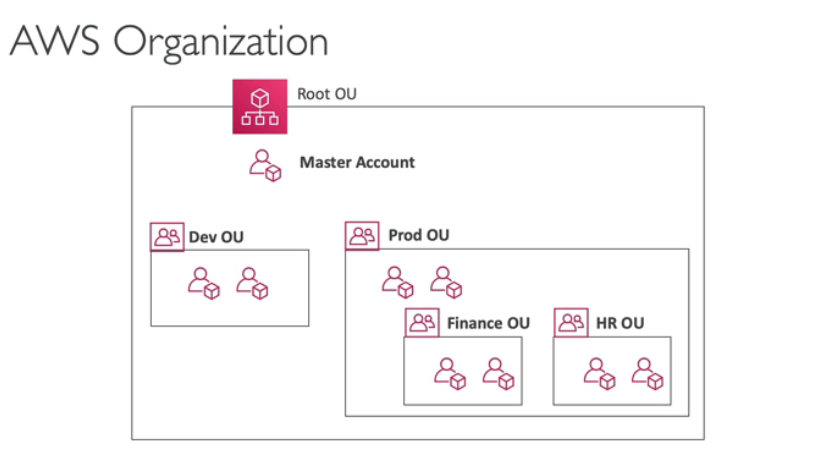
**Account Management, Billing Support**

## AWS Organizations

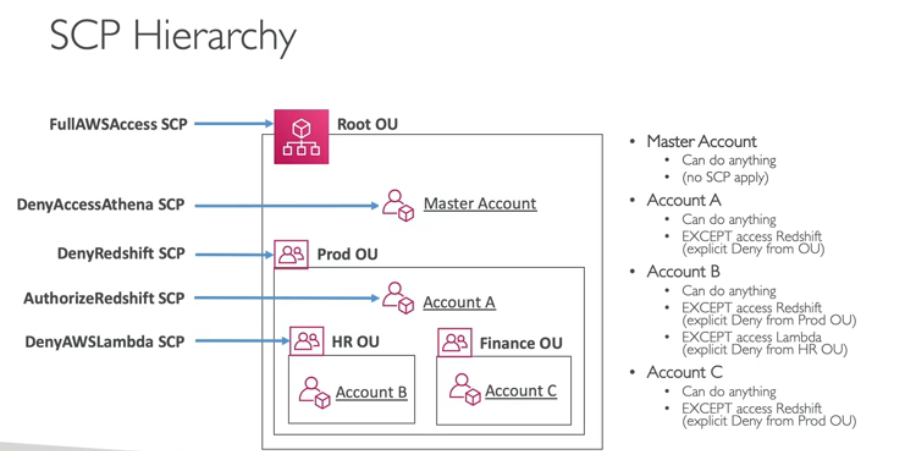
* Global service
* Allows to manage **multiple AWS accounts**
* The main account is the master account
* Cost Benefits:
  + Consolidated Billing across all accounts - single payment method
  + Pricing benefits from aggregated usage (volume discount for EC2, S3…)
  + Pooling of Reserved EC2 instances for optimal savings
* API is available to **automate AWS account creation**
* Restrict account privileges using Service Control Policies (SCP)

## Multi Account Strategies

* Create accounts per **department**, per **cost center**, per **dev / test / prod**, based on regulatory restrictions (using SCP), for better resource isolation (ex: VPC), to have separate per-account service limits, isolated account for logging
* Multi Account vs One Account Multi VPC
* Use standards for billing purposes
* Enable CloudTrail on all accounts, send logs to central S3 account
* Send CloudWatch Logs to central logging account



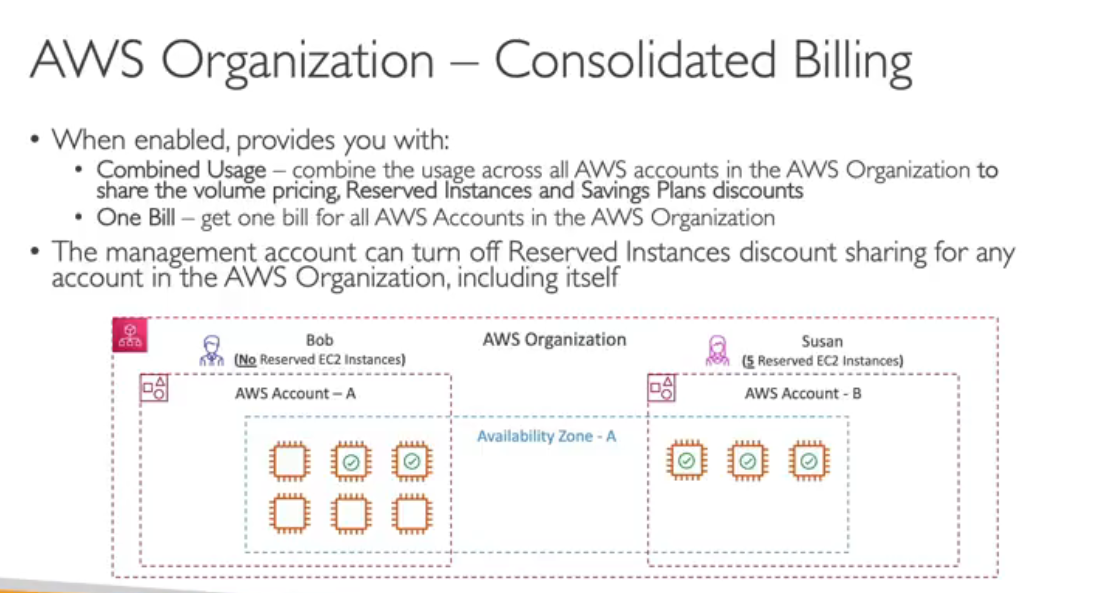
## Service Control Policies (SCP)

* Whitelist or blacklist IAM actions
* Applied at the OU or Account level
* Does not apply to the Master Account
* SCP is applied to all the Users and Roles of the Account, including Root user
* The SCP does not affect service-linked roles
  + Service-linked roles enable other AWS services to integrate with AWS Organizations and can't be restricted by SCPs.
* SCP must have an explicit Allow (does not allow anything by default)
* Use cases:
  + Restrict access to certain services (for example: can’t use EMR)
  + Enforce PCI compliance by explicitly disabling services
* 



## AWS Organization - Consolidated Billing

* When enabled, provides you with:
  + Combined Usage – combine the usage across all AWS accounts in the AWS Organization to share the volume pricing, Reserved Instances and Savings Plans discounts
  + One Bill – get one bill for all AWS Accounts in the AWS Organization
* The management account can turn off Reserved Instances discount sharing for any account in the AWS Organization, including itself



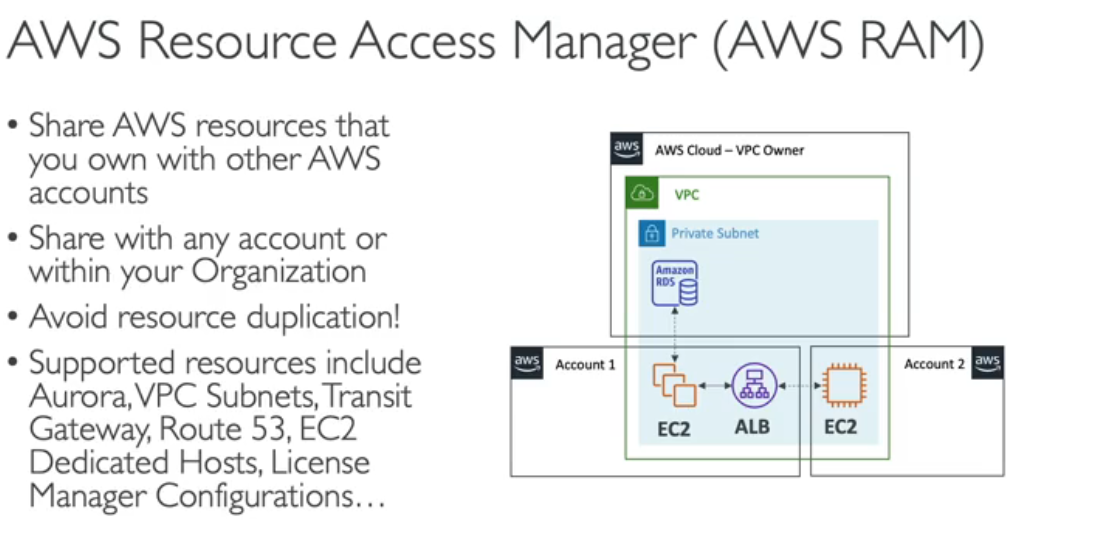


## AWS Control Tower

* Easy way to set up and govern a secure and compliant multi-account AWS environment based on best practices
* Benefits:
  + Automate the set up of your environment in a few clicks
  + Automate ongoing policy management using guardrails
  + Detect policy violations and remediate them
  + Monitor compliance through an interactive dashboard
* AWS Control Tower runs on top of AWS Organizations:
  + It automatically sets up AWS Organizations to organize accounts and implement SCPs (Service Control Policies)
  + Imagine you have a big house with many rooms. A multi-account strategy in AWS is like dividing the house into sections. Each section (account) has a specific purpose, like a kitchen for cooking (development account) or a bedroom for sleeping (production account). This keeps things organized and avoids a mess (security breaches)

## AWS Resource Access Manager (AWS RAM)

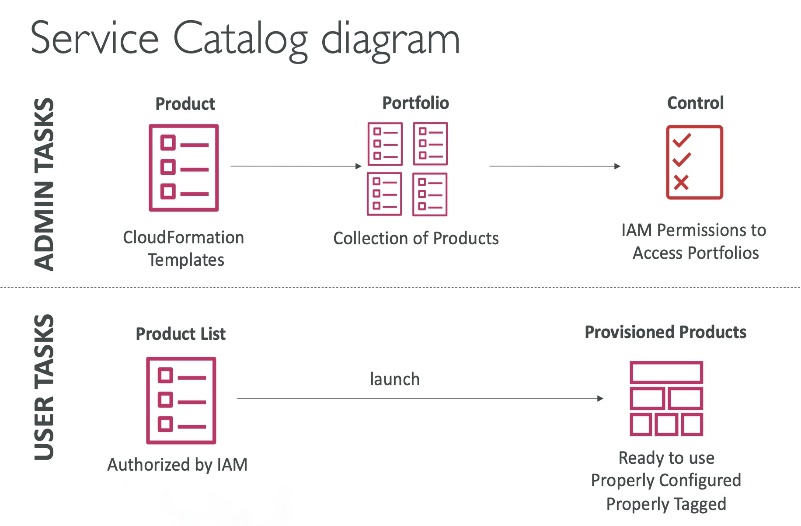
* Share AWS resources that you own with other AWS accounts
* Share with any account or within your Organization
* Avoid resource duplication!
* Supported resources include Aurora, VPC Subnets, Transit Gateway, Route 53, EC2 Dedicated Hosts, License Manager Configurations.





## AWS Service Catalo

* Users that are new to AWS have too many options, and may create stacks that are not compliant or in line with the rest of the organization
* Some users just want a quick self-service portal to launch a set of authorized products pre-defined by admins
* Includes: virtual machines, databases, storage options, etc…
* Enter AWS Service Catalog!
* Imagine you're moving into a new house (AWS) with many rooms (services) and tools (resources). It can be overwhelming for someone new to the house (new AWS user) to know what to use and how to put things together (configure resources).

[](https://github.com/kananinirav/AWS-Certified-Cloud-Practitioner-Notes/blob/master/images/service_catalog.png)

## 

## Pricing Models in AWS

* AWS has 4 pricing models:
* **Pay as you go**: pay for what you use, remain agile, responsive, meet scale demands
* **Save when you reserve**: minimize risks, predictably manage budgets, comply with long-terms requirements
  + Reservations are available for EC2 Reserved Instances, DynamoDB Reserved Capacity, ElastiCache Reserved Nodes, RDS Reserved Instance, Redshift Reserved Nodes
* **Pay less by using more**: volume-based discounts
* **Pay less as AWS grows**

### EC2

* Only charged for what you use
* Number of instances
* Instance configuration:
  + Physical capacity
  + Region
  + OS and software
  + Instance type
  + Instance size
* ELB running time and amount of data processed
* Detailed monitoring
* On-demand instances:
  + Minimum of 60s
  + Pay per second (Linux/Windows) or per hour (other)
* Reserved instances:
  + Up to 75% discount compared to On-demand on hourly rate
  + 1- or 3-years commitment
  + All upfront, partial upfront, no upfront
* Spot instances:
  + Up to 90% discount compared to On-demand on hourly rate
  + Bid for unused capacity
* Dedicated Host:
  + On-demand
  + Reservation for 1 year or 3 years commitment
  + Savings plans as an alternative to save on sustained usage

### Lambda & ECS

* Lambda:
  + Pay per call
  + Pay per duration
* ECS:
  + EC2 Launch Type Model: No additional fees, you pay for AWS resources stored and created in your application
* Fargate:
  + Fargate Launch Type Model: Pay for vCPU and memory resources allocated to your applications in your containers

## Storage Pricing

### S3

* Storage class: S3 Standard, S3 Infrequent Access, S3 One-Zone IA, S3 Intelligent Tiering, S3 Glacier and S3 Glacier Deep Archive
* Number and size of objects: Price can be tiered (based on volume)
* Number and type of requests
* Data transfer OUT of pay the S3 region
* S3 Transfer Acceleration
* Lifecycle transitions
* Similar service: EFS (pay per use, has infrequent access & lifecycle rules)

### EBS

* Volume type (based on performance)
* Storage volume in GB per month provisioned
* IOPS:
  + General Purpose SSD: Included
  + Provisioned IOPS SSD: provisioned amount in IOPS
  + Magnetic: Number of requests
* Snapshots:
  + Added data cost per GB per month
* Data transfer:
  + Outbound data transfer are tiered for volume discounts
  + Inbound is free

## Database Pricing - RDS

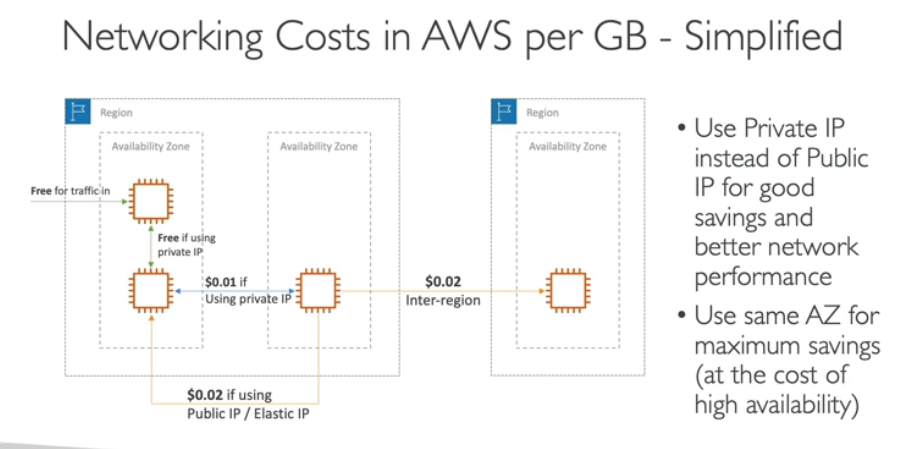
* Per hour billing
* Database characteristics:
  + Engine
  + Size
  + Memory class
* Purchase type:
  + On-demand
  + Reserved instances (1 or 3 years) with required up-front
* Backup Storage: There is no additional charge for backup storage up to 100% of your total database storage for a region.
* Additional storage (per GB per month)
* Number of input and output requests per month
* Deployment type (storage and I/O are variable):
  + Single AZ
  + Multiple AZs
* Data transfer:
  + Outbound data transfer are tiered for volume discounts
  + Inbound is free

## Database Pricing - RDS

* Per hour billing
* Database characteristics:
  + Engine
  + Size
  + Memory class
* Purchase type:
  + On-demand
  + Reserved instances (1 or 3 years) with required up-front
* Backup Storage: There is no additional charge for backup storage up to 100% of your total database storage for a region.
* Additional storage (per GB per month)
* Number of input and output requests per month
* Deployment type (storage and I/O are variable):
  + Single AZ
  + Multiple AZs
* Data transfer:
  + Outbound data transfer are tiered for volume discounts
  + Inbound is free

## Content Delivery - CloudFront

* Pricing is different across different geographic regions
* Aggregated for each edge location, then applied to your bill
* Data Transfer Out (volume discount)
* Number of HTTP/HTTPS requests



## Content Delivery - CloudFront

* Pricing is different across different geographic regions
* Aggregated for each edge location, then applied to your bill
* Data Transfer Out (volume discount)
* Number of HTTP/HTTPS requests

## Savings Plan

* Commit a certain $ amount per hour for 1 or 3 years
* Easiest way to setup long-term commitments on AWS

**EC2 Savings Plans:**

* Offer the deepest discounts (up to 72%).
* Require commitment to a specific **instance family** (e.g., M5 instances) in a chosen **region** (e.g., N. Virginia).
* Provide flexibility **within the family** (switch between instance sizes, operating systems (OSes), and tenancies).
* Payment options include all upfront, partial upfront, or no upfront.

**Compute Savings Plans:**

* Offer slightly lower discounts (up to 66%).
* Provide the most flexibility - apply to any compute usage across **regions, families, sizes, OSes, tenancies, and compute options (EC2 instances, Fargate, and Lambda functions).**
* Currently set up from the AWS Cost Explorer console.
* **Machine Leaming** Savings Plan: SageMaker. ..
* • Setup from the AWS Cost Explorer console

## AWS Compute Optimizer

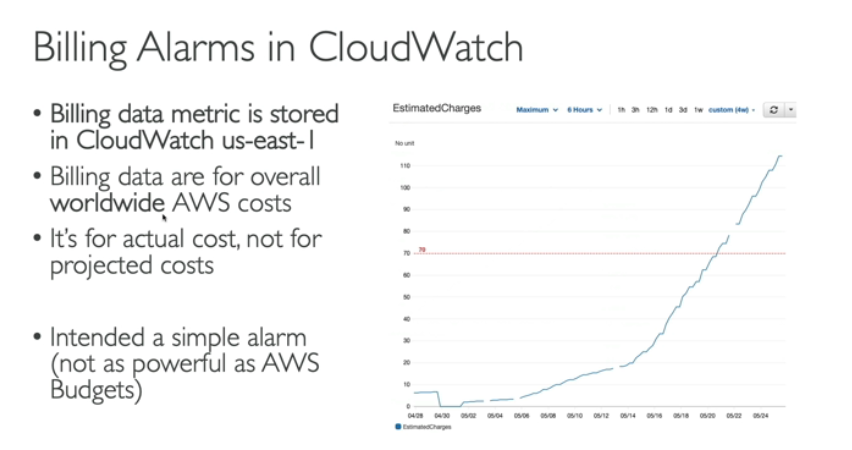
* Reduce costs and improve performance by recommending optimal AWS resources for your workloads
* Helps you choose optimal configurations and right - size your workloads (over/under provisioned)
* Uses Machine Learning to analyze your resources’ configurations and their utilization CloudWatch metrics
* Supported resources
  + EC2 instances
  + EC2 Auto Scaling Groups
  + EBS volumes
  + Lambda functions
* Lower your costs by up to 25%
* Recommendations can be exported to S3

## Billing and Costing Tools

* Estimating costs in the cloud:
  + Pricing Calculator
* Tracking costs in the cloud:
  + Billing Dashboard
  + Cost Allocation Tags
  + Cost and Usage Reports
  + Cost Explorer
* Monitoring against costs plans:
  + Billing Alarms
  + Budgets

## Billing Alarms in CloudWatch

* Billing data metric is stored in CloudWatch us-east1
* Billing data are for overall worldwide AWS costs
* It’s for actual cost, not for projected costs
* Intended a simple alarm (not as powerful as AWS Budgets)

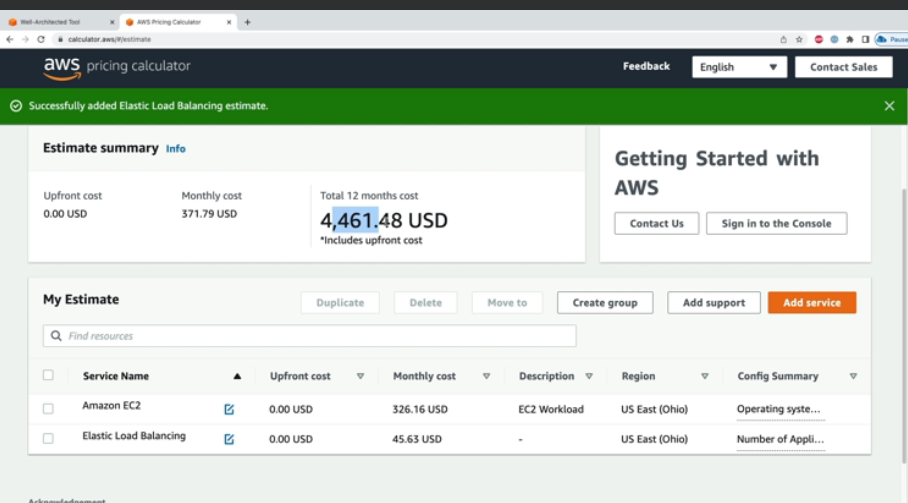


## AWS Budgets

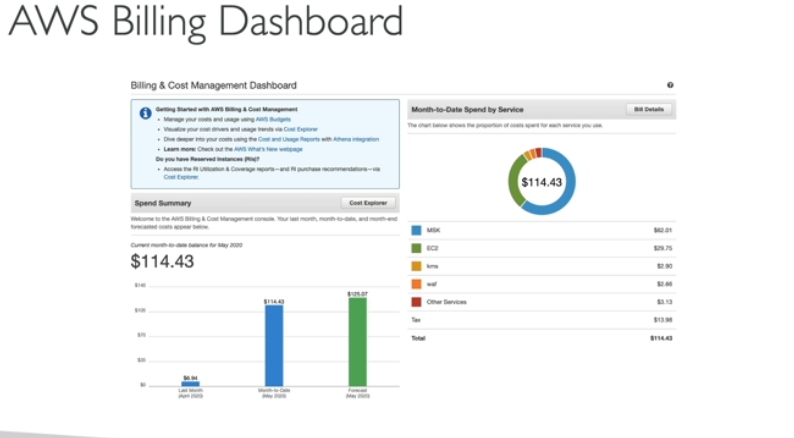
* Create budget and send alarms when costs exceeds the budget
* 3 types of budgets: Usage, Cost, Reservation
* For Reserved Instances (RI)
  + Track utilization
  + Supports EC2, ElastiCache, RDS, Redshift
* Up to 5 SNS notifications per budget
* Can filter by: Service, Linked Account, Tag, Purchase Option, Instance Type, Region, Availability Zone, API Operation, etc…
* Same options as AWS Cost Explorer!
* 2 budgets are free, then $0.02/day/budget

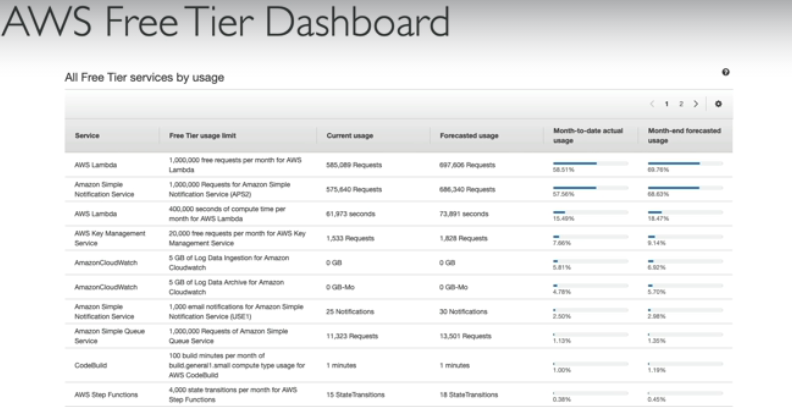
**AWS Pricing Calculator**

• Estimate the cost for your solution architecture

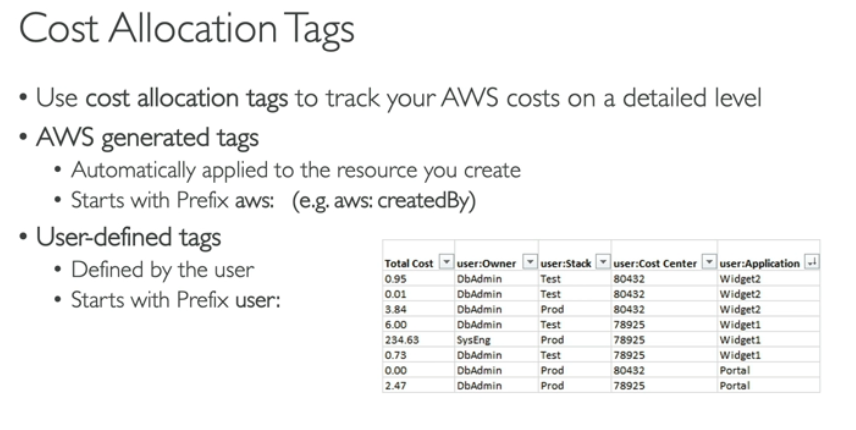


The AWS Billing Dashboard is a tool within the AWS Billing and Cost Management console that allows you to monitor your AWS cloud spending and analyze your costs





track your Free Tier usage





* **Billing Dashboard:** Imagine a monthly bill that summarizes your total electricity usage for your home.
* **Cost Allocation Tags:** Think of breaking down that electricity bill further based on usage in specific areas of your home (kitchen, living room, etc.) using smart plugs that track individual appliance usage.

**Deep Dive into Costs and Usage:**

* CUR goes beyond the basic view of the Billing Dashboard and offers a granular level of detail.

**Most Comprehensive Data Set:**

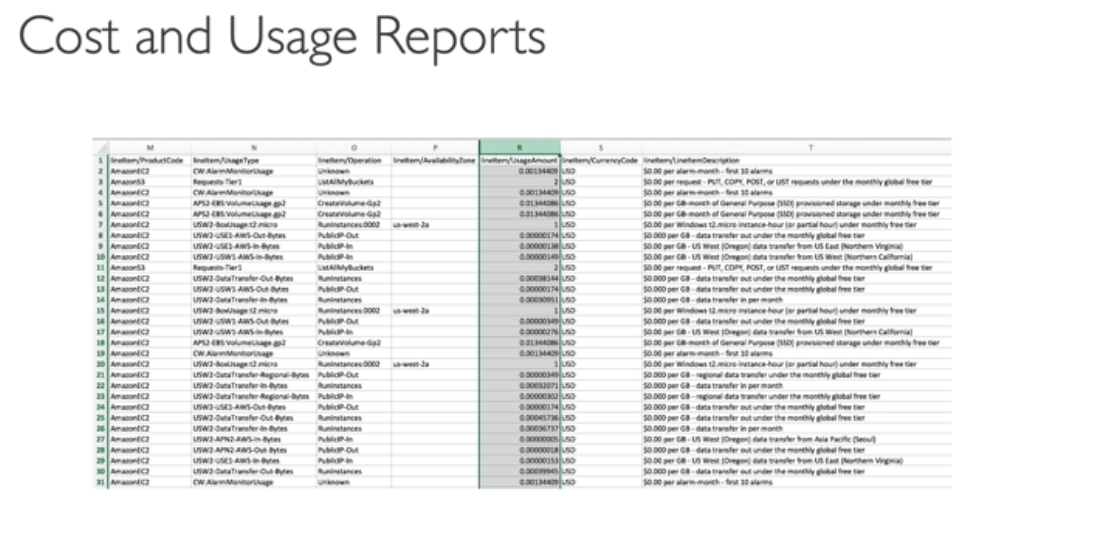
* It provides the most extensive set of cost and usage information for your AWS account.
* This includes additional details like service specifics, reservations (e.g., EC2), and metadata.

**Detailed Line Items:**

* CUR breaks down your AWS usage by service category, user, and time (hourly or daily).
* It also includes any cost allocation tags you've defined.

**Integration with Analytics Tools:**

* The report data can be exported to formats compatible with analytics services like Amazon Athena, Redshift, or QuickSight.



**Cost Explorer**

• Visualize, understand, and manage your AWS costs and usage over time

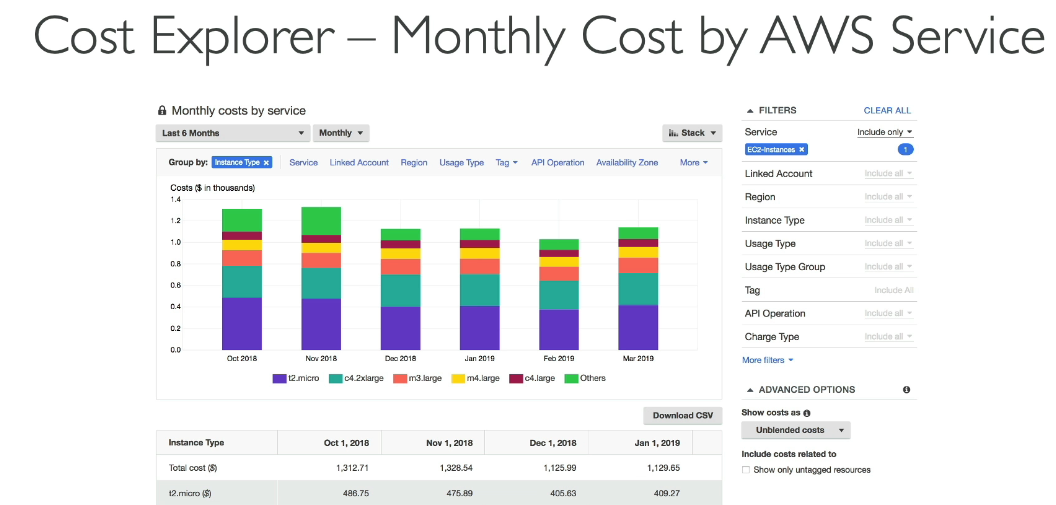
• Create custom reports that analyze cost and usage data.

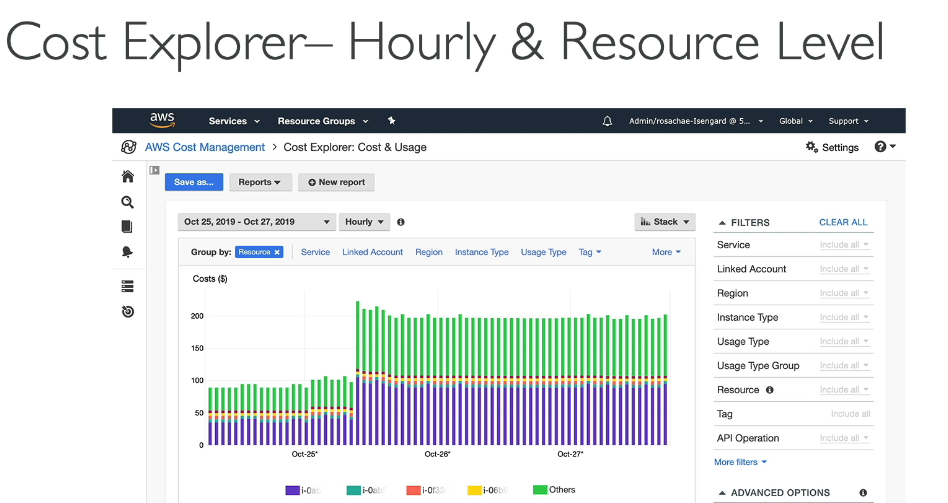
• Analyze your data at a high level: total costs and usage across all accounts

• Or Monthly, hourly, resource level granularity

• Choose an optimal Savings Plan (to lower prices on your bill)

• Forecast usage up to 12 months based on previous usage





## Billing Alarms in CloudWatch

* Billing data metric is stored in CloudWatch us-east1
* Billing data are for overall worldwide AWS costs
* It’s for actual cost, not for projected costs
* Intended a simple alarm (not as powerful as AWS Budgets)
* CloudWatch stores billing data metrics in the us-east-1 region by default. This doesn't affect your ability to monitor costs for all regions

## AWS Budgets

* Create budget and send alarms when costs exceeds the budget
* 3 types of budgets: Usage, Cost, Reservation
* For Reserved Instances (RI)
  + Track utilization
  + Supports EC2, ElastiCache, RDS, Redshift
* Up to 5 SNS notifications per budget
* Can filter by: Service, Linked Account, Tag, Purchase Option, Instance Type, Region, Availability Zone, API Operation, etc…
* Same options as AWS Cost Explorer!
* 2 budgets are free, then $0.02/day/budget

CloudWatch Billing Alarms provide a basic way to monitor your overall AWS spending across regions. It's a good option if you just need to be notified when actual costs reach a certain level. However, if you require more sophisticated cost management features like budgeting, forecasts, and detailed filtering, then AWS Budgets would be a better fit.

## AWS Cost Anomaly Detection

* Continuously monitor your cost and usage using ML to detect unusual spends
* It learns your unique, historic spend patterns to detect one-time cost spike and/or continuous cost increases (you don't need to define thresholds)
* Monitor AWS services, member accounts, cost allocation tags, or cost categories
* Sends you the anomaly detection report with root-cause analysis
* Get notified with individual alerts or daily/weekly summary (using SNS)