# Cost Optimization Pillar

## AWS Well-Architected Framework

# Cost Optimization Pillar: AWS Well-Architected Framework

# Table of Contents

# Cost Optimization Pillar - AWS Well-Architected Framework

Publication date: **October 20, 2022** (*Document revisions* (p. 56))

## 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, and cost-effective 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

Cloud Financial Management (CFM) enables organizations to realize business value and financial success as they optimize their cost and usage and scale on AWS.

The following are Cloud Financial Management best practices:

**Best practices**



## COST01-BP01 Establish a cost optimization function

Create a team that is responsible for establishing and maintaining cost awareness across your organization. The team requires people from finance, technology, and business roles across the organization.

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

### Implementation guidance

This function is responsible for establishing and maintaining a culture of cost awareness. It 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-disciplined approach, with capabilities in project management, data science, financial analysis, and software/infrastructure development. The function can improve efficiencies of workloads by executing cost optimizations (centralized approach), influencing technology teams to execute optimizations (decentralized), or a combination of both (hybrid). The function may be measured against their ability to execute and deliver against cost optimization goals (for example, workload efficiency metrics).

You must secure executive sponsorship for this function. The sponsor is regarded as champion for cost efficient cloud consumption, and provides escalation support for the function to ensure that cost optimization activities are treated with the level of priority defined by the organization. Together, the sponsor and function ensure that your organization consumes the cloud efficiently and continue to deliver business value.

**Implementation steps**

- **Define key members:** You need to ensure that all relevant parts of your organization contribute and have a stake 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, technical), others periodically as required.
- **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 executing 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 the organization's cost optimization capability.
- **Establish regular cadence:** 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-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 executing 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.

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. 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.
- **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.

**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:** 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

Customers use the cloud for efficiency, speed and agility, which creates a highly variable amount of cost and usage. Costs can decrease with increases in workload efficiency, or as new workloads and features are deployed. Or, workloads will scale to serve more of your customers, which increases cloud usage and costs. Existing organizational budgeting processes must be modified to incorporate this variability.

Adjust existing budgeting and forecasting processes to become more dynamic using either a trend-based algorithm (using historical costs as inputs), or using business-driver-based algorithms (for example, new product launches or regional expansion), or a combination of both trend and business drivers.

You can use AWS Cost Explorer to forecast daily (up to 3 months) or monthly (up to 12 months) cloud costs based on machine learning algorithms applied to your historical costs (trend based).

**Implementation steps**

- **Update existing budget and forecasting processes:** Implement trend-based, business driver-based, or a combination of both in your budgeting and forecasting processes.

  Perform regular reviews in line with changes in business direction and usage.

## Resources

**Related documents:**

- AWS News Blog

# COST01-BP04 Implement cost awareness in your organizational processes

Implement cost awareness 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 recommended to re-use and modify existing processes where possible—this minimizes the impact to agility and velocity. The following recommendations will help implement cost awareness in your workload:

- Ensure that change management includes a cost measurement to quantify the financial impact of your changes. This helps pro-actively address cost-related concerns and highlight cost savings.
- Ensure 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 cause for cost and usage anomalies (cost overages).
- Accelerate cost savings and business value realization through automation or tooling. When thinking about the cost of implementing, frame the conversation to include an ROI component to justify the investment of time or money.
- Extend existing training and development programs to include cost aware 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.

**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.
- **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.

## Resources

**Related documents:**

- AWS News Blog

# COST01-BP05 Report and notify on cost optimization

Configure AWS Budgets to provide notifications on cost and usage against targets. Have regular meetings to analyze your workload's cost efficiency and to promote cost-aware culture.

**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 cost optimization, or include cost optimization in your regular operational reporting cycles for your workloads. AWS Cost Explorer provides dashboards and reports. You can track your progress of cost and usage against configured budgets with AWS Budgets Reports.

You can also use Amazon QuickSight with Cost and Usage Report (CUR) data, to provide highly customized reporting with more granular data.

Implement notifications on cost and usage to ensure that changes in cost and usage can be acted upon quickly. AWS Budgets allows you to provide notifications against targets. We recommend configuring notifications on both increases and decreases, and in both cost and usage for workloads.

**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, and identify positive trends that you can promote across your organization. Reporting should involve representatives from the application teams and owners, finance, and management.
  - Well-Architected Labs: Visualization

## Resources

**Related documents:**

- AWS News Blog

**Related examples:**

- Well-Architected Labs: Cost and Governance Usage
- Well-Architected Labs: Visualization

# COST01-BP06 Monitor cost proactively

Implement tooling and dashboards to monitor cost proactively for the workload. Do not just look at costs and categories when you receive notifications. This helps to identify positive trends and promote them throughout your organization.

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

## 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.

**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.

## Resources

**Related documents:**

- AWS News Blog

**Related examples:**

- Well-Architected Labs: Advanced Visualization
- Well-Architected Labs: Visualization

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

Consult regularly with experts or APN 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:** Low

## Implementation guidance

You may be able to implement new AWS services and features to increase cost efficiency in your workload. Regularly review the AWS News Blog, the AWS Cost Management blog, and What's New with AWS for information on new service and feature releases.

**Implementation steps**

- **Subscribe to blogs:** Go to the AWS blogs pages and subscribe to the What's New Blog and other relevant blogs.
- **AWS events and meetups:** Attend your local AWS summit, and any local meetups with other organizations from your local area.
- **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, architect, and support team.

## Resources

**Related documents:**

- AWS Blog
- AWS Cost Management
- AWS News Blog

# 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 through best practices that are performed in an organic and decentralized manner across your organization. This creates high levels of capability across your organization with minimal effort, compared to a strict top-down, centralized approach.

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

- 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, 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.
- Create top-down organizational requirements for workloads to run at pre-defined budgets.

**Implementation steps**

- **Subscribe to blogs:** Go to the AWS blogs pages and subscribe to the What's New Blog and other relevant blogs.
- **AWS events and meetups:** Attend your local AWS summit, and any local meetups with other organizations from your local area.
- **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, architect, and support team.

## Resources

**Related documents:**

- AWS Blog
- AWS Cost Management
- AWS News Blog

# 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

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 On-Demand 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 terminated, 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**

- **Executing cost optimization best practices:** For example, resource lifecycle management reduces infrastructure and operational costs and creates time and unexpected budget for experimentation. This increases organization agility and uncovers new opportunities for revenue generation.
- **Implementing automation:** For example, Auto Scaling, which ensures elasticity at minimal effort, and increases staff productivity by eliminating manual capacity planning work. For more details on operational resiliency, refer to the Well-Architected Reliability Pillar whitepaper.
- **Forecasting future AWS costs:** Forecasting enables finance stakeholders to set expectations with other internal and external organization stakeholders, and helps improve your organization's financial predictability. AWS Cost Explorer can be used to perform forecasting for your cost and usage.

# Resources

**Related documents:**

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

# 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

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

Develop policies that define how resources are managed by your organization. Policies should cover cost aspects of resources and workloads, including creation, modification and decommission over the resource 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 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.

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 decommission over the resource's lifetime.

Policies should be simple so that they are easily understood and can be implemented effectively throughout the organization. Start with broad, high-level policies, such as which geographic Region usage is allowed in, 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 or development environments), and which types of resources can be used by different groups (for example, the largest size of resource in a development account is medium).

**Implementation steps**

- **Meet with team members:** To develop policies, get all team members from your organization to specify their requirements and document them accordingly. 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.
- **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:** Ensure 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:**

- AWS Managed Policies for Job Functions
- AWS multiple account billing strategy
- Actions, Resources, and Condition Keys for AWS Services

- Cloud Products
- Control access to AWS Regions using IAM policies
- Global Infrastructures Regions and AZs

# COST02-BP02 Implement goals and targets

Implement both cost and usage goals for your workload. Goals provide direction to your organization on cost and usage, and targets provide measurable outcomes 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. Goals provide guidance and direction to your organization on expected outcomes. Targets provide specific measurable outcomes to be achieved. An example of a goal is: 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 5% increase in costs. Another common goal is that workloads need to be more efficient every 6 months. The accompanying target would be that the cost per output of the workload needs to decrease by 5% every 6 months.

A common goal for cloud workloads is to increase workload efficiency, which is to decrease the cost per business outcome of the workload over time. It is recommended to implement this goal for all workloads, and also set a target such as a 5% increase in efficiency every 6- 12 months. This can be achieved in the cloud through building capability in cost optimization, and through the release of new services and service features.

**Implementation steps**

- **Define expected usage levels:** Focus on usage levels to begin with. Engage with the application owners, marketing, and greater business teams to understand what the expected usage levels will be for the workload. How will customer demand change over time, and will there be any changes due to seasonal increases or marketing campaigns.
- **Define workload resourcing and costs:** With the usage levels defined, quantify the changes in workload resources required to meet these 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 what the costs will be at each of these major points, and what the changes in cost will be when there are changes in usage.
- **Define business goals:** Taking 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, cost and the relation between the two. 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 a goal is to increase efficiency in the workload, the target will quantify the amount of improvement, typical in business outputs for each dollar spent, and when it will be delivered.

## Resources

**Related documents:**

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

# 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 has a one-parent-to-many-children account structure that is commonly known as a management account (the parent, formerly payer) account-member (the child, formerly linked) account. A best practice is to always have at least one management account with one member account, regardless of your organization size or usage. All workload resources should reside only within member accounts.

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 and/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.

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 ensures that access and data requirements are adhered to. Reliability ensures that limits are managed so that environments and workloads do not impact others. Financial constructs ensure that there is strict financial separation 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 a third-party organization.
- **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 ensure that separation requirements are maintained. 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, and 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:**

- AWS managed policies for job functions
- AWS multiple account billing strategy
- Control access to AWS Regions using IAM policies
- AWS Control Tower
- AWS Organizations
- Consolidated billing

**Related examples:**

- Splitting the CUR and Sharing Access

# 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

After you develop policies, you can create logical groups and roles of users within your organization. This allows you to assign permissions and control usage. Begin with high-level groupings of people. Typically this aligns with organizational units and job roles (for example, systems administrator in the IT Department, or financial controller). The groups join people that do similar tasks and need similar access. Roles define what a group must do. 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. Refer to the security pillar for best practices on users, groups, and authentication.
- **Implement roles and policies:** Using the actions defined in your organizational policies, create the required roles and access policies. Refer to the security pillar for best practices on roles and policies.

## Resources

**Related documents:**

- AWS managed policies for job functions
- AWS multiple account billing strategy
- Control access to AWS Regions using IAM policies
- Well-Architected Security Pillar

**Related examples:**

- Well-Architected Lab Basic Identity and Access

# 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: for example, control access to regions or resource types with AWS Identity and Access Management (IAM) policies.

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

## Implementation guidance

A common first step in implementing cost controls is to set up notifications when cost or usage events occur outside of the organization policies. This enables you to 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. In AWS, notifications are conducted with AWS Budgets, which allows you to define a monthly budget 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.

As a second step, you can enforce governance policies in AWS through AWS Identity and Access Management (IAM), 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 and manage AWS resources, the type of resources that can be created, and where they can be created. This minimizes the creation of resources that are not required. 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, ensuring that your accounts stay within your access control guidelines. SCPs are available only in an organization that has all features enabled, and you can configure the SCPs to either deny or allow actions for member accounts by default. Refer to the Well-Architected Security Pillar whitepaper for more details on implementing access management.

Governance can also be implemented through management of 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 provide notifications when spending is outside of your policies. Configure multiple cost budgets, one for each account, which notifies you about overall account spending. Then 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. Ensure that you 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.

- **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 perform. 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

**Related examples:**

- Well-Architected Labs: Cost and Usage Governance
- Well-Architected Labs: Cost and Usage Governance

# 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

Ensure that you track the entire lifecycle of the workload. This ensures that when workloads or workload components are no longer required, they can be decommissioned or modified. This is especially useful when you release new services or features. The existing workloads and components may appear to be in use, but should be decommissioned to redirect customers to the new service. Notice 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.

AWS provides a number of 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.

Refer to the Well-Architected Operational Excellence Pillar whitepaper for more details on implementing entity lifecycle tracking.

**Implementation steps**

- **Perform workload reviews:** As defined by your organizational policies, audit your existing projects. 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:**

- AWS Config
- AWS Systems Manager
- AWS managed policies for job functions
- AWS multiple account billing strategy
- Control access to AWS Regions using IAM policies

# 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

Configure the AWS Cost and Usage Report, and Cost Explorer hourly granularity, to provide detailed cost and usage information. Configure your workload to have log entries for every delivered business outcome.

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

### Implementation guidance

Enable hourly granularity in AWS Cost Explorer and create a AWS Cost and Usage Report (CUR). These data sources provide the most accurate view of cost and usage across your entire organization. The CUR provides daily or hourly usage granularity, rates, costs, and usage attributes for all chargeable AWS services. All possible dimensions are in the CUR including: tagging, location, resource attributes, and account IDs.

Configure your CUR with the following customizations:

- Include resource IDs
- Automatically refresh the CUR
- Hourly granularity
- **Versioning:** Overwrite existing report
- **Data integration:** Amazon Athena (Parquet format and compression)

Use AWS Glue to prepare the data for analysis, and use Amazon Athena to perform data analysis, using SQL to query the data. You can also use Amazon QuickSight to build custom and complex visualizations and distribute them throughout your organization.

**Implementation steps**

- **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:** Using the billing console, enable Hourly and Resource Level Data.

  **Note**
  There will be associated costs with enabling this feature. For details, refer to the pricing.
- **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. Refer to the Well-Architected Operational Excellence Pillar for more detail on logging and monitoring.

## Resources

**Related documents:**

- AWS Account Setup
- AWS Cost and Usage Report (CUR)
- AWS Glue
- Amazon QuickSight
- AWS Cost Management Pricing
- Tagging AWS resources
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports
- Well-Architected Operational Excellence Pillar

**Related examples:**

- AWS Account Setup

# COST03-BP02 Identify cost attribution categories

Identify organization categories that could be used to allocate cost within your organization.

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

## Implementation guidance

Work with your finance team and other relevant stakeholders to understand the requirements of how costs must be allocated within your organization. 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.

**Implementation steps**

- **Define your organization categories:** Meet with stakeholders to define categories that reflect your organization's structure and requirements. These will 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. Multiple categories can be assigned to a resource, and a resource can be in multiple different categories, so define as many categories as needed.
- **Define your functional categories:** Meet with stakeholders 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. Multiple categories can be assigned to a resource, and a resource can be in multiple different categories, so define as many categories as needed.

## 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-BP03 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-BP04 Configure billing and cost management tools

Configure AWS Cost Explorer and AWS Budgets inline with your organization policies.

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

## Implementation guidance

To modify usage and adjust costs, each person in your organization must have access to their cost and usage information. It is recommended that all workloads and teams have the following tooling configured when they use the cloud:

- **Reports:** Summarize of all cost and usage information
- **Notifications:** Provide notifications when cost or usage is outside of defined limits.
- **Current State:** Configure a dashboard showing current levels of cost and usage. The dashboard should be available in a highly visible place within the work environment (similar to an operations dashboard).
- **Trending:** Provide the capability to show the variability in cost and usage over the required period of time, with the required granularity.
- **Forecasts:** Provide the capability to show estimated future costs.
- **Tracking:** Show the current cost and usage against configured goals or targets.
- **Analysis:** Provide the capability for team members to perform custom and deep analysis down to the hourly granularity, with all possible dimensions.

You can use AWS native tooling, such as AWS Cost Explorer, AWS Budgets, and Amazon Athena with Amazon QuickSight to provide this capability. You can also use third-party tooling — however, you must ensure that the costs of this tooling provide value to your organization.

**Implementation steps**

- **Create a Cost Optimization group:** Configure your account and create a group that has access to the required Cost and Usage reports. This group 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.
- **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.
- **Configure AWS Cost Explorer:** Configure AWS Cost Explorer for your workload and accounts. Create a dashboard for the workload that tracks overall spend, and key usage metrics for the workload.
- **Configure advanced tooling:** Optionally, you can create custom tooling for your organization that provides additional detail and granularity. You can implement advanced analysis capability using Amazon Athena, and dashboards using Amazon QuickSight.

## 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:**

- Well-Architected Labs - AWS Account Setup
- Well-Architected Labs: Billing Visualization
- Well-Architected Labs: Cost and Governance Usage
- Well-Architected Labs: Cost and Usage Analysis
- Well-Architected Labs: Cost and Usage Visualization

# COST03-BP05 Add organization information to cost and usage

Define a tagging schema based on organization, and workload attributes, and cost allocation categories. Implement tagging across all resources. Use Cost Categories to group costs and usage according to organization attributes.

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

## 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 enables you to 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.

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 Systems Manager, to increase efficiency.
- **Implement 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 it 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 the features within the service, or services such as AWS CloudFormation, to ensure that resources are tagged when created. You can also create a custom 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:**

- AWS CloudFormation Resource Tag
- AWS Cost Categories
- Tagging AWS resources
- Amazon EC2 and Amazon EBS add support for tagging resources upon creation
- Analyzing your costs with AWS Budgets
- Analyzing your costs with Cost Explorer
- Managing AWS Cost and Usage Reports

# COST03-BP06 Allocate costs based on workload metrics

Allocate the workload's costs by metrics or business outcomes to measure workload cost efficiency. Implement a process to analyze the AWS Cost and Usage Report with Amazon Athena, which can provide insight and charge back capability.

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

## Implementation guidance

Cost Optimization is delivering business outcomes at the lowest price point, which can only be achieved by allocating workload costs by workload metrics (measured by workload efficiency). Monitor the defined workload metrics through log files or other application monitoring. Combine this data with the workload costs, which can be obtained by looking at costs with a specific tag value or account ID. It is recommended to perform this analysis at the hourly level. Your efficiency will typically change if you have some static cost components (for example, a backend database running 24/7) with a varying request rate (for example, usage peaks at 9am – 5pm, with few requests at night). Understanding the relationship between the static and variable costs will help you to focus your optimization activities.

**Implementation Steps**

- **Allocate costs to workload metrics:** Using the defined metrics and tagging configured, create a metric that combines the workload output and workload cost. Use the 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

# 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 (p. 27)
- COST04-BP02 Implement a decommissioning process (p. 28)
- COST04-BP03 Decommission resources (p. 28)
- COST04-BP04 Decommission resources automatically (p. 29)

# 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) will help you identify assets for decommission. 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.

**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.
- **Implement workload throughput or output monitoring:** Implement workload throughput monitoring or alarming, triggering 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.

## Resources

**Related documents:**

- AWS Auto Scaling
- AWS Trusted Advisor
- Tagging AWS resources
- Publishing Custom Metrics

# COST04-BP02 Implement a decommissioning process

Implement a process to identify and decommission orphaned 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 ensure that all organization requirements are met.

**Implementation steps**

- **Create and implement a decommissioning process:** Working with the workload developers and owners, 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. The process should also cover the steps necessary to decommission the resource, removing them from service while ensuring compliance with any regulatory requirements. Any associated resources are also covered, such as licenses or attached storage. The process should provide notification to the workload owners that the decommissioning process has been executed.

## Resources

**Related documents:**

- AWS Auto Scaling
- AWS Trusted Advisor

# COST04-BP03 Decommission resources

Decommission resources triggered by events such as periodic audits, or changes in usage. Decommissioning is typically performed periodically, and is 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 triggered by state changes in the workload, such as a product going end of life or being replaced. Searches and decommission events may also be triggered by external events, such as changes in market conditions or product termination.

**Implementation steps**

- **Decommission resources:** Using the decommissioning process, decommission each of the resources that have been identified as orphaned.

## 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 AWS Auto Scaling to perform the decommissioning process. You can also implement custom code using the API or SDK to decommission workload resources automatically.

**Implementation steps**

- **Implement AWS Auto Scaling:** For resources that are supported, configure them with AWS Auto Scaling.
- **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 (Amazon EC2) 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.

## Resources

**Related documents:**

- AWS Auto Scaling
- AWS Trusted Advisor
- Create Alarms to Stop, Terminate, Reboot, or Recover an Instance
- Getting Started with Amazon EC2 Auto Scaling

# 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

When selecting services for your workload, it is key that you understand your organization priorities. Ensure that you have a balance between cost and other Well-Architected pillars, such as performance and reliability. 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 within your organization to collect information, such as product, business, technical, and finance.

**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 is a list of the pillars in order. You can also add a weighting to each, which can indicate how much additional focus a pillar has, or how similar the focus is between two pillars.

## Resources

**Related documents:**

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

# COST05-BP02 Analyze all components of this 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:** Low

## Implementation guidance

Perform a thorough analysis on all components in your workload. Ensure that balance between the cost of analysis and the potential savings in the workload over its lifecycle. You must find the current impact, and potential future impact, of the component. For example, if the cost of the proposed resource is $10 a month, and under forecasted loads would not exceed $15 a month, spending a day of effort to reduce costs by 50% ($5 a month) could exceed the potential benefit over the life of the system. Using a faster and more efficient data-based estimation will 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.

AWS Cost Explorer and the AWS Cost and Usage Report (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.

**Implementation steps**

- **List the workload components:** Build the list of all the workload components. This is used as verification to check that each component was analyzed. The effort spent should reflect the criticality to the workload as defined by your organization's priorities. Grouping together resources functionally improves efficiency, for example production database storage, if there are multiple databases.
- **Prioritize 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 chose the option that aligns best with your organizational priorities.

## Resources

**Related documents:**

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

# COST05-BP03 Perform a thorough analysis of each component

Look at overall cost to the organization of each component. Look at total cost of ownership by factoring in cost of operations and management, especially when using managed services. 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:** Low

## Implementation guidance

Consider the time savings that will allow your team to focus on retiring technical debt, innovation, and value-adding features. For example, you might need to lift and shift your on-premises environment to the cloud as rapidly as possible and optimize later. It is worth exploring the savings you could realize by using managed services that remove or reduce license costs. Managed services remove the operational and administrative burden of maintaining a service, which allows you to focus on innovation. Additionally, because managed services operate at cloud scale, they can offer a lower cost per transaction or service.

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 enable 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.

## Resources

**Related documents:**

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

# 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

The cost of software licenses can be eliminated 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 ensure that you have the most optimized 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.

**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 will align the cost of the software to the benefit 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 execute audits, and penalties that could be imposed.

## Resources

**Related documents:**

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

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

Factor in cost 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, Amazon Simple

Storage Service (Amazon S3)for static websites, and Amazon Elastic Container Service (Amazon ECS). 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.

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

## Implementation guidance

You can use serverless or application-level services such as AWS 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.

For more information on Serverless, refer to the 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.

## Resources

**Related documents:**

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

# 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:** Low

## 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 trigger for review is change in usage patterns. Significant changes in usage can indicate that alternate services would be more optimal. For example, for higher data transfer rates a direct connect service may be cheaper than a VPN, and provide the required connectivity. Predict the potential impact of service changes, so you can monitor for these usage level triggers 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.

- **Perform cost analysis at predicted usage:** Using the usage patterns defined, perform the 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.

## Resources

**Related documents:**

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

# 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, triggered 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 (p. 35)
- COST06-BP02 Select resource type, size, and number based on data (p. 36)
- COST06-BP03 Select resource type, size, and number automatically based on metrics (p. 37)

# COST06-BP01 Perform cost modeling

Identify organization requirements and perform cost modeling 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 modeling 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. 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 whitepaper.

AWS Compute Optimizer can assist with cost modelling for running workloads. It provides right-sizing recommendations for compute resources based on historical usage. This is the ideal data source for compute resources because it is a free service, and it utilizes machine learning to make multiple recommendations depending on levels of risk. You can also use Amazon CloudWatch and Amazon CloudWatch Logs with custom logs as data sources for right sizing operations for other services and workload components.

The following are recommendations for cost modelling data and metrics:

- The monitoring must accurately reflect the end-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.

**Implementation steps**

- **Perform cost modeling:** 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. Then redeploy the workload or change the resource types and sizes and re-run the test.

## Resources

**Related documents:**

- AWS Auto Scaling
- Amazon CloudWatch features
- Cost Optimization: Amazon EC2 Right Sizing
- AWS Compute Optimizer

# 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

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, published solutions).

**Implementation steps**

- **Select resources based on data:** Using your cost modeling data, select the expected workload usage level, then select the specified resource type and size.

## Resources

**Related documents:**

- AWS Auto Scaling
- Amazon CloudWatch features

- Cost Optimization: EC2 Right Sizing

# 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 services such as Amazon Elastic Compute Cloud (Amazon EC2), Amazon DynamoDB, Amazon Elastic Block Store (Amazon EBS) (PIOPS), Amazon Relational Database Service (Amazon RDS), Amazon EMR, and networking. 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 Elastic Compute Cloud(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(Amazon S3) 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**

- **Configure workload metrics:** Ensure you capture the key metrics for the workload. These metrics provide an indication of the customer experience, such as the workload output, and align to the differences between resource types and sizes, such as CPU and memory usage.
- **View rightsizing recommendations:** Use the rightsizing recommendations in AWS Compute Optimizer to make adjustments to your workload.
- **Select resource type and size automatically based on metrics:** Using the workload metrics, manually or automatically select your workload 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.

## Resources

**Related documents:**

- AWS Auto Scaling
- AWS Compute Optimizer
- Amazon CloudWatch features
- CloudWatch Getting Set Up
- CloudWatch Publishing Custom Metrics
- Cost Optimization: Amazon EC2 Right Sizing
- Getting Started with Amazon EC2 Auto Scaling
- Amazon S3 Intelligent-Tiering
- Launch an EC2 Instance Using the SDK

# Select the best pricing model

**Best practices**



## 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 Instances). Perform an analysis on the workload using the Recommendations feature in AWS Cost Explorer.

**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.

**Implementation steps**

- **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.
- **Analyze workload elasticity:** Using the hourly granularity in Cost Explorer, or a custom dashboard. Analyze the workload 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

### Resources

**Related documents:**

- Accessing Reserved Instance recommendations
- Instance purchasing options

**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 Implement Regions based on cost

Resource pricing can be different in each Region. Factoring in Region cost helps ensure that you pay the lowest overall price for this workload.

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

## Implementation guidance

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 Pricing Calculator to estimate the costs of your workload in various Regions.

**Implementation steps**

- **Review 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.

## 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

# 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

When you utilize 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 Elastic Compute Cloud(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.

**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

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.

**Implementation steps**

- **Implement pricing models:** Using your analysis results, purchase Savings Plans (SPs), Reserved Instances (RIs) or implement Spot Instances. If it is your first RI purchase then choose the top 5 or

10 recommendations in the list, then monitor and analyze the results over the next month or two. Purchase small numbers of commitment discounts regular cycles, for example every two weeks or monthly. Implement Spot Instances for workloads that can be interrupted or are stateless.

- **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 every two to four weeks, or as your organization usage changes.

## Resources

**Related documents:**

- Accessing Reserved Instance recommendations
- EC2 Fleet
- How to Purchase Reserved Instances
- Instance purchasing options
- Spot Instances

**Related videos:**

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

# COST07-BP05 Perform pricing model analysis at the master account level

Use Cost Explorer Savings Plans and Reserved Instance recommendations to perform regular analysis at the management account level for commitment discounts.

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

## Implementation guidance

Performing regular cost modeling ensures that opportunities to optimize across multiple workloads can be implemented. 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 will 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.

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

**Implementation steps**

- **Perform a commitment discount analysis:** Using Cost Explorer in your account review the Savings Plans and Reserved Instance recommendations. To verify 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

**Related videos:**

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

**Related examples:**

- Well-Architected Lab: Pricing Models

# 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 (p. 42)
- COST08-BP02 Select components to optimize data transfer cost (p. 43)
- COST08-BP03 Implement services to reduce data transfer costs (p. 43)

## 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

Understand where the data transfer occurs in your workload, the cost of the transfer, and its associated benefit. 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 cost of structure and decide that this is an acceptable cost (similar to paying for compute and storage in both Availability Zone) 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.

Use AWS Cost Explorer or the 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**

- **Calculate data transfer costs:** Use the AWS pricing pages and 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.

## Resources

**Related documents:**

- AWS caching solutions
- AWS Pricing
- Amazon EC2 Pricing
- Amazon VPC pricing
- Deliver content faster with Amazon CloudFront

# 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:** Low

## Implementation guidance

Architecting for data transfer ensures that you minimize 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.

**Implementation steps**

- **Select components for data transfer:** Using the 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 documents:**

- AWS caching solutions
- Deliver content faster with Amazon CloudFront

# COST08-BP03 Implement services to reduce data transfer costs

Implement services to reduce data transfer. For example, using a content delivery network (CDN) such as Amazon CloudFront to deliver content to end users, caching layers using Amazon ElastiCache, or using AWS Direct Connect instead of VPN for connectivity to AWS.

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

## Implementation guidance

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.

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 quick and easy 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 gateways costs. Gateway VPC endpoints have no hourly charges, and support Amazon Simple Storage Service(Amazon S3) and Amazon DynamoDB. Interface VPC endpoints are provided by AWS PrivateLink and have an hourly fee and per GB usage cost.

**Implementation steps**

- **Implement services:** Using the data transfer modeling, 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
- Deliver content faster with Amazon CloudFront

# 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 (p. 45)
- COST09-BP02 Implement a buffer or throttle to manage demand (p. 46)
- COST09-BP03 Supply resources dynamically (p. 47)

# 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

Know the requirements of the workload. The organization 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 will 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. Ensure that the analysis is performed over a long enough period to incorporate any seasonal variance, such as end-of- month processing or holiday peaks.

Ensure that the analysis effort reflects 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 lifetime.

You can use AWS Cost Explorer or Amazon QuickSight with the AWS Cost and Usage Report (CUR) or your application logs to perform a visual analysis of workload demand.

**Implementation steps**

- **Analyze existing workload data:** Analyze data from the existing workload, previous versions of the workload, or predicted usage patterns. Use log files and monitoring data to gain insight on how customers use the workload. Typical metrics are the actual demand in requests per second, the times when the rate of demand changes or when it is at different levels, and the rate of change of demand. Ensure you analyze a full cycle of the workload, ensuring you 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. Common metrics for value are risk, brand awareness, revenue or workload cost.
- **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:**

- AWS Auto Scaling
- AWS Instance Scheduler
- Getting started with Amazon SQS
- AWS Cost Explorer
- Amazon QuickSight

# 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:** Low

## Implementation guidance

**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 will wait for a period of time and then re-try 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. Refer to the Well-Architected Reliability pillar whitepaper for more details on implementing throttling.

**Buffer based:** Similar to throttling, a buffer defers request processing, allowing applications that run at different rates to communicate effectively. A buffer-based approach uses a queue to accept messages (units of work) from producers. Messages are read by consumers and processed, allowing the messages to run at the rate that meets the consumers' business requirements. You don't have to worry about producers having to deal with throttling issues, such as data durability and backpressure (where producers slow down because their consumer is running slowly).

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.

When architecting with a buffer-based approach, ensure that you architect your workload to service the request in the required time, and that you are able to handle duplicate requests for work.

**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 will 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 documents:**

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

# 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 or under-provisioning.

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

## Implementation guidance

You can use AWS Auto Scaling, or incorporate scaling in your code with the AWS API or SDKs. This reduces your overall workload costs by removing the operational cost from manually making changes to your environment, and can be performed much faster. This will ensure that the workload resourcing best matches the demand at any time.

**Demand-based supply:** Leverage the elasticity of the cloud to supply resources to meet changing demand. Take advantage of APIs or service features to programmatically vary the amount of cloud resources in your architecture dynamically. This allows you to scale components in your architecture, and automatically increase the number of resources during demand spikes to maintain performance, and decrease capacity when demand subsides to reduce costs.

AWS Auto Scaling helps you adjust your capacity to maintain steady, predictable performance at the lowest possible cost. It is a fully managed and free service that integrates with Amazon Elastic Compute

Cloud (Amazon EC2) instances and Spot Fleets, Amazon Elastic Container Service (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.

Auto Scaling can implement manual, scheduled or demand-based scaling. You can also use metrics and alarms from Amazon CloudWatch to trigger scaling events for your workload. Typical metrics can be standard Amazon EC2 metrics, such as CPU utilization, network throughput, and Elastic Load Balancing(ELB) observed request or response latency. When possible, you should use a metric that is indicative of customer experience, which is typically a custom metric that might originate from application code within your workload.

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.

ELB helps you to scale by distributing demand across multiple resources. As you implement more resources, you add them to the load balancer to take on the demand. Elastic Load Balancing has support for Amazon EC2 Instances, containers, IP addresses, and AWS Lambda functions.

**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.

You can use scheduled 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) thus ensuring that resources are available when users or demand arrives.

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 Elastic Block Store (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 time-based scheduling:** 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.
- **Configure Auto Scaling:** To configure scaling based on active workload metrics, use Amazon Auto Scaling. Use the analysis and configure auto scaling to trigger on the correct resource levels, and ensure that the workload scales in the required time.

# Resources

**Related documents:**

- AWS Auto Scaling
- AWS Instance Scheduler
- Getting Started with Amazon EC2 Auto Scaling
- Getting started with Amazon SQS
- Scheduled Scaling for Amazon EC2 Auto Scaling

# 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.

**Best practices**

- COST10-BP01 Develop a workload review process (p. 50)
- COST10-BP02 Review and analyze this workload regularly (p. 51)

# 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 10% of the bill are reviewed quarterly, while workloads below 10% are reviewed annually.

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

## Implementation guidance

To ensure that you always 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 ensure that you 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 5% 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.

**Implementation steps**

- **Define review frequency:** Define how frequently the workload and its components should be reviewed. This is a combination of factors and may differ from workload to workload within your

organization, it may also differ 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 6 months, the database every 12 months, compute and short-term storage every 6 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. You may decide to spend one week of analysis on the database component, and four hours for storage reviews.

## Resources

**Related documents:**

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

# COST10-BP02 Review and analyze this workload regularly

Existing workloads are regularly reviewed based on for each defined processes.

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

## Implementation guidance

To realize the benefits of new AWS services and features, you must execute the review process on your workloads and implement new services and features as required. For example, you might review your workloads and replace the messaging component with Amazon Simple Email Service (Amazon SES). This removes the cost of operating and maintaining a fleet of instances, while providing all the functionality at a reduced cost.

**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
- Types of Cloud Computing
- What's New with AWS

# 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:

- Ben Mergen, Cost Optimization Pillar Lead, Well-Architected, Amazon Web Services
- Levon Stepanian, BDM, Cloud Financial Management, Amazon Web Services
- Keith Jarrett, Business Development Lead – Cost Optimization, Amazon Web Services
- PT Ng, Commercial Architect, 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 |
|---|---|---|
| Whitepaper updated (p. 56) | Best practices expanded and improvement plans added. | October 20, 2022 |
| Minor update (p. 1) | Added Sustainability Pillar to introduction. | December 2, 2021 |
| Minor update (p. 56) | Updated links. | April 25, 2021 |
| Minor update (p. 56) | Updated links. | March 10, 2021 |
| Updates for new Framework (p. 56) | Updated to incorporate CFM, new services, and integration with the Well-Architected too. | July 8, 2020 |
| Whitepaper updated (p. 56) | Updated to reflect changes to AWS and incorporate learnings from reviews with customers. | July 1, 2018 |
| Whitepaper updated (p. 56) | Updated to reflect changes to AWS and incorporate learnings from reviews with customers. | November 1, 2017 |
| Initial publication (p. 56) | Cost Optimization Pillar - AWS Well-Architected Framework published. | November 1, 2016 |