as a more flexible cost savings mechanism. You can refresh Savings Plans (SP) Recommendations in AWS Cost Management to generate new Savings Plans Recommendations at any time. Use Reserved Instances (RI) to reduce Amazon RDS, Amazon Redshift, Amazon ElastiCache, and Amazon OpenSearch Service costs. Saving Plans and Reserved Instances are available in three options: all upfront, partial upfront and no upfront payments. Use the recommendations provided in AWS Cost Explorer RI and SP purchase recommendations.

To find opportunities for Spot workloads, use an hourly view of your overall usage, and look for regular periods of changing usage or elasticity. You can use Spot Instances for various fault-tolerant and flexible applications. Examples include stateless web servers, API endpoints, big data and analytics applications, containerized workloads, CI/CD, and other flexible workloads.

Analyze your Amazon EC2 and Amazon RDS instances whether they can be turned off when you don’t use (after hours and weekends). This approach will allow you to reduce costs by 70% or more compared to using them 24/7. If you have Amazon Redshift clusters that only need to be available at specific times, you can pause the cluster and later resume it. When the Amazon Redshift cluster or Amazon EC2 and Amazon RDS Instance is stopped, the compute billing halts and only the storage charge applies.

Note that On-Demand Capacity reservations (ODCR) are not a pricing discount. Capacity Reservations are charged at the equivalent On-Demand rate, whether you run instances in reserved capacity or not. They should be considered when you need to provide enough capacity for the resources you plan to run. ODCRs don't have to be tied to long-term commitments, as they can be cancelled when you no longer need them, but they can also benefit from the discounts that Savings Plans or Reserved Instances provide.

Implementation steps

- **Analyze workload elasticity:** Using the hourly granularity in Cost Explorer or a custom dashboard, analyze your workload's elasticity. Look for regular changes in the number of instances that are running. Short duration instances are candidates for Spot Instances or Spot Fleet.
  - Well-Architected Lab: Cost Explorer
  - Well-Architected Lab: Cost Visualization
- **Review existing pricing contracts:** Review current contracts or commitments for long term needs. Analyze what you currently have and how much those commitments are in use. Leverage pre-existing contractual discounts or enterprise agreements. Enterprise Agreements give customers the option to tailor agreements that best suit their needs. For long term commitments, consider reserved pricing discounts, Reserved Instances or Savings Plans for the specific instance type, instance family, AWS Region, and Availability Zones.
- **Perform a commitment discount analysis:** Using Cost Explorer in your account, review the Savings Plans and Reserved Instance recommendations. To verify that you implement the correct recommendations with the required discounts and risk, follow the Well-Architected labs.

Resources

Related documents:

- Accessing Reserved Instance recommendations
- Instance purchasing options
- AWS Enterprise

Related videos:

- Save up to 90% and run production workloads on Spot

Related examples:

- Well-Architected Lab: Cost Explorer
- Well-Architected Lab: Cost Visualization
- Well-Architected Lab: Pricing Models

COST07-BP02 Choose Regions based on cost

Resource pricing may be different in each Region. Identify Regional cost differences and only deploy in Regions with higher costs to meet latency, data residency and data sovereignty requirements. Factoring in Region cost helps you pay the lowest overall price for this workload.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

The AWS Cloud Infrastructure is global, hosted in multiple locations world-wide, and built around AWS Regions, Availability Zones, Local Zones, AWS Outposts, and Wavelength Zones. A Region is a physical location in the world and each Region is a separate geographic area where AWS has multiple Availability Zones. Availability Zones which are multiple isolated locations within each Region consist of one or more discrete data centers, each with redundant power, networking, and connectivity.

Each AWS Region operates within local market conditions, and resource pricing is different in each Region due to differences in the cost of land, fiber, electricity, and taxes, for example. Choose a specific Region to operate a component of or your entire solution so that you can run at the lowest possible price globally. Use AWS Calculator to estimate the costs of your workload in various Regions by searching services by location type (Region, wave length zone and local zone) and Region.

When you architect your solutions, a best practice is to seek to place computing resources closer to users to provide lower latency and strong data sovereignty. Select the geographic location based on your business, data privacy, performance, and security requirements. For applications with global end users, use multiple locations.

Use Regions that provide lower prices for AWS services to deploy your workloads if you have no obligations in data privacy, security and business requirements. For example, if your default Region is Asia Pacific (Sydney) (`ap-southwest-2`), and if there are no restrictions (data privacy, security, for example) to use other Regions, deploying non-critical (development and test) Amazon EC2 instances in US East (N. Virginia) (`us-east-1`) will cost you less.

	Compliance	*Latency*	*Cost*	*Services / Features*
Region 1	✓	15 ms	$$	✓
Region 2	✓	20 ms	$$$	X
Region 3	✓	80 ms	$	✓
Region 4	✓	15 ms	$$	✓
Region 5	✓	20 ms	$$$	X
Region 6	✓	15 ms	$	✓
Region 7	✓	80 ms	$	✓
Region 8	✓	15 ms	$	X

Region feature matrix table

The preceding matrix table shows us that Region 6 is the best option for this given scenario because latency is low compared to other Regions, service is available, and it is the least expensive Region.

Implementation steps

- **Review AWS Region pricing:** Analyze the workload costs in the current Region. Starting with the highest costs by service and usage type, calculate the costs in other Regions that are available. If the forecasted saving outweighs the cost of moving the component or workload, migrate to the new Region.
- **Review requirements for multi-Region deployments:** Analyze your business requirements and obligations (data privacy, security, or performance) to find out if there are any restrictions for you to not to use multiple Regions. If there are no obligations to restrict you to use single Region, then use multiple Regions.
- **Analyze required data transfer:** Consider data transfer costs when selecting Regions. Keep your data close to your customer and close to the resources. Select less costly AWS Regions where data flows and where there is minimal data transfer. Depending on your business requirements for data transfer, you can use Amazon CloudFront, AWS PrivateLink, AWS Direct Connect, and AWS Virtual Private Network to reduce your networking costs, improve performance, and enhance security.

Resources

Related documents:

- Accessing Reserved Instance recommendations
- Amazon EC2 pricing
- Instance purchasing options
- Region Table

Related videos:

- Save up to 90% and run production workloads on Spot

Related examples:

- Overview of Data Transfer Costs for Common Architectures
- Cost Considerations for Global Deployments
- What to Consider when Selecting a Region for your Workloads

COST07-BP03 Select third-party agreements with cost-efficient terms

Cost efficient agreements and terms ensure the cost of these services scales with the benefits they provide. Select agreements and pricing that scale when they provide additional benefits to your organization.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

There are multiple products on the market that can help you manage costs in your cloud environments. They may have some differences in terms of features that depend on customer requirements, such as some focusing on cost governance or cost visibility and others on cost optimization. One key factor for effective cost optimization and governance is using the right tool with necessary features and the right pricing model. These products have different pricing models. Some charge you a certain percentage of your monthly bill, while others charge a percentage of your realized savings. Ideally, you should pay only for what you need.

When you use third-party solutions or services in the cloud, it's important that the pricing structures are aligned to your desired outcomes. Pricing should scale with the outcomes and value it provides. For example, in software that takes a percentage of savings it provides, the more you save (outcome), the more it charges. License agreements where you pay more as your expenses increase might not always be in your best interest for optimizing costs. However, if the vendor offers clear benefits for all parts of your bill, this scaling fee might be justified.

For example, a solution that provides recommendations for Amazon EC2 and charges a percentage of your entire bill can become more expensive if you use other services that provide no benefit. Another example is a managed service that is charged at a percentage of the cost of managed resources. A larger instance size may not necessarily require more management effort, but can be charged more. Verify that these service pricing arrangements include a cost optimization program or features in their service to drive efficiency.

Customers may find these products on the market more advanced or easier to use. You need to consider the cost of these products and think about potential cost optimization outcomes in the long term.

Implementation steps

- **Analyze third-party agreements and terms:** Review the pricing in third party agreements. Perform modeling for different levels of your usage, and factor in new costs such as new service usage, or increases in current services due to workload growth. Decide if the additional costs provide the required benefits to your business.

Resources

Related documents:

- Accessing Reserved Instance recommendations
- Instance purchasing options

Related videos:

- Save up to 90% and run production workloads on Spot

COST07-BP04 Implement pricing models for all components of this workload

Permanently running resources should utilize reserved capacity such as Savings Plans or Reserved Instances. Short-term capacity is configured to use Spot Instances, or Spot Fleet. On-Demand Instances are only used for short-term workloads that cannot be interrupted and do not run long enough for reserved capacity, between 25% to 75% of the period, depending on the resource type.

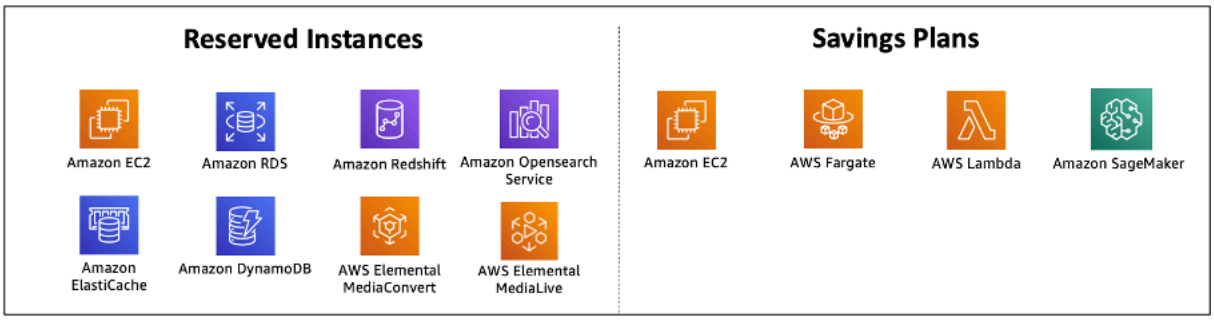
Level of risk exposed if this best practice is not established: Low

Implementation guidance

To improve cost efficiency, AWS provides multiple commitment recommendations based on your past usage. You can use these recommendations to understand what you can save, and how the commitment will be used. You can use these services as On-Demand, Spot, or make a commitment for a certain period of time and reduce your on-demand costs with Reserved Instances (RIs) and Savings Plans (SPs). You need to understand not only each workload components and multiple AWS services, but also commitment discounts, purchase options, and Spot Instances for these services to optimize your workload.

Consider the requirements of your workload's components, and understand the different pricing models for these services. Define the availability requirement of these components. Determine if there are multiple independent resources that perform the function in the workload, and what the workload requirements are over time. Compare the cost of the resources using the default On-Demand pricing model and other applicable models. Factor in any potential changes in resources or workload components.

For example, let's look at this Web Application Architecture on AWS. This sample workload consists of multiple AWS services, such as Amazon Route 53, AWS WAF, Amazon CloudFront, Amazon EC2 instances, Amazon RDS instances, Load Balancers, Amazon S3 storage, and Amazon Elastic File System (Amazon EFS). You need to review each of these services, and identify potential cost saving opportunities with different pricing models. Some of them may be eligible for RIs or SPs, while some of them may be only available by on-demand. As the following image shows, some of the AWS services can be committed using RIs or SPs.



AWS services committed using Reserved Instances and Savings Plans

Implementation steps

- **Implement pricing models:** Using your analysis results, purchase Savings Plans, Reserved Instances, or implement Spot Instances. If it is your first commitment purchase, choose the top five or ten recommendations in the list, then monitor and analyze the results over the next month or two. AWS Cost Management Console guides you through the process. Review the RI or SP recommendations from the console, customize the recommendations (type, payment, and term), and review hourly commitment (for example $20 per hour), and then add to cart. Discounts apply automatically to eligible usage. Purchase a small amount of commitment discounts in regular cycles (for example every 2 weeks or monthly). Implement Spot Instances for workloads that can be interrupted or are stateless. Finally, select on-demand Amazon EC2 instances and allocate resources for the remaining requirements.
- **Workload review cycle:** Implement a review cycle for the workload that specifically analyzes pricing model coverage. Once the workload has the required coverage, purchase additional commitment discounts partially (every few months), or as your organization usage changes.

Resources

Related documents:

- Understanding your Savings Plans recommendations
- Accessing Reserved Instance recommendations
- How to Purchase Reserved Instances
- Instance purchasing options
- Spot Instances
- Reservation models for other AWS services
- Savings Plans Supported Services

Related videos:

- Save up to 90% and run production workloads on Spot

Related examples:

- What should you consider before purchasing Savings Plans?
- How can I use Cost Explorer to analyze my spending and usage?

COST07-BP05 Perform pricing model analysis at the management account level

Check billing and cost management tools and see recommended discounts with commitments and reservations to perform regular analysis at the management account level.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

Performing regular cost modeling helps you implement opportunities to optimize across multiple workloads. For example, if multiple workloads use On-Demand Instances at an aggregate level, the risk of change is lower, and implementing a commitment-based discount can achieve a lower overall cost. It is recommended to perform analysis in regular cycles of two weeks to one month. This allows you to make small adjustment purchases, so the coverage of your pricing models continues to evolve with your changing workloads and their components.

Use the AWS Cost Explorer recommendations tool to find opportunities for commitment discounts in your management account. Recommendations at the management account level are calculated considering usage across all of the accounts in your AWS organization that have Reserve Instances (RI) or Savings Plans (SP). They're also calculated when discount sharing is activated to recommend a commitment that maximizes savings across accounts.

While purchasing at the management account level optimizes for max savings in many cases, there may be situations where you might consider purchasing SPs at the linked account level, like when you want the discounts to apply first to usage in that particular linked account. Member account recommendations are calculated at the individual account level, to maximize savings for each isolated account. If your account owns both RI and SP commitments, they will be applied in this order:

1. Zonal RI
2. Standard RI
3. Convertible RI
4. Instance Savings Plan
5. Compute Savings Plan

If you purchase an SP at the management account level, the savings will be applied based on highest to lowest discount percentage. SPs at the management account level look across all linked accounts and apply the savings wherever the discount will be the highest. If you wish to restrict where the savings are applied, you can purchase a Savings Plan at the linked account level and any time that account is running eligible compute services, the discount will be applied there first. When the account is not running eligible compute services, the discount will be shared across the other linked accounts under the same management account. Discount sharing is turned on by default, but can be turned off if needed.

In a Consolidated Billing Family, Savings Plans are applied first to the owner account's usage, and then to other accounts' usage. This occurs only if you have sharing enabled. Your Savings Plans are applied to your highest savings percentage first. If there are multiple usages with equal savings percentages, Savings Plans are applied to the first usage with the lowest Savings Plans rate. Savings Plans continue to apply until there are no more remaining uses or your commitment is exhausted. Any remaining usage is charged at the On-Demand rates. You can refresh Savings Plans Recommendations in AWS Cost Management to generate new Savings Plans Recommendations at any time.

After analyzing flexibility of instances, you can commit by following recommendations. Create cost modeling by analyzing the workload’s short-term costs with potential different resource options, analyzing AWS pricing models, and aligning them with your business requirements to find out total cost of ownership and cost optimization opportunities.

Implementation steps

Perform a commitment discount analysis: Use Cost Explorer in your account review the Savings Plans and Reserved Instance recommendations. Make sure you understand Saving Plan recommendations, and estimate your monthly spend and monthly savings. Review recommendations at the management account level, which are calculated considering usage across all of the member accounts in your AWS organization that have RI or Savings Plans discount sharing enabled for maximum savings across accounts. You can verify that you implemented the

correct recommendations with the required discounts and risk by following the Well-Architected labs.

Resources

Related documents:

- How does AWS pricing work?
- Instance purchasing options
- Saving Plan Overview
- Saving Plan recommendations
- Accessing Reserved Instance recommendations
- Understanding your Saving Plans recommendation
- How Savings Plans apply to your AWS usage
- Saving Plans with Consolidated Billing
- Turning on shared reserved instances and Savings Plans discounts

Related videos:

- Save up to 90% and run production workloads on Spot

Related examples:

- What should I consider before purchasing a Savings Plan?
- How can I use rolling Savings Plans to reduce commitment risk?
- When to Use Spot Instances

Plan for data transfer

An advantage of the cloud is that it is a managed network service. There is no longer the need to manage and operate a fleet of switches, routers, and other associated network equipment. Networking resources in the cloud are consumed and paid for in the same way you pay for CPU and storage—you only pay for what you use. Efficient use of networking resources is required for cost optimization in the cloud.

Best practices

- COST08-BP01 Perform data transfer modeling
- COST08-BP02 Select components to optimize data transfer cost
- COST08-BP03 Implement services to reduce data transfer costs

COST08-BP01 Perform data transfer modeling

Gather organization requirements and perform data transfer modeling of the workload and each of its components. This identifies the lowest cost point for its current data transfer requirements.

Level of risk exposed if this best practice is not established: High

Implementation guidance

When designing a solution in the cloud, data transfer fees are usually neglected due to habits of designing architecture using on-premises data centers or lack of knowledge. Data transfer charges in AWS are determined by the source, destination, and volume of traffic. Factoring in these fees during the design phase can lead to cost savings. Understanding where the data transfer occurs in your workload, the cost of the transfer, and its associated benefit is very important to accurately estimate total cost of ownership (TCO). This allows you to make an informed decision to modify or accept the architectural decision. For example, you may have a Multi-Availability Zone configuration where you replicate data between the Availability Zones.

You model the components of services which transfer the data in your workload, and decide that this is an acceptable cost (similar to paying for compute and storage in both Availability Zones) to achieve the required reliability and resilience. Model the costs over different usage levels. Workload usage can change over time, and different services may be more cost effective at different levels.

While modelling your data transfer, think about how much data is ingested and where that data comes from. Additionally, consider how much data is processed and how much storage or compute capacity is needed. During modelling, follow networking best practices for your workload architecture to optimize your potential data transfer costs.

The AWS Pricing Calculator can help you see estimated costs for specific AWS services and expected data transfer. If you have a workload already running (for test purposes or in a pre-production environment), use AWS Cost Explorer or AWS Cost and Usage Report (CUR) to understand and model your data transfer costs. Configure a proof of concept (PoC) or test your

workload, and run a test with a realistic simulated load. You can model your costs at different workload demands.

Implementation steps

- **Identify requirements:** What is the primary goal and business requirements for the planned data transfer between source and destination? What is the expected business outcome at the end? Gather business requirements and define expected outcome.
- **Identify source and destination:** What is the data source and destination for the data transfer, such as within AWS Regions, to AWS services, or out to the internet?
  - Data transfer within an AWS Region
  - Data transfer between AWS Regions
  - Data transfer out to the internet
- **Identify data classifications:** What is the data classification for this data transfer? What kind of data is it? How big is the data? How frequently must data be transferred? Is data sensitive?
- **Identify AWS services or tools to use:** Which AWS services are used for this data transfer? Is it possible to use an already-provisioned service for another workload?
- **Calculate data transfer costs:** Use AWS Pricing the data transfer modeling you created previously to calculate the data transfer costs for the workload. Calculate the data transfer costs at different usage levels, for both increases and reductions in workload usage. Where there are multiple options for the workload architecture, calculate the cost for each option for comparison.
- **Link costs to outcomes:** For each data transfer cost incurred, specify the outcome that it achieves for the workload. If it is transfer between components, it may be for decoupling, if it is between Availability Zones it may be for redundancy.
- **Create data transfer modeling:** After gathering all information, create a conceptual base data transfer modeling for multiple use cases and different workloads.

Resources

Related documents:

- AWS caching solutions
- AWS Pricing
- Amazon EC2 Pricing
- Amazon VPC pricing

- Understanding data transfer charges

Related videos:

- Monitoring and Optimizing Your Data Transfer Costs
- S3 Transfer Acceleration

Related examples:

- Overview of Data Transfer Costs for Common Architectures
- AWS Prescriptive Guidance for Networking

COST08-BP02 Select components to optimize data transfer cost

All components are selected, and architecture is designed to reduce data transfer costs. This includes using components such as wide-area-network (WAN) optimization and Multi-Availability Zone (AZ) configurations

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

Architecting for data transfer minimizes data transfer costs. This may involve using content delivery networks to locate data closer to users, or using dedicated network links from your premises to AWS. You can also use WAN optimization and application optimization to reduce the amount of data that is transferred between components.

When transferring data to or within the AWS Cloud, it is essential to know the destination based on varied use cases, the nature of the data, and the available network resources in order to select the right AWS services to optimize data transfer. AWS offers a range of data transfer services tailored for diverse data migration requirements. Select the right data storage and data transfer options based on the business needs within your organization.

When planning or reviewing your workload architecture, consider the following:

- **Use VPC endpoints within AWS:** VPC endpoints allow for private connections between your VPC and supported AWS services. This allows you to avoid using the public internet, which can lead to data transfer costs.

- **Use a NAT gateway:** Use a NAT gateway so that instances in a private subnet can connect to the internet or to the services outside your VPC. Check whether the resources behind the NAT gateway that send the most traffic are in the same Availability Zone as the NAT gateway. If they are not, create new NAT gateways in the same Availability Zone as the resource to reduce cross-AZ data transfer charges.
- **Use AWS Direct Connect** AWS Direct Connect bypasses the public internet and establishes a direct, private connection between your on-premises network and AWS. This can be more cost-effective and consistent than transferring large volumes of data over the internet.
- **Avoid transferring data across Regional boundaries:** Data transfers between AWS Regions (from one Region to another) typically incur charges. It should be a very thoughtful decision to pursue a multi-Region path. For more detail, see Multi-Region scenarios.
- **Monitor data transfer:** Use Amazon CloudWatch and VPC flow logs to capture details about your data transfer and network usage. Analyze captured network traffic information in your VPCs, such as IP address or range going to and from network interfaces.
- **Analyze your network usage:** Use metering and reporting tools such as AWS Cost Explorer, CUDOS Dashboards, or CloudWatch to understand data transfer cost of your workload.

Implementation steps

- **Select components for data transfer:** Using the data transfer modeling explained in COST08-BP01 Perform data transfer modeling, focus on where the largest data transfer costs are or where they would be if the workload usage changes. Look for alternative architectures or additional components that remove or reduce the need for data transfer (or lower its cost).

Resources

Related best practices:

- COST08-BP01 Perform data transfer modeling
- COST08-BP03 Implement services to reduce data transfer costs

Related documents:

- Cloud Data Migration
- AWS caching solutions

- Deliver content faster with Amazon CloudFront

Related examples:

- Overview of Data Transfer Costs for Common Architectures
- AWS Network Optimization Tips
- Optimize performance and reduce costs for network analytics with VPC Flow Logs in Apache Parquet format

COST08-BP03 Implement services to reduce data transfer costs

Implement services to reduce data transfer. For example, use edge locations or content delivery networks (CDN) to deliver content to end users, build caching layers in front of your application servers or databases, and use dedicated network connections instead of VPN for connectivity to the cloud.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

There are various AWS services that can help you to optimize your network data transfer usage. Depending on your workload components, type, and cloud architecture, these services can assist you in compression, caching, and sharing and distribution of your traffic on the cloud.

- Amazon CloudFront is a global content delivery network that delivers data with low latency and high transfer speeds. It caches data at edge locations across the world, which reduces the load on your resources. By using CloudFront, you can reduce the administrative effort in delivering content to large numbers of users globally with minimum latency. The security savings bundle can help you to save up to 30% on your CloudFront usage if you plan to grow your usage over time.
- AWS Direct Connect allows you to establish a dedicated network connection to AWS. This can reduce network costs, increase bandwidth, and provide a more consistent network experience than internet-based connections.
- AWS VPN allows you to establish a secure and private connection between your private network and the AWS global network. It is ideal for small offices or business partners because it provides simplified connectivity, and it is a fully managed and elastic service.

- VPC Endpoints allow connectivity between AWS services over private networking and can be used to reduce public data transfer and NAT gateway costs. Gateway VPC endpoints have no hourly charges, and support Amazon S3 and Amazon DynamoDB. Interface VPC endpoints are provided by AWS PrivateLink and have an hourly fee and per-GB usage cost.
- NAT gateways provide built-in scaling and management for reducing costs as opposed to a standalone NAT instance. Place NAT gateways in the same Availability Zones as high traffic instances and consider using VPC endpoints for the instances that need to access Amazon DynamoDB or Amazon S3 to reduce the data transfer and processing costs.
- Use AWS Snow Family devices which have computing resources to collect and process data at the edge. AWS Snow Family devices (Snowball Edge, Snowball Edge and Snowmobile) allow you to move petabytes of data to the AWS Cloud cost effectively and offline.

Implementation steps

- **Implement services:** Select applicable AWS network services based on your service workload type using the data transfer modeling and reviewing VPC Flow Logs. Look at where the largest costs and highest volume flows are. Review the AWS services and assess whether there is a service that reduces or removes the transfer, specifically networking and content delivery. Also look for caching services where there is repeated access to data or large amounts of data.

Resources

Related documents:

- AWS Direct Connect
- AWS Explore Our Products
- AWS caching solutions
- Amazon CloudFront
- AWS Snow Family
- Amazon CloudFront Security Savings Bundle

Related videos:

- Monitoring and Optimizing Your Data Transfer Costs
- AWS Cost Optimization Series: CloudFront

- How can I reduce data transfer charges for my NAT gateway?

Related examples:

- How-to chargeback shared services: An AWS Transit Gateway example
- Understand AWS data transfer details in depth from cost and usage report using Athena query and QuickSight
- Overview of Data Transfer Costs for Common Architectures
- Using AWS Cost Explorer to analyze data transfer costs
- Cost-Optimizing your AWS architectures by utilizing Amazon CloudFront features
- How can I reduce data transfer charges for my NAT gateway?

Manage demand and supply resources

When you move to the cloud, you pay only for what you need. You can supply resources to match the workload demand at the time they're needed — eliminating the need for costly and wasteful overprovisioning. You can also modify the demand using a throttle, buffer, or queue to smooth the demand and serve it with less resources.

The economic benefits of just-in-time supply should be balanced against the need to provision to account for resource failures, high availability, and provision time. Depending on whether your demand is fixed or variable, plan to create metrics and automation that will ensure that management of your environment is minimal – even as you scale. When modifying the demand, you must know the acceptable and maximum delay that the workload can allow.

In AWS, you can use a number of different approaches for managing demand and supplying resources. The following best practices describe how to use these approaches.

Best practices

- COST09-BP01 Perform an analysis on the workload demand
- COST09-BP02 Implement a buffer or throttle to manage demand
- COST09-BP03 Supply resources dynamically

COST09-BP01 Perform an analysis on the workload demand

Analyze the demand of the workload over time. Verify that the analysis covers seasonal trends and accurately represents operating conditions over the full workload lifetime. Analysis effort should reflect the potential benefit, for example, time spent is proportional to the workload cost.

Level of risk exposed if this best practice is not established: High

Implementation guidance

Analyzing workload demand for cloud computing involves understanding the patterns and characteristics of computing tasks that are initiated in the cloud environment. This analysis helps users optimize resource allocation, manage costs, and verify that performance meets required levels.

Know the requirements of the workload. Your organization's requirements should indicate the workload response times for requests. The response time can be used to determine if the demand is managed, or if the supply of resources should change to meet the demand.

The analysis should include the predictability and repeatability of the demand, the rate of change in demand, and the amount of change in demand. Perform the analysis over a long enough period to incorporate any seasonal variance, such as end-of-month processing or holiday peaks.

Analysis effort should reflect the potential benefits of implementing scaling. Look at the expected total cost of the component and any increases or decreases in usage and cost over the workload's lifetime.

The following are some key aspects to consider when performing workload demand analysis for cloud computing:

1. **Resource utilization and performance metrics**: Analyze how AWS resources are being used over time. Determine peak and off-peak usage patterns to optimize resource allocation and scaling strategies. Monitor performance metrics such as response times, latency, throughput, and error rates. These metrics help assess the overall health and efficiency of the cloud infrastructure.
2. **User and application scaling behaviour**: Understand user behavior and how it affects workload demand. Examining the patterns of user traffic assists in enhancing the delivery of content and the responsiveness of applications. Analyze how workloads scale with increasing demand. Determine whether auto-scaling parameters are configured correctly and effectively for handling load fluctuations.
3. **Workload types**: Identify the different types of workloads running in the cloud, such as batch processing, real-time data processing, web applications, databases, or machine learning. Each type of workload may have different resource requirements and performance profiles.
4. **Service-level agreements (SLAs)**: Compare actual performance with SLAs to ensure compliance and identify areas that need improvement.

You can use Amazon CloudWatch to collect and track metrics, monitor log files, set alarms, and automatically react to changes in your AWS resources. You can also use Amazon CloudWatch to gain system-wide visibility into resource utilization, application performance, and operational health.

With AWS Trusted Advisor, you can provision your resources following best practices to improve system performance and reliability, increase security, and look for opportunities to save money.

You can also turn off non-production instances and use Amazon CloudWatch and Auto Scaling to match increases or reductions in demand.

Finally, you can use AWS Cost Explorer or QuickSight with the AWS Cost and Usage Report (CUR) file or your application logs to perform advanced analysis of workload demand.

Overall, a comprehensive workload demand analysis allows organizations to make informed decisions about resource provisioning, scaling, and optimization, leading to better performance, cost efficiency, and user satisfaction.

Implementation steps

- **Analyze existing workload data:** Analyze data from the existing workload, previous versions of the workload, or predicted usage patterns. Use Amazon CloudWatch, log files and monitoring data to gain insight on how workload was used. Analyze a full cycle of the workload, and collect data for any seasonal changes such as end-of-month or end-of-year events. The effort reflected in the analysis should reflect the workload characteristics. The largest effort should be placed on high-value workloads that have the largest changes in demand. The least effort should be placed on low-value workloads that have minimal changes in demand.
- **Forecast outside influence:** Meet with team members from across the organization that can influence or change the demand in the workload. Common teams would be sales, marketing, or business development. Work with them to know the cycles they operate within, and if there are any events that would change the demand of the workload. Forecast the workload demand with this data.

Resources

Related documents:

- Amazon CloudWatch
- AWS Trusted Advisor
- AWS X-Ray
- AWS Auto Scaling
- AWS Instance Scheduler
- Getting started with Amazon SQS
- AWS Cost Explorer

- QuickSight

Related examples:

- Monitor, Track and Analyze for cost optimization
- Searching and analyzing logs in CloudWatch

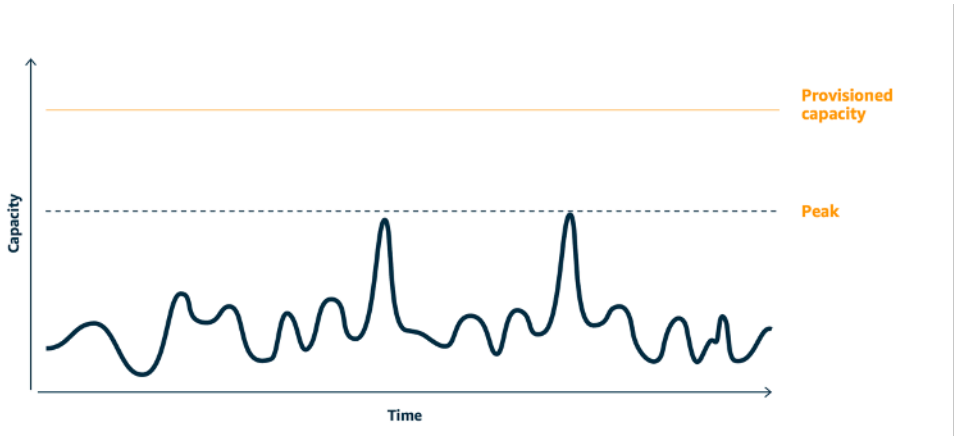
COST09-BP02 Implement a buffer or throttle to manage demand

Buffering and throttling modify the demand on your workload, smoothing out any peaks. Implement throttling when your clients perform retries. Implement buffering to store the request and defer processing until a later time. Verify that your throttles and buffers are designed so clients receive a response in the required time.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

Implementing a buffer or throttle is crucial in cloud computing in order to manage demand and reduce the provisioned capacity required for your workload. For optimal performance, it's essential to gauge the total demand, including peaks, the pace of change in requests, and the necessary response time. When clients have the ability to resend their requests, it becomes practical to apply throttling. Conversely, for clients lacking retry functionalities, the ideal approach is implementing a buffer solution. Such buffers streamline the influx of requests and optimize the interaction of applications with varied operational speeds.



Demand curve with two distinct peaks that require high provisioned capacity

Assume a workload with the demand curve shown in preceding image. This workload has two peaks, and to handle those peaks, the resource capacity as shown by orange line is provisioned. The resources and energy used for this workload are not indicated by the area under the demand curve, but the area under the provisioned capacity line, as provisioned capacity is needed to handle those two peaks. Flattening the workload demand curve can help you to reduce the provisioned capacity for a workload and reduce its environmental impact. To smooth out the peak, consider to implement throttling or buffering solution.

To understand them better, let's explore throttling and buffering.

Throttling: If the source of the demand has retry capability, then you can implement throttling. Throttling tells the source that if it cannot service the request at the current time, it should try again later. The source waits for a period of time, and then retries the request. Implementing throttling has the advantage of limiting the maximum amount of resources and costs of the workload. In AWS, you can use Amazon API Gateway to implement throttling.

Buffer based: A buffer-based approach uses *producers* (components that send messages to the queue), *consumers* (components that receive messages from the queue), and a *queue* (which holds messages) to store the messages. Messages are read by consumers and processed, allowing the messages to run at the rate that meets the consumers' business requirements. By using a buffer-centric methodology, messages from producers are housed in queues or streams, ready to be accessed by consumers at a pace that aligns with their operational demands.

In AWS, you can choose from multiple services to implement a buffering approach. Amazon Simple Queue Service(Amazon SQS) is a managed service that provides queues that allow a single consumer to read individual messages. Amazon Kinesis provides a stream that allows many consumers to read the same messages.

Buffering and throttling can smooth out any peaks by modifying the demand on your workload. Use throttling when clients retry actions and use buffering to hold the request and process it later. When working with a buffer-based approach, architect your workload to service the request in the required time, verify that you are able to handle duplicate requests for work. Analyze the overall demand, rate of change, and required response time to right size the throttle or buffer required.

Implementation steps

- **Analyze the client requirements:** Analyze the client requests to determine if they are capable of performing retries. For clients that cannot perform retries, buffers need to be implemented.

Analyze the overall demand, rate of change, and required response time to determine the size of throttle or buffer required.

- **Implement a buffer or throttle:** Implement a buffer or throttle in the workload. A queue such as Amazon Simple Queue Service (Amazon SQS) can provide a buffer to your workload components. Amazon API Gateway can provide throttling for your workload components.

Resources

Related best practices:

- SUS02-BP06 Implement buffering or throttling to flatten the demand curve
- REL05-BP02 Throttle requests

Related documents:

- AWS Auto Scaling
- AWS Instance Scheduler
- Amazon API Gateway
- Amazon Simple Queue Service
- Getting started with Amazon SQS
- Amazon Kinesis

Related videos:

- Choosing the Right Messaging Service for Your Distributed App

Related examples:

- Managing and monitoring API throttling in your workloads
- Throttling a tiered, multi-tenant REST API at scale using API Gateway
- Enabling Tiering and Throttling in a Multi-Tenant Amazon EKS SaaS Solution Using Amazon API Gateway
- Application integration Using Queues and Messages

COST09-BP03 Supply resources dynamically

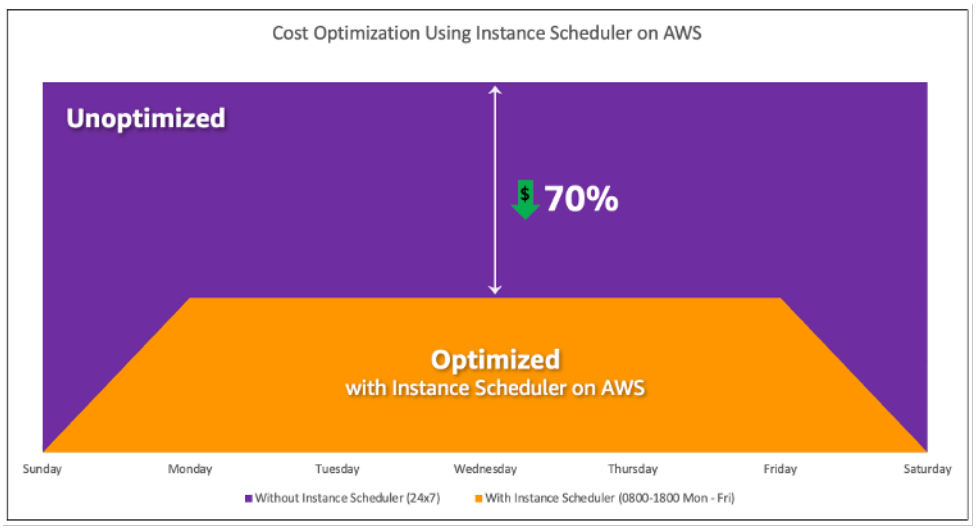
Resources are provisioned in a planned manner. This can be demand-based, such as through automatic scaling, or time-based, where demand is predictable and resources are provided based on time. These methods result in the least amount of over-provisioning or under-provisioning.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

There are several ways for AWS customers to increase the resources available to their applications and supply resources to meet the demand. One of these options is to use AWS Instance Scheduler, which automates the starting and stopping of Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Relational Database Service (Amazon RDS) instances. The other option is to use AWS Auto Scaling, which allows you to automatically scale your computing resources based on the demand of your application or service. Supplying resources based on demand will allow you to pay for the resources you use only, reduce cost by launching resources when they are needed, and terminate them when they aren't.

AWS Instance Scheduler allows you to configure the stop and start of your Amazon EC2 and Amazon RDS instances at defined times so that you can meet the demand for the same resources within a consistent time pattern such as every day user access Amazon EC2 instances at eight in the morning that they don’t need after six at night. This solution helps reduce operational cost by stopping resources that are not in use and starting them when they are needed.



Cost optimization with AWS Instance Scheduler.

You can also easily configure schedules for your Amazon EC2 instances across your accounts and Regions with a simple user interface (UI) using AWS Systems Manager Quick Setup. You can schedule Amazon EC2 or Amazon RDS instances with AWS Instance Scheduler and you can stop and start existing instances. However, you cannot stop and start instances which are part of your Auto Scaling group (ASG) or that manage services such as Amazon Redshift or Amazon OpenSearch Service. Auto Scaling groups have their own scheduling for the instances in the group and these instances are created.

AWS Auto Scaling helps you adjust your capacity to maintain steady, predictable performance at the lowest possible cost to meet changing demand. It is a fully managed and free service to scale the capacity of your application that integrates with Amazon EC2 instances and Spot Fleets, Amazon ECS, Amazon DynamoDB, and Amazon Aurora. Auto Scaling provides automatic resource discovery to help find resources in your workload that can be configured, it has built-in scaling strategies to optimize performance, costs, or a balance between the two, and provides predictive scaling to assist with regularly occurring spikes.

There are multiple scaling options available to scale your Auto Scaling group:

- Maintain current instance levels at all times
- Scale manually
- Scale based on a schedule
- Scale based on demand
- Use predictive scaling

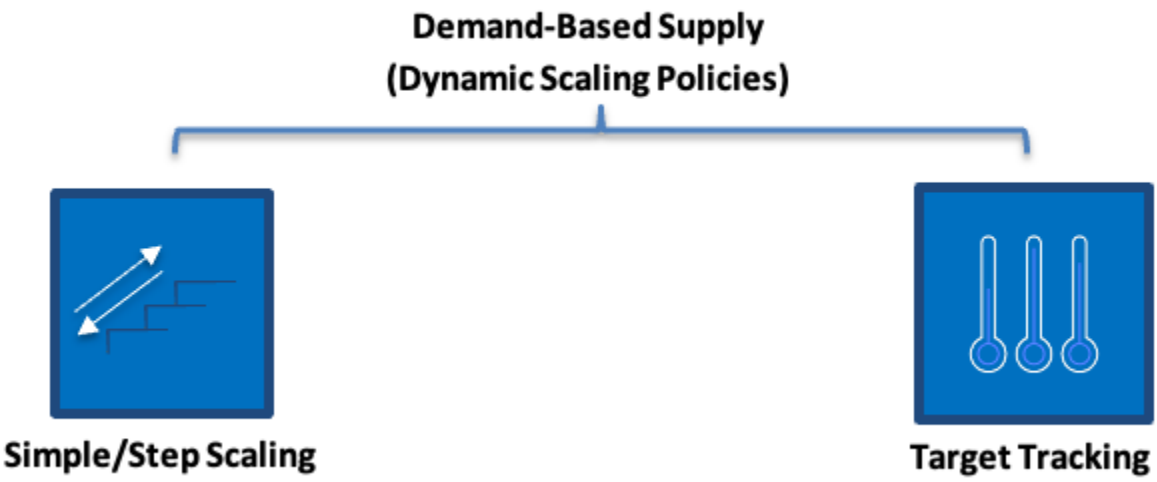
Auto Scaling policies differ and can be categorized as dynamic and scheduled scaling policies. Dynamic policies are manual or dynamic scaling which, scheduled or predictive scaling. You can use scaling policies for dynamic, scheduled, and predictive scaling. You can also use metrics and alarms from Amazon CloudWatch to trigger scaling events for your workload. We recommend you use launch templates, which allow you to access the latest features and improvements. Not all Auto Scaling features are available when you use launch configurations. For example, you cannot create an Auto Scaling group that launches both Spot and On-Demand Instances or that specifies multiple instance types. You must use a launch template to configure these features. When using launch templates, we recommended you version each one. With versioning of launch templates, you can create a subset of the full set of parameters. Then, you can reuse it to create other versions of the same launch template.

You can use AWS Auto Scaling or incorporate scaling in your code with AWS APIs or SDKs. This reduces your overall workload costs by removing the operational cost from manually making changes to your environment, and changes can be performed much faster. This also matches your workload resourcing to your demand at any time. In order to follow this best practice and supply resources dynamically for your organization, you should understand horizontal and vertical scaling in the AWS Cloud, as well as the nature of the applications running on Amazon EC2 instances. It is better for your Cloud Financial Management team to work with technical teams to follow this best practice.

Elastic Load Balancing (Elastic Load Balancing) helps you scale by distributing demand across multiple resources. With using ASG and Elastic Load Balancing, you can manage incoming requests by optimally routing traffic so that no one instance is overwhelmed in an Auto Scaling group. The requests would be distributed among all the targets of a target group in a round-robin fashion without consideration for capacity or utilization.

Typical metrics can be standard Amazon EC2 metrics, such as CPU utilization, network throughput, and Elastic Load Balancing observed request and response latency. When possible, you should use a metric that is indicative of customer experience, typically a custom metric that might originate from application code within your workload. To elaborate how to meet the demand dynamically in this document, we will group Auto Scaling into two categories as demand-based and time-based supply models and deep dive into each.

Demand-based supply: Take advantage of elasticity of the cloud to supply resources to meet changing demand by relying on near real-time demand state. For demand-based supply, use APIs or service features to programmatically vary the amount of cloud resources in your architecture. This allows you to scale components in your architecture and increase the number of resources during demand spikes to maintain performance and decrease capacity when demand subsides to reduce costs.

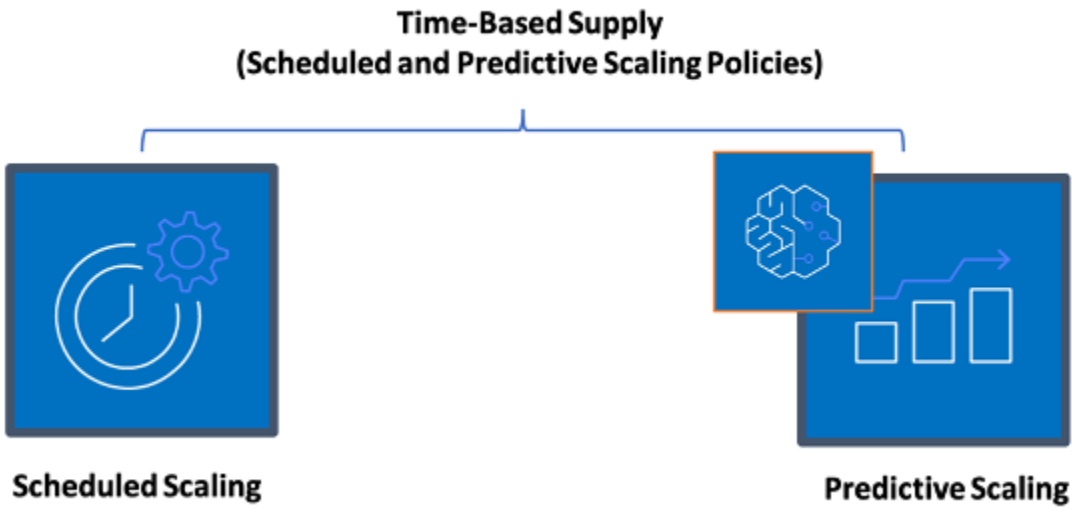


Demand-based dynamic scaling policies

- **Simple/Step Scaling:** Monitors metrics and adds/removes instances as per steps defined by the customers manually.
- **Target Tracking:** Thermostat-like control mechanism that automatically adds or removes instances to maintain metrics at a customer defined target.

When architecting with a demand-based approach keep in mind two key considerations. First, understand how quickly you must provision new resources. Second, understand that the size of margin between supply and demand will shift. You must be ready to cope with the rate of change in demand and also be ready for resource failures.

Time-based supply: A time-based approach aligns resource capacity to demand that is predictable or well-defined by time. This approach is typically not dependent upon utilization levels of the resources. A time-based approach ensures that resources are available at the specific time they are required and can be provided without any delays due to start-up procedures and system or consistency checks. Using a time-based approach, you can provide additional resources or increase capacity during busy periods.



Time-based scaling policies

You can use scheduled or predictive auto scaling to implement a time-based approach. Workloads can be scheduled to scale out or in at defined times (for example, the start of business hours), making resources available when users arrive or demand increases. Predictive scaling uses patterns to scale out while scheduled scaling uses pre-defined times to scale out. You can also use attribute-based instance type selection (ABS) strategy in Auto Scaling groups, which lets you express your instance requirements as a set of attributes, such as vCPU, memory, and storage. This also allows you to automatically use newer generation instance types when they are released and access a broader range of capacity with Amazon EC2 Spot Instances. Amazon EC2 Fleet and Amazon EC2 Auto Scaling select and launch instances that fit the specified attributes, removing the need to manually pick instance types.

You can also leverage the AWS APIs and SDKs and AWS CloudFormation to automatically provision and decommission entire environments as you need them. This approach is well suited for development or test environments that run only in defined business hours or periods of time. You can use APIs to scale the size of resources within an environment (vertical scaling). For example, you could scale up a production workload by changing the instance size or class. This can be achieved by stopping and starting the instance and selecting the different instance size or class. This technique can also be applied to other resources, such as Amazon EBS Elastic Volumes, which can be modified to increase size, adjust performance (IOPS) or change the volume type while in use.

When architecting with a time-based approach keep in mind two key considerations. First, how consistent is the usage pattern? Second, what is the impact if the pattern changes? You can increase the accuracy of predictions by monitoring your workloads and by using business intelligence. If you see significant changes in the usage pattern, you can adjust the times to ensure that coverage is provided.

Implementation steps

- **Configure scheduled scaling:** For predictable changes in demand, time-based scaling can provide the correct number of resources in a timely manner. It is also useful if resource creation and configuration is not fast enough to respond to changes on demand. Using the workload analysis configure scheduled scaling using AWS Auto Scaling. To configure time-based scheduling, you can use predictive scaling of scheduled scaling to increase the number of Amazon EC2 instances in your Auto Scaling groups in advance according to expected or predictable load changes.
- **Configure predictive scaling:** Predictive scaling allows you to increase the number of Amazon EC2 instances in your Auto Scaling group in advance of daily and weekly patterns in traffic flows. If you have regular traffic spikes and applications that take a long time to start, you should consider using predictive scaling. Predictive scaling can help you scale faster by initializing capacity before projected load compared to dynamic scaling alone, which is reactive in nature. For example, if users start using your workload with the start of the business hours and don't use after hours, then predictive scaling can add capacity before the business hours which eliminates delay of dynamic scaling to react to changing traffic.
- **Configure dynamic automatic scaling:** To configure scaling based on active workload metrics, use Auto Scaling. Use the analysis and configure Auto Scaling to launch on the correct resource levels, and verify that the workload scales in the required time. You can launch and automatically scale a fleet of On-Demand Instances and Spot Instances within a single Auto Scaling group. In addition to receiving discounts for using Spot Instances, you can use Reserved Instances or a Savings Plan to receive discounted rates of the regular On-Demand Instance pricing. All of these factors combined help you to optimize your cost savings for Amazon EC2 instances and help you get the desired scale and performance for your application.

Resources

Related documents:

- AWS Auto Scaling

- AWS Instance Scheduler
- Scale the size of your Auto Scaling group
- Getting Started with Amazon EC2 Auto Scaling
- Getting started with Amazon SQS
- Scheduled Scaling for Amazon EC2 Auto Scaling
- Predictive scaling for Amazon EC2 Auto Scaling

Related videos:

- Target Tracking Scaling Policies for Auto Scaling
- AWS Instance Scheduler

Related examples:

- Attribute based Instance Type Selection for Auto Scaling for Amazon EC2 Fleet
- Optimizing Amazon Elastic Container Service for cost using scheduled scaling
- Predictive Scaling with Amazon EC2 Auto Scaling
- How do I use Instance Scheduler with AWS CloudFormation to schedule Amazon EC2 instances?

Optimize over time

In AWS, you optimize over time by reviewing new services and implementing them in your workload.

As AWS releases new services and features, it is a best practice to review your existing architectural decisions to ensure that they remain cost effective. As your requirements change, be aggressive in decommissioning resources, components, and workloads that you no longer require. Consider the following best practices to help you optimize over time.

While optimizing your workloads over time and improving your CFM culture in your organization, evaluate the cost of effort for operations in the cloud, review your time-consuming cloud operations, and automate them to reduce human efforts and cost by adopting related AWS services, third party products, or custom tools (like AWS CLI or AWS SDKs).

Topics

- Define a review process and analyze your workload regularly
- Automating operations

Define a review process and analyze your workload regularly

Best practices

- COST10-BP01 Develop a workload review process
- COST10-BP02 Review and analyze this workload regularly

COST10-BP01 Develop a workload review process

Develop a process that defines the criteria and process for workload review. The review effort should reflect potential benefit. For example, core workloads or workloads with a value of over ten percent of the bill are reviewed quarterly or every six months, while workloads below ten percent are reviewed annually.

Level of risk exposed if this best practice is not established: High

Implementation guidance

To have the most cost-efficient workload, you must regularly review the workload to know if there are opportunities to implement new services, features, and components. To achieve overall lower costs the process must be proportional to the potential amount of savings. For example, workloads that are 50% of your overall spend should be reviewed more regularly, and more thoroughly, than workloads that are five percent of your overall spend. Factor in any external factors or volatility. If the workload services a specific geography or market segment, and change in that area is predicted, more frequent reviews could lead to cost savings. Another factor in review is the effort to implement changes. If there are significant costs in testing and validating changes, reviews should be less frequent.

Factor in the long-term cost of maintaining outdated and legacy, components and resources and the inability to implement new features into them. The current cost of testing and validation may exceed the proposed benefit. However, over time, the cost of making the change may significantly increase as the gap between the workload and the current technologies increases, resulting in even larger costs. For example, the cost of moving to a new programming language may not currently be cost effective. However, in five years time, the cost of people skilled in that language may increase, and due to workload growth, you would be moving an even larger system to the new language, requiring even more effort than previously.

Break down your workload into components, assign the cost of the component (an estimate is sufficient), and then list the factors (for example, effort and external markets) next to each component. Use these indicators to determine a review frequency for each workload. For example, you may have webservers as a high cost, low change effort, and high external factors, resulting in high frequency of review. A central database may be medium cost, high change effort, and low external factors, resulting in a medium frequency of review.

Define a process to evaluate new services, design patterns, resource types, and configurations to optimize your workload cost as they become available. Similar to performance pillar review and reliability pillar review processes, identify, validate, and prioritize optimization and improvement activities and issue remediation and incorporate this into your backlog.

Implementation steps

- **Define review frequency:** Define how frequently the workload and its components should be reviewed. Allocate time and resources to continual improvement and review frequency to improve the efficiency and optimization of your workload. This is a combination of factors and may differ from workload to workload within your organization and between components in

the workload. Common factors include the importance to the organization measured in terms of revenue or brand, the total cost of running the workload (including operation and resource costs), the complexity of the workload, how easy is it to implement a change, any software licensing agreements, and if a change would incur significant increases in licensing costs due to punitive licensing. Components can be defined functionally or technically, such as web servers and databases, or compute and storage resources. Balance the factors accordingly and develop a period for the workload and its components. You may decide to review the full workload every 18 months, review the web servers every six months, the database every 12 months, compute and short-term storage every six months, and long-term storage every 12 months.

- **Define review thoroughness:** Define how much effort is spent on the review of the workload or workload components. Similar to the review frequency, this is a balance of multiple factors. Evaluate and prioritize opportunities for improvement to focus efforts where they provide the greatest benefits while estimating how much effort is required for these activities. If the expected outcomes do not satisfy the goals, and required effort costs more, then iterate using alternative courses of action. Your review processes should include dedicated time and resources to make continuous incremental improvements possible. As an example, you may decide to spend one week of analysis on the database component, one week of analysis for compute resources, and four hours for storage reviews.

Resources

Related documents:

- AWS News Blog
- Types of Cloud Computing
- What's New with AWS

Related examples:

- AWS Support Proactive Services
- Regular workload reviews for SAP workloads

COST10-BP02 Review and analyze this workload regularly

Existing workloads are regularly reviewed based on each defined process to find out if new services can be adopted, existing services can be replaced, or workloads can be re-architected.

Level of risk exposed if this best practice is not established: Medium

Implementation guidance

AWS is constantly adding new features so you can experiment and innovate faster with the latest technology. AWS What's New details how AWS is doing this and provides a quick overview of AWS services, features, and Regional expansion announcements as they are released. You can dive deeper into the launches that have been announced and use them for your review and analyze of your existing workloads. To realize the benefits of new AWS services and features, you review on your workloads and implement new services and features as required. This means you may need to replace existing services you use for your workload, or modernize your workload to adopt these new AWS services. For example, you might review your workloads and replace the messaging component with Amazon Simple Email Service. This removes the cost of operating and maintaining a fleet of instances, while providing all the functionality at a reduced cost.

To analyze your workload and highlight potential opportunities, you should consider not only new services but also new ways of building solutions. Review the This is My Architecture videos on AWS to learn about other customers' architecture designs, their challenges and their solutions. Check the All-In series to find out real world applications of AWS services and customer stories. You can also watch the Back to Basics video series that explains, examines, and breaks down basic cloud architecture pattern best practices. Another source is How to Build This videos, which are designed to assist people with big ideas on how to bring their minimum viable product (MVP) to life using AWS services. It is a way for builders from all over the world who have a strong idea to gain architectural guidance from experienced AWS Solutions Architects. Finally, you can review the Getting Started resource materials, which has step by step tutorials.

Before starting your review process, follow your business' requirements for the workload, security and data privacy requirements in order to use specific service or Region and performance requirements while following your agreed review process.

Implementation steps

- **Regularly review the workload:** Using your defined process, perform reviews with the frequency specified. Verify that you spend the correct amount of effort on each component. This process would be similar to the initial design process where you selected services for cost optimization. Analyze the services and the benefits they would bring, this time factor in the cost of making the change, not just the long-term benefits.

- **Implement new services:** If the outcome of the analysis is to implement changes, first perform a baseline of the workload to know the current cost for each output. Implement the changes, then perform an analysis to confirm the new cost for each output.

Resources

Related documents:

- AWS News Blog
- What's New with AWS
- AWS Documentation
- AWS Getting Started
- AWS General Resources

Related videos:

- AWS - This is My Architecture
- AWS - Back to Basics
- AWS - All-In series
- How to Build This

Automating operations

Best practices

- COST11-BP01 Perform automation for operations

COST11-BP01 Perform automation for operations

Evaluate the operational costs on the cloud, focusing on quantifying the time and effort savings in administrative tasks, deployments, mitigating the risk of human errors, compliance, and other operations through automation. Assess the time and associated costs required for operational efforts and implement automation for administrative tasks to minimize manual effort wherever feasible.

Level of risk exposed if this best practice is not established: Low

Implementation guidance

Automating operations reduces the frequency of manual tasks, improves efficiency, and benefits customers by delivering a consistent and reliable experience when deploying, administering, or operating workloads. You can free up infrastructure resources from manual operational tasks and use them for higher value tasks and innovations, which improves business value. Enterprises require a proven, tested way to manage their workloads in the cloud. That solution must be secure, fast, and cost effective, with minimum risk and maximum reliability.

Start by prioritizing your operational activities based on required effort by looking at overall operations cost. For example, how long does it take to deploy new resources in the cloud, make optimization changes to existing ones, or implement necessary configurations? Look at the total cost of human actions by factoring in cost of operations and management. Prioritize automations for admin tasks to reduce the human effort.

Review effort should reflect the potential benefit. For example, examine time spent performing tasks manually as opposed to automatically. Prioritize automating repetitive, high value, time consuming and complex activities. Activities that pose a high value or high risk of human error are typically the better place to start automating as the risk often poses an unwanted additional operational cost (like operations team working extra hours).

Use automation tools like AWS Systems Manager or AWS Config to streamline operations, compliance, monitoring, lifecycle, and termination processes. With AWS services, tools, and third-party products, you can customize the automations you implement to meet your specific requirement. Following table shows some of the core operation functions and capabilities you can achieve with AWS services to automate administration and operation:

- AWS Audit Manager: Continually audit your AWS usage to simplify risk and compliance assessment
- AWS Backup: Centrally manage and automate data protection.
- AWS Config: Configure compute resources, asses, audit, evaluate configurations and resource inventory.
- AWS CloudFormation: Launch highly available resources with Infrastructure as Code.
- AWS CloudTrail: IT change management, compliance, and control.
- Amazon EventBridge Schedule events and trigger AWS Lambda to take action.
- AWS Lambda: Automate repetitive processes by triggering them with events or by running them on a fixed schedule with AWS EventBridge.

- AWS Systems Manager: Start and stop workloads, patch operating systems, automate configuration, and ongoing management.
- AWS Step Functions: Schedule jobs and automate workflows.
- AWS Service Catalog: Template consumption, infrastructure as code with compliance and control.

If you would like to adopt automations immediately with using AWS products and service and if don't have skills in your organization, reach out to AWS Managed Services (AMS), AWS Professional Services, or AWS Partners to increase adoption of automation and improve your operational excellence in the cloud.

AWS Managed Services (AMS) is a service that operates AWS infrastructure on behalf of enterprise customers and partners. It provides a secure and compliant environment that you can deploy your workloads onto. AMS uses enterprise cloud operating models with automation to allow you to meet your organization requirements, move into the cloud faster, and reduce your on-going management costs.

AWS Professional Services can also help you achieve your desired business outcomes and automate operations with AWS. They help customers to deploy automated, robust, agile IT operations, and governance capabilities optimized for the cloud. For detailed monitoring examples and recommended best practices, see Operational Excellence Pillar whitepaper.

Implementation steps

- **Build once and deploy many**: Use infrastructure-as-code such as CloudFormation, AWS SDK, or AWS CLI to deploy once and use many times for similar environments or for disaster recovery scenarios. Tag while deploying to track your consumption as defined in other best practices. Use AWS Launch Wizard to reduce the time to deploy many popular enterprise workloads. AWS Launch Wizard guides you through the sizing, configuration, and deployment of enterprise workloads following AWS best practices. You can also use the Service Catalog, which helps you create and manage infrastructure-as-code approved templates for use on AWS so anyone can discover approved, self-service cloud resources.
- **Automate continuous compliance:** Consider automating assessment and remediation of recorded configurations against predefined standards. When you combine AWS Organizations with the capabilities of AWS Config and AWS CloudFormation, you can efficiently manage and automate configuration compliance at scale for hundreds of member accounts. You can review changes in configurations and relationships between AWS resources and dive into the history of a resource configuration.

- **Automate monitoring tasks** AWS provides various tools that you can use to monitor services. You can configure these tools to automate monitoring tasks. Create and implement a monitoring plan that collects monitoring data from all the parts in your workload so that you can more easily debug a multi-point failure if one occurs. For example, you can use the automated monitoring tools to observe Amazon EC2 and report back to you when something is wrong for system status checks, instance status checks, and Amazon CloudWatch alarms.
- **Automate maintenance and operations**: Run routine operations automatically without human intervention. Using AWS services and tools, you can choose which AWS automations to implement and customize for your specific requirements. For example, use EC2 Image Builder for building, testing, and deployment of virtual machine and container images for use on AWS or on-premises or patching your EC2 instances with AWS SSM. If your desired action cannot be done with AWS services or you need more complex actions with filtering resources, then automate your operations with using AWS Command Line Interface (AWS CLI) or AWS SDK tools. AWS CLI provides the ability to automate the entire process of controlling and managing AWS services with scripts without using the AWS Management Console. Select your preferred AWS SDKs to interact with AWS services. For other code examples, see AWS SDK Code examples repository.
- **Create a continual lifecycle with automations:** It is important that you establish and preserve mature lifecycle policies not only for regulations or redundancy but also for cost optimization. You can use AWS Backup to centrally manage and automate data protection of data stores, such as your buckets, volumes, databases, and file systems. You can also use Amazon Data Lifecycle Manager to automate the creation, retention, and deletion of EBS snapshots and EBS-backed AMIs.
- **Delete unnecessary resources:** It's quite common to accumulate unused resources in sandbox or development AWS accounts. Developers create and experiment with various services and resources as part of the normal development cycle, and then they don't delete those resources when they're no longer needed. Unused resources can incur unnecessary and sometimes high costs for the organization. Deleting these resources can reduce the costs of operating these environments. Make sure your data is not needed or backed up if you are not sure. You can use AWS CloudFormation to clean up deployed stacks, which automatically deletes most resources defined in the template. Alternatively, you can create an automation for the deletion of AWS resources using tools like aws-nuke.

Resources

Related documents:

- Modernizing operations in the AWS Cloud
- AWS Services for Automation
- Infrastructure and automation
- AWS Systems Manager Automation
- Automated and manual monitoring
- AWS automations for SAP administration and operations
- AWS Managed Services
- AWS Professional Services

Related videos:

- Automate Continuous Compliance at Scale in AWS
- AWS Backup Demo: Cross-Account & Cross-Region Backup
- Patching for your Amazon EC2 Instances

Related examples:

- Reinventing automated operations (Part I)
- Reinventing automated operations (Part II)
- Automate deletion of AWS resources by using aws-nuke
- Delete unused Amazon EBS volumes by using AWS Config and AWS SSM
- Automate continuous compliance at scale in AWS
- IT Automations with AWS Lambda

Conclusion

Cost optimization and Cloud Financial Management is an ongoing effort. You should regularly work with your finance and technology teams, review your architectural approach, and update your component selection.

AWS strives to help you minimize cost while you build highly resilient, responsive, and adaptive deployments. To truly optimize the cost of your deployment, take advantage of the tools, techniques, and best practices discussed in this paper.

Contributors

Contributors to this document include:

- Fatih (Ben) Mergen, Cost Optimization Pillar Lead, Well-Architected, Amazon Web Services
- Keith Jarrett, Business Development Lead – Cost Optimization, Amazon Web Services
- Arthur Basbaum, Business Developer Manager, Amazon Web Services
- Jarman Hauser, Commercial Architect, Amazon Web Services

Further reading

For additional information, see:

- AWS Well-Architected Framework
- AWS Architecture Center

Document revisions

To be notified about updates to this whitepaper, subscribe to the RSS feed.

Change	Description	Date
Updated best practice guidance	Multiple best practice updates. New best practice COST06-BP04.	June 27, 2024
Updated best practice guidance	Minor best practice updates throughout.	December 6, 2023
Updated best practice guidance	Best practices were updated with new guidance across the pillar.	October 3, 2023
Updated best practice guidance	Best practices were updated with new guidance in the following areas: Governance, Monitor cost and usage, Select the best pricing model, and Manage demand and supply resources.	July 13, 2023
Updates for new Framework	Best practices updated with prescriptive guidance and new best practices added. Question COST 11 added with new best practice COST11-BP01.	April 10, 2023
Minor update	Missing guidance restored to the pricing model section.	January 13, 2023
Whitepaper updated	Best practices updated with new implementation guidance.	December 15, 2022

Whitepaper updated	Best practices expanded and improvement plans added.	October 20, 2022
Minor update	Added Sustainability Pillar to introduction.	December 2, 2021
Minor update	Updated links.	April 25, 2021
Minor update	Updated links.	March 10, 2021
Updates for new Framework	Updated to incorporate CFM, new services, and integration with the Well-Architected too.	July 8, 2020
Whitepaper updated	Updated to reflect changes to AWS and incorporate learnings from reviews with customers.	July 1, 2018
Whitepaper updated	Updated to reflect changes to AWS and incorporate learnings from reviews with customers.	November 1, 2017
Initial publication	Cost Optimization Pillar - AWS Well-Architected Framework published.	November 1, 2016

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided “as is” without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

AWS Glossary

For the latest AWS terminology, see the AWS glossary in the *AWS Glossary Reference*.