

Organizing partner:

BLOCK71  
COMMUNITY  
By NUS Enterprise  
In partnership with Sinar Group

INNOVATION  
FACTORY



# AWS Cost Monitoring & Optimization



Speaker

**Dwi Fahni Denni**  
AWS Community Builder  
& DevOps Lead - ZebraX

Tuesday, 23 November 2021  
19.15 - 21.00 WIB

Register:

[bit.ly/UGMedan-CostandMonitoring](https://bit.ly/UGMedan-CostandMonitoring)

Organizing partner:



Speaker

**Dwi Fahni Denni**  
AWS Community Builder  
& DevOps Lead - ZebraX



<https://www.linkedin.com/in/dfdenni>



<https://github.com/zeroc0d3>  
<https://github.com/ZXDfdenni>

## About Speaker

### Coffee Addict

- **Carrier Path**
  - Software Engineer (2009-2015)
  - System Administrator (2015-2017)
  - DevOps Engineer (2017 – now)
  - DevOps Engineer Lead @ ZebraX (2021 – now)
  - AWS Community Builder, Container Categories (2021)
- **Education**
  - Electronics Engineer  
Institut Teknologi Nasional (ITENAS), Bandung
- **Hobby**
  - Gaming (Minecraft & Dota 2)
  - Sport (Archery)

- Sport (Archery)
- Gaming (Minecraft & Dota 2)
- Hobby



## ABOUT US

ZebraX is a **digital transformation company** that combines world class digital technology expertise with in-field experience and local know how. It is an industrial tech company established by Indika Energy, Indonesia's leading integrated energy company, with robust partnership network in digital connectivity, automation, and deep learning analytics.

## OUR MISSION

By **bridging the gap between data and business impact**, ZebraX aims to help clients effectively harness digital technology to **bring their operations to the next level**.



## Our Key Capabilities

# Digital core capabilities to provide the best result in enabling Industry 4.0 journey

## Data Science and Industrial IoT

Delivering enterprise-scale machine learning and advanced analytics applications and enabling automated data capture using specifically designed industrial grade IoT devices.

## Digital Strategy and Business Consulting

Enabling Industry 4.0 by identifying business value through value-at-stake analysis, digital implementation road-mapping, and mobilizing suitable IoT strategy.

## Product & Technology Development

Building superior digital products by top-tier tech architect, developers, and UI/UX designers with both vast and deep knowledge in newest technology platforms.



## Advanced Analytics Services

### DATA ACQUISITION & PROCESSING

- Inspection data
- Legacy sensors (PLC, DCS)
- IoT sensors
- Maintenance data

### ANALYTICS & MACHINE LEARNING

- Modelling
- Historical data analysis
- Domain expert

### VISUALIZATION & DECISION MAKING

- Predictions & simulations
- Data visualization
- Real-time data update

“Deliver data-driven, intelligent insight for performance optimization”

Get Connected with Us!

Organizing partner:

BLOCK71 COMMUNITY  
By Mili Enterprise  
in partnership with Salim Group

INNOVATION  
FACTORY



## Contents

- Introduction AWS Well-Architected Framework (WAF)
- AWS Well-Architected Cost Optimization
- Monitoring & Alerting Cost
- Strategy of Cost Optimization
- Q & A

## Requirements

- **Authorize:** AWS Console Access & IAM Policy (Billing)
- **Container Registry:** ECR / DockerHub
- **CLI:** aws, docker, docker-compose, terraform
- **Linux:** Basic Linux Command

## Excludes

- **Dashboards:**
  - Amazon Quicksight Dashboard
  - Cloud Intelligence Dashboard (CID) – formerly: CUDOS Framework Workshop
  - Trusted Advisor Organizational (TAO) Dashboard
- **CI/CD Tools:**
  - AWS CodePipeline, etc (for AWS Agent Deployment)

Organizing partner:

BLOCK71  
COMMUNITY  
By Mili Enterprise  
in partnership with Salton Group

INNOVATION  
FACTORY



## Contents

### Introduction AWS Well-Architected Framework (WAF)

AWS Well-Architected Cost Optimization

Monitoring & Alerting Cost

Strategy of Cost Optimization

Q & A

Introduction to

# AWS Well-Architected Framework

The AWS Well-Architected Framework helps you understand the pros and cons of decisions you make while building systems on AWS. By using the Framework you will learn architectural best practices for designing and operating reliable, secure, efficient, and cost-effective systems in the cloud.

It provides a way for you to consistently measure your architectures against best practices and identify areas for improvement. The process for reviewing an architecture is a constructive conversation about architectural decisions, and is not an audit mechanism. We believe that having well-architected systems greatly increases the likelihood of business success.

**Table 1. The pillars of the AWS Well-Architected Framework**

Name	Description
Operational Excellence	The ability to support development and run workloads effectively, gain insight into their operations, and to continuously improve supporting processes and procedures to deliver business value.
Security	The security pillar describes how to take advantage of cloud technologies to protect data, systems, and assets in a way that can improve your security posture.
Reliability	The reliability pillar encompasses the ability of a workload to perform its intended function correctly and consistently when it's expected to. This includes the ability to operate and test the workload through its total lifecycle. This paper provides in-depth, best practice guidance for implementing reliable workloads on AWS.
Performance Efficiency	The ability to use computing resources efficiently to meet system requirements, and to maintain that efficiency as demand changes and technologies evolve.
Cost Optimization	The ability to run systems to deliver business value at the lowest price point.

Well-Architected

## AWS Well-Architected Framework

Defining five pillars; Operational Excellence, Security, Reliability, Performance Efficiency and Cost Optimization.

- **The AWS Well-Architected Framework** describes the key concepts, design principles, and architectural best practices for designing and running workloads in the cloud. By answering a set of foundational questions, you learn how well your architecture aligns with cloud best practices and are provided guidance for making improvements.
- **AWS Well-Architected** helps cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications and workloads. Based on five pillars — operational excellence, security, reliability, performance efficiency, and cost optimization — AWS Well-Architected provides a consistent approach for customers and partners to evaluate architectures, and implement designs that can scale over time.
- **References:**
  - [AWS Well Architected Framework](#)
  - [Overview: AWS Well Architected Framework](#)
  - [Appendix: Questions and Best Practices](#)

# AWS Well-Architected Framework

Five Pillars

## AWS Well Architected Framework



### AWS Well-Architected Framework

Defining five pillars; Operational Excellence, Security, Reliability, Performance Efficiency and Cost Optimization.

- The **AWS Well-Architected Framework** describes the key concepts, design principles, and architectural best practices for designing and running workloads in the cloud. By answering a set of foundational questions, you learn how well your architecture aligns with cloud best practices and are provided guidance for making improvements.
- **AWS Well-Architected** helps cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications and workloads. Based on five pillars — operational excellence, security, reliability, performance efficiency, and cost optimization — AWS Well-Architected provides a consistent approach for customers and partners to evaluate architectures, and implement designs that can scale over time.
- **References:**
  - [AWS Well Architected Framework](#)
  - [Overview: AWS Well Architected Framework](#)
  - [Appendix: Questions and Best Practices](#)

Organizing partner:

BLOCK71 COMMUNITY  
By Mili Enterprise  
in partnership with Salim Group

INNOVATION  
FACTORY



## Contents

Introduction AWS Well-Architected Framework (WAF)

AWS Well-Architected Cost Optimization

Monitoring & Alerting Cost

Strategy of Cost Optimization

Q & A

# Practice Cloud Financial Management -1-

## Best Practices:

- **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.
- **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.
- **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.

Cloud Financial

## Cost 1: How do you implement cloud financial management?

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

- Establish a cost optimization function
- Establish a partnership between finance and technology
- Establish cloud budget and forecasts
- Implement cost awareness in your organizational process
- Report and notify on cost optimization
- Monitor cost proactively
- Keep up to date with new service releases
- **References:**
  - [Practice Cloud Financial Management](#)

# Practice Cloud Financial Management -2-

## Best Practices:

- **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.
- **Report and notify on cost optimization:** Configure AWS Budgets to provide notifications on cost and usage against targets. Have regular meetings to analyze this workload's cost efficiency and to promote cost aware culture.
- **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.
- **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.

Cloud Financial

## Cost 1: How do you implement cloud financial management?

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

- Establish a cost optimization function
- Establish a partnership between finance and technology
- Establish cloud budget and forecasts
- Implement cost awareness in your organizational process
- Report and notify on cost optimization
- Monitor cost proactively
- Keep up to date with new service releases
- References:
  - [Practice Cloud Financial Management](#)

# Expenditure and Usage Awareness -1-

## Best Practices:

- **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.
- **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.
- **Implement an account structure:** Implement a structure of accounts that maps to your organization. This assists in allocating and managing costs throughout your organization.
- **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.

Govern Usage

## Cost 2: How do you govern usage?

Establish policies and mechanisms to ensure that appropriate costs are incurred while objectives are achieved. By employing a checks-and-balances approach, you can innovate without overspending.

- Develop policies based on your organization requirements
- Implement goals and targets
- Implement an account structure
- Implement groups and roles
- Implement cost controls
- Track project lifecycle
- References:
  - [Expenditure and Usage Awareness](#)

# Expenditure and Usage Awareness -2-

In Collaboration with:



## Best Practices:

- **Implement cost controls:** Implement controls based on organization policies and defined groups and roles. These ensure that costs are only incurred as defined by organization requirements: for example, control access to regions or resource types with IAM policies.
- **Track project lifecycle:** Track, measure, and audit the lifecycle of projects, teams, and environments to avoid using and paying for unnecessary resources.

Govern Usage

## Cost 2: How do you govern usage?

Establish policies and mechanisms to ensure that appropriate costs are incurred while objectives are achieved. By employing a checks-and-balances approach, you can innovate without overspending.

- Develop policies based on your organization requirements
- Implement goals and targets
- Implement an account structure
- Implement groups and roles
- Implement cost controls
- Track project lifecycle
- References:
  - [Expenditure and Usage Awareness](#)

# Expenditure and Usage Awareness -3-

## Best Practices:

- **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.
- **Identify cost attribution categories:** Identify organization categories that could be used to allocate cost within your organization.
- **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.
- **Configure billing and cost management tools:** Configure AWS Cost Explorer and AWS Budgets inline with your organization policies.

Monitoring Usage

## Cost 3: How do you monitor usage and cost?

Establish policies and procedures to monitor and appropriately allocate your costs. This allows you to measure and improve the cost efficiency of this workload.

- Configure detailed information sources
- Identify cost attribution categories
- Establish organization metrics
- Configure billing and cost management tools
- Add organization information to cost and usage
- Allocate costs based on workload metrics
- References:
  - [Expenditure and Usage Awareness](#)

# Expenditure and Usage Awareness -4-

## Best Practices:

- **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.
- **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.

Monitoring Usage

## Cost 3: How do you monitor usage and cost?

Establish policies and procedures to monitor and appropriately allocate your costs. This allows you to measure and improve the cost efficiency of this workload.

- Configure detailed information sources
- Identify cost attribution categories
- Establish organization metrics
- Configure billing and cost management tools
- Add organization information to cost and usage
- Allocate costs based on workload metrics
- References:
  - [Expenditure and Usage Awareness](#)

# Expenditure and Usage Awareness -5-

## Best Practices:

- **Track resources over their life time:** Define and implement a method to track resources and their associations with systems over their life time. You can use tagging to identify the workload or function of the resource.
- **Implement a decommissioning process:** Implement a process to identify and decommission orphaned resources.
- **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.
- **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.

Decommissions

## Cost 4: How do you decommissions resources?

Implement change control and resource management from project inception to end-of-life. This ensures you shut down or terminate unused resources to reduce waste.

- Track resources over their life time
- Implement a decommissioning process
- Decommission resources
- Decommission resources automatically
- References:
  - [Expenditure and Usage Awareness](#)

# Cost Effective Resources -1-

In Collaboration with:



## Best Practices:

- **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.
- **Analyze all components of this workload:** Ensure every workload component is analyzed, regardless of current size or current costs. Review effort should reflect potential benefit, such as current and projected costs.
- **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. Review effort should reflect potential benefit: for example, time spent analyzing is proportional to component cost.
- **Select software with cost effective licensing:** Open source software will eliminate 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.

Evaluate Cost

## Cost 5: How do you evaluate cost when you select services?

Amazon EC2, Amazon EBS, and Amazon S3 are building-block AWS services. Managed services, such as Amazon RDS and Amazon DynamoDB, are higher level, or application level, AWS services.

- By selecting the appropriate building blocks and managed services, you can optimize this workload for cost. For example, using managed services, you can reduce or remove much of your administrative and operational overhead, freeing you to work on applications and business-related activities.
- Identify organization requirements for cost
- Analyze all components of this workload
- Perform a thorough analysis of each component
- Select software with cost effective licensing
- Select components of this workload to optimize cost in line with organization priorities
- Perform cost analysis for different usage over time
- References:
  - [Cost Effective Resources](#)

# Cost Effective Resources -2-

In Collaboration with:



## Best Practices:

- **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 RDS, Amazon DynamoDB, Amazon SNS, and Amazon SES to reduce overall organization cost. Use serverless and containers for compute, such as AWS Lambda, Amazon S3 for static websites, and 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.
- **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, you ensure the workload remains cost effective over its lifetime.

Evaluate Cost

## Cost 5: How do you evaluate cost when you select services?

Amazon EC2, Amazon EBS, and Amazon S3 are building-block AWS services. Managed services, such as Amazon RDS and Amazon DynamoDB, are higher level, or application level, AWS services.

- By selecting the appropriate building blocks and managed services, you can optimize this workload for cost. For example, using managed services, you can reduce or remove much of your administrative and operational overhead, freeing you to work on applications and business-related activities.
- Identify organization requirements for cost
- Analyze all components of this workload
- Perform a thorough analysis of each component
- Select software with cost effective licensing
- Select components of this workload to optimize cost in line with organization priorities
- Perform cost analysis for different usage over time
- References:
  - [Cost Effective Resources](#)

# Cost Effective Resources -3-

## Best Practices:

- **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 potential benefit: for example, time spent is proportional to component cost.
- **Select resource type and size 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 version of the workload (such as an on-premises version), using documentation, or using other sources of information about the workload.
- **Select resource type and size 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 EC2, Amazon DynamoDB, Amazon EBS (PIOPS), 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.

Target Cost

## Cost 6: How do you meet cost targets when you select resource type and number ?

Ensure that you choose the appropriate resource size and number of resources for the task at hand. You minimize waste by selecting the most cost effective type, size, and number.

- Perform cost modeling
- Select resource type and size based on data
- Select resource type and size automatically based on metrics
- References:
  - [Cost Effective Resources](#)

# Cost Effective Resources -4-

In Collaboration with:



## Best Practices:

- **Perform pricing model analysis:** Analyze each component of the workload. Determine if the component and resources will be running for extended periods (for commitment discounts), or dynamic and short running (for spot or on-demand). Perform an analysis on the workload using the Recommendations feature in AWS Cost Explorer.
- **Implement regions based on cost:** Resource pricing can be different in each region. Factoring in region cost ensures you pay the lowest overall price for this workload
- **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.

Pricing Models

## Cost 7: How do you use pricing models to reduce cost ?

Use the pricing model that is most appropriate for your resources to minimize expense.

- Perform pricing model analysis
- Implement regions based on cost
- Select third party agreements with cost efficient terms
- Implement pricing models for all components of this workload
- Perform pricing model analysis at the master account level
- **References:**
  - [Cost Effective Resources](#)

# Cost Effective Resources -5-

## Best Practices:

- **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 is 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.
- **Perform pricing model analysis at the master account level:** Use Cost Explorer Savings Plans and Reserved Instance recommendations to perform regular analysis at the master account level for commitment discounts.

Pricing Models

## Cost 7: How do you use pricing models to reduce cost ?

Use the pricing model that is most appropriate for your resources to minimize expense.

- Perform pricing model analysis
- Implement regions based on cost
- Select third party agreements with cost efficient terms
- Implement pricing models for all components of this workload
- Perform pricing model analysis at the master account level
- **References:**
  - [Cost Effective Resources](#)

# Cost Effective Resources -6-

## Best Practices:

- **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.
- **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 WAN optimization and Multi-AZ configurations
- **Implement services to reduce data transfer costs:** Implement services to reduce data transfer: for example, using a 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.

Data Transfer

## Cost 8: How do you for data transfer charges ?

Ensure that you plan and monitor data transfer charges so that you can make architectural decisions to minimize costs. A small yet effective architectural change can drastically reduce your operational costs over time

- Perform data transfer modeling
- Select components to optimize data transfer cost
- Select third party agreements with cost efficient terms
- Implement services to reduce data transfer costs
- References:
  - [Cost Effective Resources](#)

# Manage Demand and Supply Resources

In Collaboration with:



## Best Practices:

- **Perform an analysis on the workload demand:** Analyze the demand of the workload over time. Ensure the analysis covers seasonal trends and accurately represents operating conditions over the full workload lifetime. Analysis effort should reflect potential benefit: for example, time spent is proportional to the workload cost.
- **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. Ensure your throttles and buffers are designed so clients receive a response in the required time.
- **Supply resources dynamically:** Resources are provisioned in a planned manner. This can be demandbased, 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.

Demand &amp; Supply

## Cost 9: How do you manage demand and supply resources ?

For a workload that has balanced spend and performance, ensure that everything you pay for is used and avoid significantly underutilizing instances.

- A skewed utilization metric in either direction has an adverse impact on your organization, in either operational costs (degraded performance due to over-utilization), or wasted AWS expenditures (due to over-provisioning).
- Perform an analysis on the workload demand
- Implement a buffer or throttle to manage demand
- Supply resources dynamically
- **References:**
  - [Manage Demand and Supply Resources](#)

# AWS Well-Architected Cost Optimization

# Optimize Over Time

In Collaboration with:



## Best Practices:

- **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.
- **Review and analyze this workload regularly:** Existing workloads are regularly reviewed as per defined processes.

Evaluate New Services

## Cost 10: How do you evaluate new services ?

As AWS releases new services and features, it's a best practice to review your existing architectural decisions to ensure they continue to be the most cost effective.

- A skewed utilization metric in either direction has an adverse impact on your organization, in either operational costs (degraded performance due to over-utilization), or wasted AWS expenditures (due to over-provisioning).
- Develop a workload review process
- Review and analyze this workload regularly
- References:
  - [Optimize Over Time](#)

Organizing partner:

BLOCK71  
COMMUNITY  
By Mili Enterprise  
in partnership with Salim Group

INNOVATION  
FACTORY



## Contents

Introduction AWS Well-Architected Framework (WAF)

AWS Well-Architected Cost Optimization

Monitoring & Alerting Cost

Strategy of Cost Optimization

Q & A

Organizing partner:

By Mili Enterprise  
in partnership with Salto Group



INNOVATION  
FACTORY



## Contents

Introduction AWS Well-Architected Framework (WAF)

AWS Well-Architected Cost Optimization

Monitoring & Alerting Cost

Strategy of Cost Optimization

Q & A

Strategy of Cost Optimization

# AWS Resources for Each Use Cases

## Comparing Use Cases with AWS Resources

Use Cases	Capabilities	AWS Resources
Organize	Construct your cost allocation and governance foundation with your own tagging strategy	<a href="#">AWS Cost Allocation Tags</a>   <a href="#">AWS Cost Categories</a>
Report	Raise awareness and accountability of your cloud spend with the detailed, allocable cost data	<a href="#">AWS Cost Explorer</a>   <a href="#">AWS Cost and Usage Report</a>   <a href="#">AWS Application Cost Profiler</a>
Access	Track billing information across the organization in a consolidated view	<a href="#">AWS Consolidated Billing</a>   <a href="#">AWS Purchase Order Management</a>   <a href="#">AWS Credits</a>
Control	Establish effective governance mechanisms with the right guardrails in place	<a href="#">AWS Cost Anomaly Detection</a>   <a href="#">AWS Identity and Access Management</a>   <a href="#">AWS Organizations</a>   <a href="#">AWS Control Tower</a>   <a href="#">AWS Service Catalog</a>
Forecast	Estimate your resource utilization and spend with forecast dashboards that you create	<a href="#">AWS Cost Explorer (Self-Service)</a>   <a href="#">AWS Budgets (Event-Driven)</a>
Budget	Keep your spend in check with custom budget threshold and auto alert notification	<a href="#">AWS Budgets</a>   <a href="#">AWS Budget Actions</a>   <a href="#">AWS Service Catalog</a>
Purchase	Leverage free trials and programmatic discounts based on your workload pattern and needs	<a href="#">AWS Free Tier</a>   <a href="#">AWS Reserved Instances</a>   <a href="#">AWS Savings Plans</a>   <a href="#">AWS Spot Instances</a>   <a href="#">Amazon DynamoDB On-demand</a>
Elasticity	Scale and schedule your services based on your expected utilization pattern and needs	<a href="#">AWS Instance Scheduler</a>   <a href="#">Amazon Redshift pause and resume</a>   <a href="#">EC2 Auto Scaling</a>   <a href="#">AWS Trusted Advisor</a>
Rightsize	Align your service allocation size to your actual workload demand	<a href="#">AWS Cost Explorer Right Sizing Recommendations</a>   <a href="#">AWS Compute Optimizer</a>   <a href="#">Amazon Redshift resize</a>   <a href="#">Amazon S3 Intelligent Tiering</a>
Inspect	Stay up-to-date with your resource deployment and cost optimization opportunities	<a href="#">AWS Cost Explorer</a>

AWS Resources

## Specific AWS Resources for each use cases

Different use case have different capabilities. Each AWS Resources can be use for multiple use cases.

- Whether you want to organize and track your cost and usage, enhance control through consolidated billing and access permission, enable better planning through budgeting and forecasting, or further lower cost with resources and pricing optimizations, you can leverage our services, tools, and resources to help reduce your AWS bill.

# Tracking and Allocate IT Cost

Use Cases	AWS Resources	Functions
<b>Cost Organization</b>	<ul style="list-style-type: none"> <li>• AWS Cost Allocation Tags</li> <li>• AWS Marketplace Vendor Metered Tagging</li> </ul>	Apply meaningful metadata to your AWS usage and software purchase through AWS Marketplace
	AWS Cost Categories	Allows you to create rules to locally group cost and usage information by account, tags, service, charge type, or other cost categories
<b>Cost Visualization</b>	AWS Cost Explorer	Out-of-the-box reporting solution to visualize your cost by filtering and grouping with few dimensions. Forecasting and planning cost in the future.
	<ul style="list-style-type: none"> <li>• AWS Cost Usage and Report (CUR)</li> <li>• Amazon Athena for querying CUR</li> <li>• Amazon QuickSight for visualize CUR</li> </ul>	Further Intelligence Analysis
<b>Cost Allocation</b>	AWS Application Cost Profiler	Measuring cost for software applications shared by multiple tenants

AWS Resources

## Specific AWS Resources for each use cases

Different use case have different capabilities. Each AWS Resources can be use for multiple use cases.

- Whether you want to organize and track your cost and usage, enhance control through consolidated billing and access permission, enable better planning through budgeting and forecasting, or further lower cost with resources and pricing optimizations, you can leverage our services, tools, and resources to help reduce your AWS bill.

# Manage Billing and Control Cost

Use Cases	AWS Resources	Functions
<b>Cost Control</b>	AWS Billing Console	Present overview of cost incurred by your member accounts from last month, current month actuals, and the projected current month spend
	AWS Cost Anomaly Detection	Investigate historical spend pattern, compare actual spend with expected spend and sending anomalies alert with root cause analysis
	AWS Budget Action	<ul style="list-style-type: none"> <li>Preconfiguration to enforce specific IAM or SCP Policies or stop target EC2 or RDS Instances.</li> <li>This Budget Action can be execute by automatic or manually (by approval workflow)</li> </ul>
<b>Purchase Management</b>	Saving Plan Alerts	Notification to inform prior for any expiration and/or queue purchase events
	AWS Purchase Order Management	Create and manage AWS POs in one place, set invoices matching rules, and track the balance or expiration POs

AWS Resources

## Specific AWS Resources for each use cases

Different use case have different capabilities. Each AWS Resources can be use for multiple use cases.

- Whether you want to organize and track your cost and usage, enhance control through consolidated billing and access permission, enable better planning through budgeting and forecasting, or further lower cost with resources and pricing optimizations, you can leverage our services, tools, and resources to help reduce your AWS bill.

## Strategy of Cost Optimization

# Strategic IT Planning and Evaluation

Use Cases	AWS Resources	Functions
Data Driven Business Cases	AWS Cloud Value Framework	Provide instruments to create business case with economic performance indicators, gauging cost saving, operational efficiency, staff productivity, and business agility
Migration Acceleration	AWS Migration Evaluator	Project future-state AWS configuration and costs by baselining current on-premises infrastructure and resources utilization
Self Service Cost Estimate	AWS Pricing Calculator	Estimate AWS spend by services, service groups as part of your architecture setup, your architecture solution as a whole.
Efficient Organizational Planning	AWS Organizations	Centrally your AWS environments
Pre-approved IT Support	AWS Service Catalog	Allow your organization to create and manage catalogs of pre-approved AWS services and AWS Marketplaces software
Granular Budget Thresholds, Reporting & Alerting	AWS Budget	Configure granular thresholds for costs and usage incurred on AWS and AWS Marketplace

AWS Resources

## Specific AWS Resources for each use cases

Different use case have different capabilities. Each AWS Resources can be use for multiple use cases.

- Whether you want to organize and track your cost and usage, enhance control through consolidated billing and access permission, enable better planning through budgeting and forecasting, or further lower cost with resources and pricing optimizations, you can leverage our services, tools, and resources to help reduce your AWS bill.

# Choosing EC2 Instances -1-

Use Cases	Types	Describe
General Purpose	M and T Families	<ul style="list-style-type: none"> <li>In general, the M5 family is a good fit for web and application servers, back-end servers for enterprise applications, gaming servers, caching fleets, and app development environments. Many companies run their production applications on M5 instances</li> <li>The T3 family is a lower-cost option than the M family. It's also aimed at general purpose workloads, but is meant for applications that are "bursty," meaning it's only CPU-intensive in bursts. It turns out that many applications fit that profile</li> </ul>
Optimization	C5 Optimizing Compute	<p>For workloads that require high amounts of compute, the Amazon EC2 C5 family is the best instance to spin up. The CPUs in this family run at very high clock speeds.</p> <p>C5 family offers a 25% price/performance improvement over the C4 instances, with over 50% for some workloads.</p> <p>The C5 family also has additional memory per vCPU, and twice the performance of the C4 family for vector and floating-point workloads.</p>

EC2 Instances

## Choosing EC2 Instance for Best Fit/Valuable for your Resources

Amazon EC2 is an AWS service that provide compute capacity in the cloud. Like a server, an Amazon EC2 instance has resources like a Central Processing Unit (CPU), an Operating System, local/attached storage, RAM, etc

- Many customers assume that an older generation of an Amazon EC2 family will be cheaper than the newest generation, but usually the opposite is true. Newer generations run on newer processors that usually require less power and cooling. Simply put, the newer generation is cheaper to operate, so AWS charges you less.
- To save money, take a look at the most recent addition to a family. They generally offer a good price-performance ratio. For example, Amazon EC2 M5 instances deliver 14% better price-performance than Amazon EC2 M4 instances on a per-core basis.
- Amazon EC2 prices vary from region to region. If you can be flexible about where your Amazon EC2 instances live, then taking the time to do some price comparisons can really pay off.
- Automate turning on and turning off your instances. You'll save money if you don't rely on doing these actions manually.

# Choosing EC2 Instances -2-

Use Cases	Types	Describe
Optimization	R5, X1, and X1e Optimize for Memory	<ul style="list-style-type: none"> <li>R5 instances for memory-intensive workloads such as data mining, in-memory analytics, caching, high-performance databases, distributed web scale in-memory caches, midsize in-memory databases, and real-time big data analytics</li> <li>The X1 and X1e families deliver very large amounts of storage and are designed for applications with a huge memory footprint that must be on a single instance</li> </ul>
	G4, G3, P3 and F1 Optimize for Accelerated Computing and Graphics	<ul style="list-style-type: none"> <li>For deploying machine learning models in production and graphics- intensive applications, Amazon EC2 G4 instances are cost effective and versatile. With up to 100 Gbps of networking throughput and up to 1.8 TB of local NVMe storage, G4 instances are available in different instance sizes, giving you the flexibility to choose the size that is best suited for your applications</li> <li>The G3 family is a high-performance platform for graphics applications. It's good for 3D visualizations, streaming graphics, server-side graphics workloads and graphics applications based on DirectX and OpenGL</li> </ul>

EC2 Instances

## Choosing EC2 Instance for Best Fit/Valuable for your Resources

Amazon EC2 is an AWS service that provide compute capacity in the cloud. Like a server, an Amazon EC2 instance has resources like a Central Processing Unit (CPU), an Operating System, local/attached storage, RAM, etc

- Many customers assume that an older generation of an Amazon EC2 family will be cheaper than the newest generation, but usually the opposite is true. Newer generations run on newer processors that usually require less power and cooling. Simply put, the newer generation is cheaper to operate, so AWS charges you less.
- To save money, take a look at the most recent addition to a family. They generally offer a good price-performance ratio. For example, Amazon EC2 M5 instances deliver 14% better price-performance than Amazon EC2 M4 instances on a per-core basis.
- Amazon EC2 prices vary from region to region. If you can be flexible about where your Amazon EC2 instances live, then taking the time to do some price comparisons can really pay off.
- Automate turning on and turning off your instances. You'll save money if you don't rely on doing these actions manually.

# Choosing EC2 Instances -3-

Use Cases	Types	Describe
Optimization	G4, G3, P3 and F1 Optimize for Accelerated Computing and Graphics	<ul style="list-style-type: none"> <li>The P3 family is designed for applications such as deep learning, HPC simulations and batch rendering. In fact, any workload that can take advantage of GPUs for computational capabilities is a good fit for P3 instances</li> <li>The F1 family (F stands for FPGA or field programmable gate arrays) is useful for all sorts of workloads that can benefit from specialized hardware designed to meet the demands of the compute pipeline</li> </ul>
Optimize for Storage	H1, D2, I3 (High I/O Optimization)	<ul style="list-style-type: none"> <li>The H1 family was specifically designed for big data and data-intensive workloads, including MapReduce, distributed file systems like Hadoop Distributed File System (HDFS) and MapR-FS, network file systems, log or data processing applications like Apache Kafka, and big data clusters</li> <li>The D2 family uses high-capacity magnetic disks, just as the H1 does, but it has an even higher ratio of disk to CPU and memory, which makes it a good fit for applications such as Massively Parallel Processing (MPP), MapReduce, and Hadoop distributed computing, or a distributed storage system that needs a large amount of local storage or streaming throughput</li> <li>I3 instance used for a high number of IOPS for random reads and writes. They're really great for transactional workloads like NoSQL databases, clustered databases and online transaction processing (OLTP) systems</li> </ul>

EC2 Instances

## Choosing EC2 Instance for Best Fit/Valuable for your Resources

Amazon EC2 is an AWS service that provide compute capacity in the cloud. Like a server, an Amazon EC2 instance has resources like a Central Processing Unit (CPU), an Operating System, local/attached storage, RAM, etc

- Many customers assume that an older generation of an Amazon EC2 family will be cheaper than the newest generation, but usually the opposite is true. Newer generations run on newer processors that usually require less power and cooling. Simply put, the newer generation is cheaper to operate, so AWS charges you less.
- To save money, take a look at the most recent addition to a family. They generally offer a good price-performance ratio. For example, Amazon EC2 M5 instances deliver 14% better price-performance than Amazon EC2 M4 instances on a per-core basis.
- Amazon EC2 prices vary from region to region. If you can be flexible about where your Amazon EC2 instances live, then taking the time to do some price comparisons can really pay off.
- Automate turning on and turning off your instances. You'll save money if you don't rely on doing these actions manually.

# Visibility and Control -1-

Category	View																																		
Tag Mapping	<p><b>Tag Mapping</b></p> <p>Map your tags to Cloudability dimensions for use in our system (you can map multiple tags to a single dimension). Learn more about mapping tags in our <a href="#">Knowledge Base</a>.</p> <table border="1"> <thead> <tr> <th>Cloudability Dimension</th> <th>Tags (Keys)</th> </tr> </thead> <tbody> <tr> <td>DIMENSION 1 Creator</td> <td>creator deployment</td> </tr> <tr> <td>DIMENSION 2 Name</td> <td>Name nuk_id DnsCode</td> </tr> <tr> <td>DIMENSION 3 Role</td> <td>Role</td> </tr> <tr> <td>DIMENSION 4 Class</td> <td>Class</td> </tr> <tr> <td>DIMENSION 5 Team</td> <td>Team</td> </tr> <tr> <td>DIMENSION 6 Batchly</td> <td>Alarms</td> </tr> <tr> <td>DIMENSION 7 Environment</td> <td>Environment environment</td> </tr> <tr> <td>DIMENSION 8 Service</td> <td>Service service</td> </tr> <tr> <td>DIMENSION 9 Application</td> <td>Application</td> </tr> <tr> <td>DIMENSION 10 Cluster</td> <td>Cluster</td> </tr> <tr> <td>DIMENSION 11 Node</td> <td>Node</td> </tr> <tr> <td>DIMENSION 12 Owner</td> <td>Owner</td> </tr> <tr> <td>DIMENSION 13 Deployment</td> <td>deployment</td> </tr> <tr> <td>DIMENSION 14 EMR Job Flow ID</td> <td>awselasticmapreduce:job-flow-id</td> </tr> <tr> <td>DIMENSION 15 StackName</td> <td>awscloudformation:stack-name</td> </tr> <tr> <td>DIMENSION 16 ResourceGroup</td> <td>clby_azure:resourcegroup</td> </tr> </tbody> </table> <p><b>Mapping Options</b></p> <p>Automatically map newly discovered tags <input checked="" type="checkbox"/> ON</p>	Cloudability Dimension	Tags (Keys)	DIMENSION 1 Creator	creator deployment	DIMENSION 2 Name	Name nuk_id DnsCode	DIMENSION 3 Role	Role	DIMENSION 4 Class	Class	DIMENSION 5 Team	Team	DIMENSION 6 Batchly	Alarms	DIMENSION 7 Environment	Environment environment	DIMENSION 8 Service	Service service	DIMENSION 9 Application	Application	DIMENSION 10 Cluster	Cluster	DIMENSION 11 Node	Node	DIMENSION 12 Owner	Owner	DIMENSION 13 Deployment	deployment	DIMENSION 14 EMR Job Flow ID	awselasticmapreduce:job-flow-id	DIMENSION 15 StackName	awscloudformation:stack-name	DIMENSION 16 ResourceGroup	clby_azure:resourcegroup
Cloudability Dimension	Tags (Keys)																																		
DIMENSION 1 Creator	creator deployment																																		
DIMENSION 2 Name	Name nuk_id DnsCode																																		
DIMENSION 3 Role	Role																																		
DIMENSION 4 Class	Class																																		
DIMENSION 5 Team	Team																																		
DIMENSION 6 Batchly	Alarms																																		
DIMENSION 7 Environment	Environment environment																																		
DIMENSION 8 Service	Service service																																		
DIMENSION 9 Application	Application																																		
DIMENSION 10 Cluster	Cluster																																		
DIMENSION 11 Node	Node																																		
DIMENSION 12 Owner	Owner																																		
DIMENSION 13 Deployment	deployment																																		
DIMENSION 14 EMR Job Flow ID	awselasticmapreduce:job-flow-id																																		
DIMENSION 15 StackName	awscloudformation:stack-name																																		
DIMENSION 16 ResourceGroup	clby_azure:resourcegroup																																		

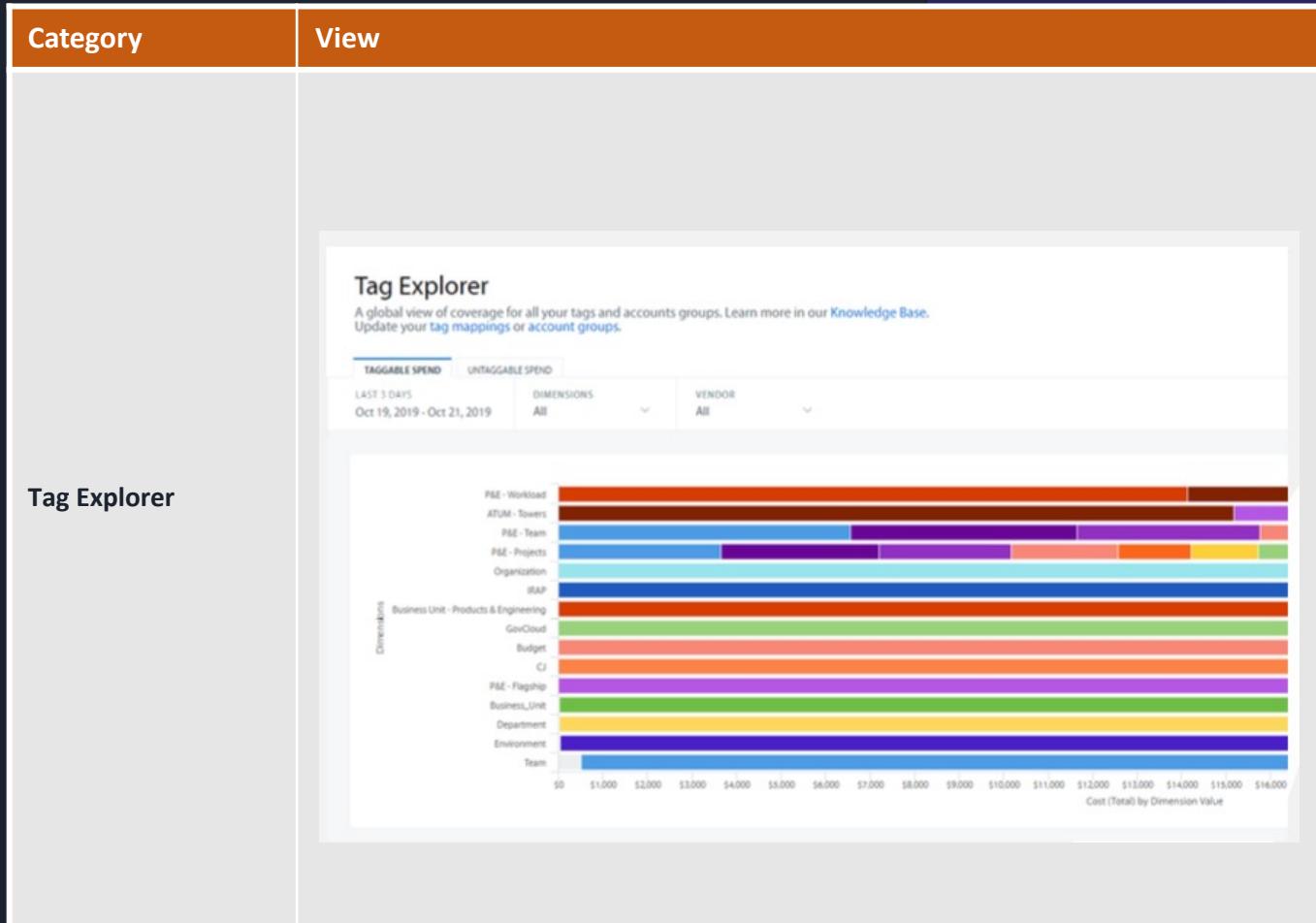
Tag Mapping

## Visibility and Control Resources with Tags

Tags are metadata labels (each with a customer-defined key and value) that you assign to resources so you can keep track of them. Tagging, which most AWS resources support, is essential to making sense of the enormous amount of Amazon EC2 cost and usage data AWS produces

- AWS views tags as a string of characters, and tags need to match exactly if they're going to be lumped together. That means "Environment," "environment," and typos like "environment" will all be sorted into different tags. Tag Mapping lets you take multiple versions of what should be a single tag and map them to one dimension so you can make sure your tags are grouped accurately.
- Tag Explorer gives you a global view by sorting all of your resources into tag keys and breaking down those resources by tag values. At a glance, you'll be able to see how your spend is distributed—and which spend isn't tagged.

# Visibility and Control -2-



Tag Explorer

## Visibility and Control Resources with Tags

Tags are metadata labels (each with a customer-defined key and value) that you assign to resources so you can keep track of them. Tagging, which most AWS resources support, is essential to making sense of the enormous amount of Amazon EC2 cost and usage data AWS produces

- AWS views tags as a string of characters, and tags need to match exactly if they're going to be lumped together. That means "Environment," "environment," and typos like "environment" will all be sorted into different tags. Tag Mapping lets you take multiple versions of what should be a single tag and map them to one dimension so you can make sure your tags are grouped accurately.
- Tag Explorer gives you a global view by sorting all of your resources into tag keys and breaking down those resources by tag values. At a glance, you'll be able to see how your spend is distributed—and which spend isn't tagged.

# Tag Strategy -1-

Category	View
AWS-Generated Cost Allocation Tags	<ul style="list-style-type: none"> <li>They are available only through Billing and Cost Management consoles and reports. You won't see them via AWS Tag Editor</li> <li>Activating them requires a management account</li> <li>You can't use them to collect insights from before you tagged the resource (backdating) and turned it on as cost allocation tags</li> <li>You cannot edit or delete them</li> <li>The number of active tag keys for Billing and Cost Management reports is limited to 500</li> <li>You cannot control the names and values of these variables because AWS auto-generates them</li> <li>Because the system uses CloudTrail logs to generate them, creating the tags can fail if the tag is too large to accept new entries</li> <li>AWS Budgets and Cost Explorer do not display null tag values</li> </ul>
User-Defined Cost Allocation Tags	<ul style="list-style-type: none"> <li>You can only use a key once for each resource.</li> <li>You can't backdate them</li> <li>Like AWS-generated tags, you need to manually activate the tags before they can appear in the Cost Allocation Report</li> <li>Billing and Cost Management does not decode or encode tags for you, so you have to do it yourself</li> <li>Tags on non-metered services do not appear in Cost Management, even after you activate them</li> </ul>

Tag Resources

## AWS Cost Allocation Limitations with Tags

There are two types of tags for AWS cost allocation: AWS-generated tags and User-defined tags.

- AWS tags work by enabling users to add descriptive metadata ("tags") to cloud resources, such as EC2 instances, S3 buckets, databases, and Lambda functions.
- Each tag comprises a value and a key. For each resource, you can create a unique key with only one value. The key can be a business unit such as **Team**. You can then assign the values of **DevTeam1**, **DevTeam2**, and **DevTeam3** to that key.
- Adding tags to a resource adds context to it by describing its use. This approach simplifies categorizing resource utilization insights for the organization. When organizing large quantities of usage and cost data, this can be extremely useful.

# Strategy of Cost Optimization

## Tag Strategy -2-

Issue	Solutions
Coming up with a solid AWS tagging strategy is tough	<ul style="list-style-type: none"> <li>Develop an organizational policy on AWS tagging. It's a good idea to share this policy with all branches in different regions or departments</li> <li>To achieve this, you'll need to get buy-in from your various teams. The best way to do this is to get engineers or engineering and finance teams to share ideas about maintaining consistency in tagging, such as using the same keys, values, and formats</li> <li>Use API AWS Cost to monitoring untag resources by schedule</li> </ul>
Some things you just can't tag	<p>AWS services that support tagging:</p> <ul style="list-style-type: none"> <li>AWS Elastic Compute Cloud (EC2) Service</li> <li>AWS Data Pipeline</li> <li>AWS ElasticCache</li> <li>AWS CloudFormation</li> <li>Amazon Elastic Beanstalk</li> <li>Amazon S3 Glacier</li> <li>Amazon Elastic Load Balancing (ELB)</li> <li>Amazon Kinesis</li> <li>Amazon Relational Database Service (RDS)</li> </ul> <p>You could do some rough math in spreadsheets. But that peanut butter approach could lead to inexact cost visibility. Inexact results may cause you to over-report your cost of goods sold (COGS). Overstating COGS can lead to other problems, such as reporting weaker margins and shooting down your stock price, which can affect your valuation.</p>

Tag Resources

### AWS Cost Allocation Limitations with Tags

There are two types of tags for AWS cost allocation: AWS-generated tags and User-defined tags.

- AWS tags work by enabling users to add descriptive metadata ("tags") to cloud resources, such as EC2 instances, S3 buckets, databases, and Lambda functions.
- Each tag comprises a value and a key. For each resource, you can create a unique key with only one value. The key can be a business unit such as **Team**. You can then assign the values of **DevTeam1**, **DevTeam2**, and **DevTeam3** to that key.
- Adding tags to a resource adds context to it by describing its use. This approach simplifies categorizing resource utilization insights for the organization. When organizing large quantities of usage and cost data, this can be extremely useful.

# Strategy of Cost Optimization

## Tag Strategy -3-

Issue	Solutions
<b>Can't see cost allocation tags in AWS Billing and Cost Management</b>	<ul style="list-style-type: none"> <li>You can only see your cost allocation tags in Billing and Cost Management once you've enabled AWS Cost and Usage Reports, AWS Cost Explorer, legacy reports, or AWS Budgets</li> <li>Because Billing and Cost Management does not decode or encode tags for you, you need to manually tag your resources to begin collecting and understanding your AWS expenses</li> <li>The sooner you do this, the more accurate the cost allocation data you will be able to compile and use to guide your subsequent cost-related decisions</li> </ul>
<b>Have to move an account to another organization</b>	<ul style="list-style-type: none"> <li>If an account is moved to another organization as a member, you must reactivate the cost allocation tags. This can be a time-consuming process in AWS</li> <li>But if you can find a platform that doesn't rely too heavily on tags or native AWS cost management tools such as UCR and AWS Budgets, you can still correctly allocate costs after a few tweaks</li> </ul>
<b>Have dozen or hundreds of tags you need to edit quickly and efficiently</b>	<ul style="list-style-type: none"> <li>It is exhausting and time-consuming to search for and edit individual tags one by one. Instead, use the Tag Editor to find existing resources by type of resource, region, and existing tags. After locating the resources you want to change, Tag Editor lets you add new Values and Keys in bulk</li> <li>Now, keep in mind that new stack deployments might remove these tags if they also you did not add them to the original infrastructure-as-code that deployed the resource</li> <li>Make sure you inform anyone even remotely involved in tagging your AWS resources about this change in advance</li> <li>Avoid confusing different teams with a situation that will lead to more mixups. For example, not letting everyone know you will use the value name Production moving forward might cause four separate dev teams to refer to the same resource as Production, Prod, prod, and production. Chaos would ensue</li> </ul>

Tag Resources

### AWS Cost Allocation Limitations with Tags

There are two types of tags for AWS cost allocation: AWS-generated tags and User-defined tags.

- AWS tags work by enabling users to add descriptive metadata ("tags") to cloud resources, such as EC2 instances, S3 buckets, databases, and Lambda functions.
- Each tag comprises a value and a key. For each resource, you can create a unique key with only one value. The key can be a business unit such as **Team**. You can then assign the values of **DevTeam1**, **DevTeam2**, and **DevTeam3** to that key.
- Adding tags to a resource adds context to it by describing its use. This approach simplifies categorizing resource utilization insights for the organization. When organizing large quantities of usage and cost data, this can be extremely useful.

# Tag Strategy -4-

Issue	Solutions
<b>Have to chase engineers down to get them to tag resources properly</b>	<ul style="list-style-type: none"> <li>Finance is painfully aware when cloud costs go over budget. But engineering might be more interested in innovating and testing in live productions. So you'll want to align engineering with finance to prevent this disconnect from ruining your efforts to nurture a cost-aware culture in your company.</li> <li>In this case, the goal would be: <ul style="list-style-type: none"> <li>Ensure engineering understands the cost impact of its activities.</li> <li>Enable finance to see how, why, and where cloud spend is going to ensure cost optimization measures do not hinder innovation.</li> </ul> </li> <li>You can empower engineers to become more cost-aware by giving them the means to increase their cost visibility. You'll want to choose a solution that can clearly articulate engineering costs in a language they understand, such as cost per deployment, development team, and product feature.</li> <li>Only then can your engineers create cost-optimized software and react quickly whenever a cost spike occurs that would significantly increase your cloud costs.</li> </ul>
<b>Having hard time predicting cloud spend</b>	<ul style="list-style-type: none"> <li>Creating a thorough AWS cost allocation strategy may require you to look beyond AWS native tools such as Cloudtrail, CloudWatch, Budgets, Usage, and Cost Reports. This is true for everyone, from startups to scaleups to enterprises, because you do not want to wait until your resources are so numerous that they become difficult to manage.</li> <li>Instead, you'll want to designate a single source of truth to ingest, enrich, and analyze your entire system's metrics, events, and logs. You can then correlate the metrics of your platform with the business units/outcomes you're interested in.</li> <li>The tool or method should enable you to collect more than just data on tagged resources. It should also be able to capture the key performance indicators (KPIs) generated by your infrastructure, apps, containers, Kubernetes, and many other components.</li> </ul>

Tag Resources

## AWS Cost Allocation Limitations with Tags

There are two types of tags for AWS cost allocation: AWS-generated tags and User-defined tags.

- AWS tags work by enabling users to add descriptive metadata ("tags") to cloud resources, such as EC2 instances, S3 buckets, databases, and Lambda functions.
- Each tag comprises a value and a key. For each resource, you can create a unique key with only one value. The key can be a business unit such as **Team**. You can then assign the values of **DevTeam1**, **DevTeam2**, and **DevTeam3** to that key.
- Adding tags to a resource adds context to it by describing its use. This approach simplifies categorizing resource utilization insights for the organization. When organizing large quantities of usage and cost data, this can be extremely useful.

Strategy of Cost Optimization

# Cost Planning and Controlling -1-

## Cost Budgeting (Fixed Plan / Monthly Plan)

Set your budget [Info](#)

▼ How to set up your budget



**Step 1: Set budget amount**

Select the period and whether you would like to have a fixed budget or to specify a budget plan, then enter your budget amount.



**Step 2: Scope your budget - optional**

Add dimensions of data to narrow on a set of cost information. For example, you could select a number of AWS services to track as part of this budget.



**Step 3: Enter in remaining budget details**

Define the budget name.

**Set budget amount**

**Period**

Daily budgets do not support enabling forecasted alerts, daily budget planning, or attaching actions.

Monthly

Daily

Monthly

Quarterly

Annually

**Start month**

Aug ▾ 2021 ▾

**Choose how to budget**

Fixed

Create a budget that tracks against a single monthly budgeted amount.

Monthly budget planning

Specify your budgeted amount for each budget period.

**Enter your budgeted amount (\$)**

Last month's cost: \$4,631.58

28.JUL.2021: 00:00:00 UTC  
(\$100000 budgeted over 1 year)

Organizing partner:



**INNOVATION**  
FACTORY



Strategy of Cost Optimization

# Cost Planning and Controlling -2-

## Usage Budget

**Set your budget** info

▼ How to set up your budget



**Step 1: Set budget amount**  
Select the period and whether you would like to have a fixed budget or to specify a budget plan, then enter your budget amount.



**Step 2: Scope your budget - optional**  
Add dimensions of data to narrow on a set of cost information. For example, you could select a number of AWS services to track as part of this budget.



**Step 3: Enter in remaining budget details**  
Define the budget name.

**Choose what you're budgeting against**

Budget against  
Select whether you want to measure your budget by usage type groups or usage types.

Usage type groups  
Usage type groups are filters that collect a specific category of usage type filters into one filter.

Usage types  
Usage types are the units that each service uses to measure the usage of a specific type of resource.

Usage type groups  
Select which usage type groups you would like to budget against

Cancel

Cancel

zoeva molla. Dabo Bumta  
zoeva molla molla zoeva Bumta koma molla jemaa profiles ad molla  
zoeva zoeva Bumta

Strategy of Cost Optimization

# Cost Planning and Controlling -3-

## Usage Budget

**Set your budget** [Info](#)

▼ How to set up your budget



**Step 1: Set budget amount**  
Select the period and whether you would like to have a fixed budget or to specify a budget plan, then enter your budget amount.



**Step 2: Scope your budget - optional**  
Add dimensions of data to narrow on a set of cost information. For example, you could select a number of AWS services to track as part of this budget.



**Step 3: Enter in remaining budget details**  
Define the budget name.

**Choose what you're budgeting against**

Budget against  
Select whether you want to measure your budget by usage type groups or usage types.

**Usage type groups**  
Usage type groups are filters that collect a specific category of usage type filters into one filter.

**Usage types**  
Usage types are the units that each service uses to measure the usage of a specific type of resource.

Usage type groups  
Select which usage type groups you would like to budget against

Select usage type groups

Cancel Previous Next

Strategy of Cost Optimization

# Cost Planning and Controlling -3-

## Saving Plan Budget

**Set your budget** [Info](#)

▼ How to set up your budget



**Step 1: Set threshold**  
Select the period and whether you would like to monitor your spend against utilization or coverage, then enter the preferred threshold.



**Step 2: Scope your budget - optional**  
Add dimensions of data to narrow on a set of cost information. For example, you could select a number of AWS services to track as part of this budget.



**Step 3: Enter in remaining budget details**  
Define the budget name.

---

**Utilization threshold**

**Period**  
Daily budgets do not support enabling forecasted alerts, or daily budget planning.

Monthly

**Monitor my spend against** [Info](#)  
Whether you want to budget against your Savings Plans by utilization (%) or coverage (%).

Utilization of Savings Plans  
The utilization will measure if there are unused or underutilized Savings Plans.

Coverage of Savings Plans  
The coverage will measure how much of your instance usage is covered by Savings Plans.

**Utilization threshold (%)**  
Enter the utilization percentage of your Savings Plans that you would like to stay above.

0.00% (Suggested budget: 0.00% based on last month.)

Strategy of Cost Optimization

# Cost Planning and Controlling -3-

## Reservation Budget

**Set your budget** Info

▼ How to set up your budget



**Step 1: Set threshold**  
Select the period and whether you would like to monitor your spend against utilization or coverage, then enter the preferred threshold.



**Step 2: Scope your budget - optional**  
Add dimensions of data to narrow on a set of cost information. For example, you could select a number of AWS services to track as part of this budget.



**Step 3: Enter in remaining budget details**  
Define the budget name.

---

**Coverage threshold**

**Period**  
Daily budgets do not support enabling forecasted alerts, or daily budget planning.

**Monitor my spend against** Info  
Whether you want to budget against your reservations by utilization (%) or coverage (%).  
 Utilization of reservations  
 Coverage of reservations  
The utilization will measure if your reservations are unused or underutilized.

**Service** Info  
Which service would you like to budget against?

---

**Scope your budget** Info  
ECS-Jurangkuasa (Elastic Container Service - Container)  
  
Amazon Elastic Container Service (ECS) runs your Docker containers on Amazon's infrastructure.

Strategy of Cost Optimization  
**Organization Flow**

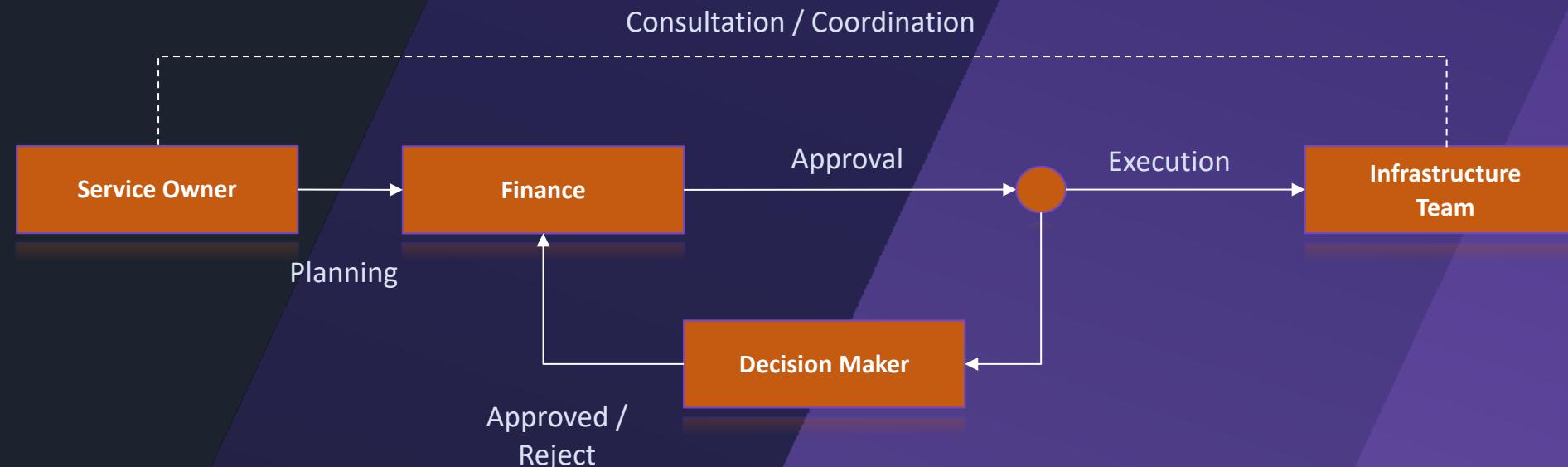
In Collaboration with:



www.zebrax.id

### General Workflow for Approval Process

- Service Owner (Engineer, including Product & Business Team) → Planner
- Infrastructure Team (DevOps / SysEng / SRE) → Consultant & Executor
- Finance → Supervisor & Acknowledge (ACK)
- Decision Maker (VP / CTO / COO / CEO) → Approval



# Summary and Highlights

- **Define Group of Tags**  
<https://ap-southeast-1.console.aws.amazon.com/resource-groups/home?region=ap-southeast-1>
- **Custom Tags**
  - **ResourceGroup (EC2, EBS, ELB, ALB, S3, etc)**
  - **ProductGroup (Products & Projects)**
- **Auto Sleep & Wakeup Instances (Schedule Stop & Run Instances)**
- **Choosing of Instance Type → AWS Calculator (<https://calculator.aws/>)**
- **Choosing the right types of Manage Services (RDS / Database, EKS, EMR, Caching / Elasticache)**
- **Cost Budgeting (Planning) and Evaluation**

Summary

## Summary for Cost Optimization

There are plenty of ways to create optimization of your infrastructure, this section we cover up for highlights

- Defining Group & Tags.
- Customizing Tags by each different resources.
- Plan to scheduling Stop and Run EC2 Instances.
- Choose the best fit for instance type.
- Choose the best fit for manage services AWS.
- Do the plan for budgeting, reserved instance will much reducing cost.
- Evaluate budget planning at the end of periods (Monthly, Quarterly, Half Year, Yearly).

Organizing partner:

BLOCK71 COMMUNITY  
By Mili Enterprise  
in partnership with Salim Group

INNOVATION  
FACTORY



## Contents

Introduction AWS Well-Architected Framework (WAF)

AWS Well-Architected Cost Optimization

Monitoring & Alerting Cost

Strategy of Cost Optimization

Q & A

## References

# Official AWS Documentation

Sources	URL Link
<b>Monitoring your usages and costs</b>	<a href="https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/monitoring-costs.html">https://docs.aws.amazon.com/awsaccountbilling/latest/aboutv2/monitoring-costs.html</a>
<b>Monitor, track and analyze</b>	<a href="https://aws.amazon.com/aws-cost-management/aws-cost-optimization/monitor-track-and-analyze/">https://aws.amazon.com/aws-cost-management/aws-cost-optimization/monitor-track-and-analyze/</a>
<b>AWS Well-Architected Framework - Cost Optimization</b> <b>How do you monitor usage and cost ?</b>	<a href="https://wa.aws.amazon.com/wat.question.COST_3.en.html">https://wa.aws.amazon.com/wat.question.COST_3.en.html</a>
<b>Managing your costs with AWS Budgets</b>	<a href="https://docs.aws.amazon.com/cost-management/latest/userguide/budgets-managing-costs.html?ref=wellarchitected">https://docs.aws.amazon.com/cost-management/latest/userguide/budgets-managing-costs.html?ref=wellarchitected</a>
<b>Creating a billing alarm to monitor your estimated AWS charges</b>	<a href="https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/monitor_estimated_charges_with_cloudwatch.html">https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/monitor_estimated_charges_with_cloudwatch.html</a>
<b>AWS Billing API</b>	<a href="https://docs.aws.amazon.com/aws-cost-management/latest/APIReference/awsbilling-api.pdf">https://docs.aws.amazon.com/aws-cost-management/latest/APIReference/awsbilling-api.pdf</a>



# Thank You

**PT ZEBRA CROSS TEKNOLOGI**

INDY Bintaro Office Park, Building A, 5<sup>th</sup> floor  
Tangerang Selatan 15224

Contact us: [inquiry@zebrax.id](mailto:inquiry@zebrax.id)

[www.zebrax.id](http://www.zebrax.id)



Organizing partner:  
**BLOCK71** **INNOVATION**  
COMMUNITY **FACTORY**  
By M3S Enterprise  
in partnership with Salim Group

**Data are about understanding...**

**... to understand your opportunity is to  
understand your data**

**ZEBRA**   
**Enabling Industry 4.0**