SAP-C01 Study Notes: Cost optimization

AWS Certified Solutions Architect– Professional (SAP-C01)

Study notes - Sep'2019

Cost Optimization pillar - Well architected framework

https://d1.awsstatic.com/whitepapers/architecture/AWS-Cost-Optimization-Pillar.pdf

Cost optimization in the cloud is composed of four areas:

1. Cost-effective resources
    a. Appropriate provisioning - Use Cloudwatch
    b. Right sizing - Use Cloudwatch & Trusted Advisor
    c. Purchasing options: On Demand Instances, Spot Instances, and Reserved Instances
    d. Geographic selection
    e. Managed services
    f. Optimize data transfer
2. Matching supply with demand
    a. Demand-based - EC2 Auto scaling with CloudWatch
    b. Buffer-based - buffer work items on SQS or Kinesis, Proces with EC2 Spot instances or lambdas
    c. Time-based - EC2 Auto scaling & CloudFormation (static setup)
3. Expenditure awareness - Cost Explorer, Tagging, CloudWatch alerts
4. Optimizing over time
    a. Establish a cost optimization function - CCoE, AWS Enterprise support
    b. Establish goals and metrics - Establish goals and metrics that your organization can use to measure its progress. Examples:
        i. Reduce the cost per transaction or output of a system by x% every 6 or 12 months.
        ii. % of your On-Demand & RI EC2 instances that are turned on and off every day.
        iii.The number of “always on” instances that are running as reserved capacity.
    c. Gather insight and perform analysis
        i. The Billing and Cost Management Dashboard (including Cost Explorer and Budgets),
        ii. Amazon CloudWatch, and AWS Trusted Advisor.
    d. Report and validate

AWS Tools for Reporting and Cost Optimization

https://docs.aws.amazon.com/whitepapers/latest/cost-optimization-laying-the-foundation/reporting-cost-optimization-tools.html

To help you track, report, and analyze costs over time, AWS provides several reporting and cost-optimization tools:

- Cost Explorer – See patterns in AWS spend over time, project future costs, identify areas that need further inquiry, observe Reserved Instance utilization, observe Reserved Instance coverage, and receive Reserved Instance recommendations.
- AWS Trusted Advisor – Get real-time identification of potential areas for optimization.
- AWS Budgets – Set custom budgets that trigger alerts when cost or usage exceed (or are forecasted to exceed) a budgeted amount. Budgets can be set based on tags and accounts as well as resource types.
- Amazon CloudWatch – Collect and track metrics, monitor log files, set alarms, and automatically react to changes in AWS resources.
- AWS CloudTrail – Log, continuously monitor, and retain account activity related to actions across AWS infrastructure at low cost.
- Amazon S3 Analytics – Automated analysis and visualization of Amazon S3 storage patterns to help you decide when to shift data to a different storage class.
- Cost Optimization Monitor (**AWS Solution**) – Automatically process detailed billing reports to get granular metrics that can be searched, analyzed, and visualized in a customizable dashboard.
- EC2 Right Sizing – Analyze EC2 instance utilization data and receive reporting recommendations for right sizing EC2 instances. The tool recommends instances that better match your usage.
- AWS Cost and Usage Report – Granular raw data files detailing your hourly AWS usage across accounts used for Do-It-Yourself (DIY) analysis (e.g., determining which S3 bucket is driving data transfer spend). The AWS Cost and Usage Report has dynamic columns that populate depending on the services you use.
- Detailed Billing Report – This report is similar to the AWS Cost and Usage Report, but it has static columns. This report is eventually deprecated. We recommend that you use the AWS Cost and Usage Report instead.

Cost optimization tools - AWS Cost Explorer, AWS Cost & Usage Report, AWS Budgets

- Get started quickly using **Cost Explorer** by exploring your data (in chart and tabular format) at a high level (for example, total costs and usage across all accounts).
- From there, you can create highly specific requests (for example, m2.2xlarge costs within account Y with the "project: secretProject" tag applied). You can save your progress as a custom report at any time.
- Examine your costs broken out by specific accounts or tags using the filtering and grouping capabilities
- If your organization requires granular information about costs and usage, or to craft a custom charge-back or show-back strategy, then enable the **AWS Cost & Usage Report**. The Cost & Usage Report contains the most comprehensive set of AWS cost and usage data available. It includes additional metadata about AWS services, pricing, and reservations
- When you have a good sense of your usage patterns and cost trends, you could decide to set budgets on your overall spending and other key cost dimensions. **AWS Budgets** gives you the ability to set custom budgets that alert you when you exceed (or are forecasted to exceed) your budget thresholds.

AWS Cost Explorer

Cost Explorer provides the foll. selection of default reports to help you pinpoint cost and usage trends.

- **Monthly costs by AWS service** – Visualize the costs and usage associated with your top five cost-accruing AWS services and get a detailed breakdown on all your services in a table view.
- **Amazon EC2 monthly cost and usage** – View all EC2 costs over the past three months, as well as current month-to-date costs.
- **Monthly costs by linked account** – View the distribution of costs across your organization. To recreate this chart, add **Linked Account** as the grouping dimension in Cost Explorer.
- **Monthly running costs** – See all running costs over the past three months and view forecasted costs for the coming month, with a corresponding confidence interval.
- **Reserved Instance (RI) reports**

To create and save personalized reports, you can use the following functionality:

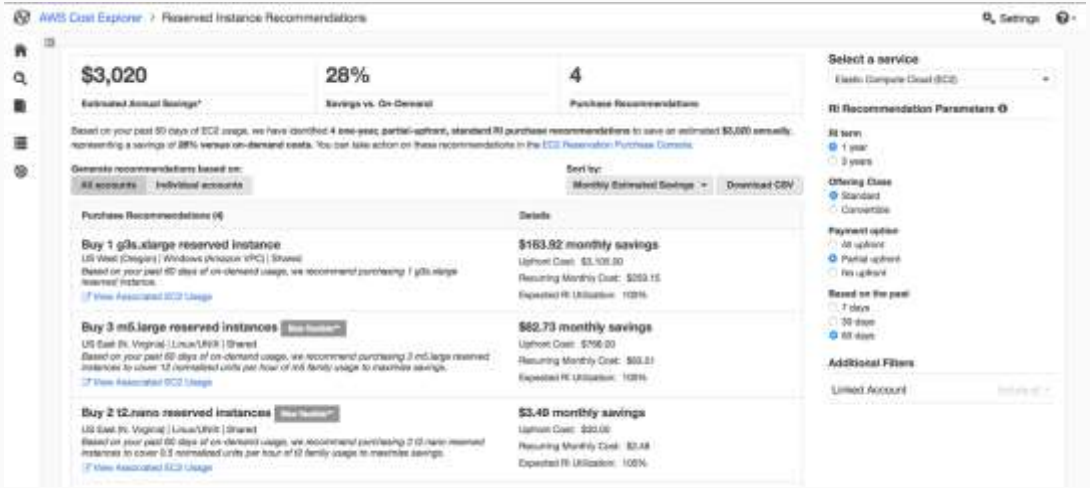
- **Set time interval and granularity** – Set a custom time interval, and determine whether you would like to view your data monthly or daily.
- **Filter/group your data** – Take advantage of filtering and grouping functionality and use a variety of available dimensions.
- **Forecast future costs and usage** – Use forecasting to get a better idea of what your costs and usage may look like in the future.

AWS Cost Explorer - Reserved Instances reporting

(RIs are currently offered by EC2, RDS, Redshift, ElastiCache, and Elasticsearch)

1. Using the **Reserved Instance reports** in **AWS Cost Explorer**
   - **RI Utilization** - % of purchased RI hours consumed by instances during a period of time
   - **RI Coverage** - discover how much of your overall instance usage is covered by RIs, so that you can make informed decisions about when to purchase or modify an RI to ensure maximum coverage

1. Access **RI purchase recommendations** via **AWS Cost Explorer**
2. Using the **AWS Cost & Usage Report** to understand your RIs
3. Track your RIs by receiving RI utilization and coverage alerts w/ **AWS Budgets**



Reserved Instances reporting - RI Utilization report

https://aws.amazon.com/blogs/enterprise-strategy/managing-your-cost-savings-with-amazon-reserved-instances/

RI Utilization report in AWS Cost Explorer -

- RI utilization is defined as the % of purchased hours that received the discounted hourly rate over a user defined period of time. Therefore, the more reservation hours you use, the greater your savings, so it is important to keep an eye on your reservation utilization.
- This report provides a summary of your reservation-related savings as compared to on-demand prices, while also helping you to visualize your utilization at a high level (e.g., your average utilization across all of your Amazon EC2 RIs).
- From there, you can drill down further into your utilization data using the available filters. Also set a custom utilization target to see how you are tracking toward your utilization goals.
- Receiving Proactive RI Utilization Alerts via AWS Budgets. (Please note that AWS Budgets is available to both master and member accounts. While master (payer) accounts are able to set up RI utilization budgets for all reservations within their organization, budgets created by member (linked) accounts track only their own utilization of the subscriptions that they own.)

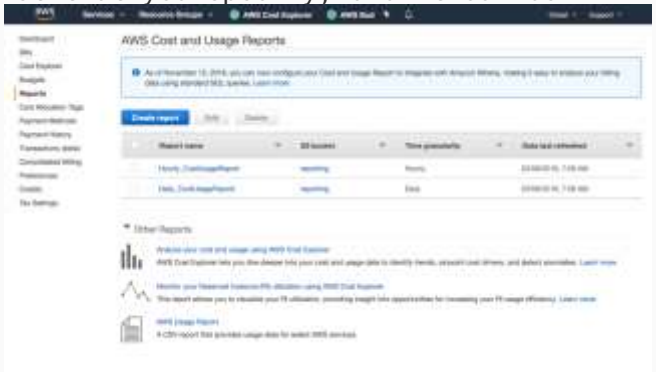
	ACCOUNT NAME	SUBSCRIPTION ID	INSTANCE TYPE	RI UTILIZATION	RI HOURS USED	RI HOURS UNUSED	NET SAVINGS
☑	Production Analytics	123456789	m4.large	66%	4,943	2,497	$175.25
☐	HRIS Database	123456789	m4.2xlarge	100%	11,160	0	$1,909.36
☐	Lead Generation...	123456789	m4.large	100%	2,976	0	$131.24

Viewing 1 to 3 of 3 Subscriptions

Cost & Usage Report

https://aws.amazon.com/blogs/enterprise-strategy/the-aws-cost-usage-report-the-next-step-on-your-cost-management-journey/

- AWS Billing Reports (which include the Cost & Usage Report and the legacy Detailed Billing Reports) are delivered at least once per day. The **most comprehensive source of information available** regarding your AWS usage is contained within the Cost & Usage Report.
- At a high level, these reports contain information about the following: a**ccounts, billing, line item details, reservations, pricing, products, and tags**.
- The Cost & Usage Reports surface the AWS usage associated with each service category used by an account and its IAM users, allowing you and your team to dive deep into your organization's usage patterns and cost trends.
- The reports can be ingested directly into Athena or Redshift, or uploaded to QuickSight for further analysis.
- The AWS Cost and Usage Report is delivered automatically to an S3 bucket that you specify, and it can be downloaded directly from there (standard S3 storage rates apply).



Advantages of C&U report over legacy Detailed Billing report ?

- Benefit #1: Comprehensive Reservation Related Information
- Benefit #2: Availability of On-Demand Pricing Information
- Benefit #3: Granular Breakdown of Discounts
- Benefit #4: Automated Data Ingestion at Scale
- Benefit #5: Cross-Product Integration

The Cost & Usage Report is best suited for organizations with complex cost management requirements, especially those who wish to establish dedicated query- or analytical-based systems in-house for cost reporting and analysis purposes. The Cost & Usage Report is also the best source of RI-related information, especially for customers who wish to view their costs in an amortized fashion.

AWS Budgets

AWS Budgets lets you set custom cost and usage budgets and receive alerts when you approach or exceed your budgeted amount. You can create budgets from the AWS Budgets Dashboard or AWS Budgets API. Budgets can track cost or usage monthly, quarterly, or yearly. You can create a budget by using the same filters available in Cost Explorer. AWS Budgets Reports allow you to create and send daily, weekly, or monthly reports to monitor the performance of your AWS Budgets.

You can monitor budgets via the Budgets Dashboard in the AWS Management Console. For both cost and usage budgets, alerts can be set against actual or forecasted budgeted values.

AWS Budgets enable you to plan your service usage, service costs, and instance reservations. Budgets provide you with a way to see the following information:

- How close your plan is to your budgeted amount or to the free tier limits
- Your usage to date, including how much you have used of your Reserved Instances (RIs)
- Your current estimated charges from AWS and how much your predicted usage will incur in charges by the end of the month
- How much of your budget has been used

AWS Budgets information is updated up to three times a day. Budgets track your unblended costs, subscriptions, refunds, and RIs. You can create the following types of budgets:

- **Cost budgets** – Plan how much you want to spend on a service.
- **Usage budgets** – Plan how much you want to use one or more services.
- **RI utilization budgets** – Define a utilization threshold and receive alerts when your RI usage falls below that threshold. This lets you see if your RIs are unused or under-utilized.
- **RI coverage budgets** – Define a coverage threshold and receive alerts when the number of your instance hours that are covered by RIs fall below that threshold. This lets you see how much of your instance usage is covered by a reservation.

Cost Optimization Best practices

Use Tagging to Organize Your Environment and Drive Accountability

https://docs.aws.amazon.com/whitepapers/latest/cost-optimization-laying-the-foundation/tagging.html

Define Mandatory Cost Tagging

An effective tagging strategy will give you improved visibility and monitoring, help you create accurate chargeback/showback models, and get more granular and precise insights into usage and spend by applications and teams. The following tag categories can help you achieve these goals:

- **Environment** – Distinguishes between development, test, and production infrastructure.
- **Application ID** – Identifies resources that are related to a specific application
- **Automation Opt-In/Opt-Out** – Indicates whether a resource should be included in an automated activity such as starting, stopping, or resizing instances.
- **Cost Center/Business Unit** – Identifies the cost center or business unit associated with a resource
- **Owner** – Used to identify who is responsible for the resource. This is typically the technical owner. If needed, you can add a separate business owner tag. You can specify the owner as an email address. Using an email address supports automated notifications to both the technical and business owners as required

Enforce Quality of Tagging

Without enforcement, tagging quality will be low, and reporting will be manual, time-consuming, and subject to debate. There are two general types of tagging enforcement: soft and hard. Soft enforcement notifies users when they have not followed policies. Hard enforcement terminates resources that are not tagged to the company standard (usually within hours after they're launched). Mature organizations find hard enforcement of tagging to be the best way to ensure that quality tagging is maintained.

Tagging Tools

- AWS Tag Editor – Finds resources with search criteria (including missing and misspelled tags) and allows you to edit tags via the AWS Management Console
- AWS Config – Identifies resources that do not comply to tagging policies
- Capital One's Cloud Custodian (open source) – Ensures tagging compliance and remediation

Right Sizing: Provisioning Instances to Match Workloads

https://docs.aws.amazon.com/whitepapers/latest/cost-optimization-right-sizing/tips-for-right-sizing-your-workloads.html

Right Size Using Performance Data:

Identify idle instances and ones that are underutilized. Key metrics to look for are CPU usage and memory usage. Identify instances with a maximum CPU usage and memory usage of less than 40% over a four-week period. These are the instances that you will want to right size to reduce costs.

Right Size Based on Usage Needs:

- **Steady state** vs **Variable, but predictable**
- **Dev/test/production** – turned off during evenings, weekends, and holidays. (You'll need to rely on tagging to identify dev/test/production instances.)
- **Temporary** – Use EC2 Spot Instance instead of using an On-Demand Instance.

Right Size by Turning Off Idle Instances:

- Terminate instead of stopping EC2 instances (EBS volumes remain if stopped)
- Stop dev test environments

Right Size by Selecting the Right Instance Family:

A good, general rule for EC2 instances is that if your maximum CPU and memory usage is less than 40% over a four-week period, you can safely cut the machine in half. For example, if you were using a c4.8xlarge EC2, you could move to a c4.4xlarge, which would save $190 every 10 days. When migrating to a different instance family, make sure the current instance type and the new instance type are compatible in terms of **virtualization type** (PV AMI vs HVM), **network** (VPC vs EC2-Classic), and **platform** (e.g. 32 bit AMIs)

Right Size Your Database Instances:

- Storage and instance type are decoupled.
- Increase allocated storage space or improve performance by changing the storage type (gp2 to iot1)
- make sure you have the correct licensing in place (if BYOL)
- apply it immediately or during the maintenance window

Using EC2 Spot instances

Unlike Reserved Instances, Spot Instances do not require an upfront commitment. However, because Spot Instances can be terminated if the Spot price exceeds your maximum price or if no capacity is available for the instance type you've specified, they are best for flexible workloads.

Workloads that constantly save data to persistent storage–including S3, EBS, EFS, DynamoDB, or RDS–can work effectively with Spot Instances.

Spot Instances are typically used to supplement On-Demand Instances, where appropriate, and are not meant to handle 100% of your workload. However, you can use all Spot Instances for any stateless, non-production application, such as development and test servers, where occasional downtime is acceptable. They are not a good choice for sensitive workloads or databases.

The Spot price is determined by long-term trends in supply and demand for EC2 spare capacity. You pay the Spot price that's in effect at the beginning of each instance-hour for your running instance, billed to the nearest second.

With Spot Instances, you never pay more than the maximum price you specify. If the Spot price exceeds your maximum price for a given instance or if capacity is no longer available, your instance will automatically be terminated (or be stopped/hibernated, if you opt for this behavior on persistent request).

Spot offers three features to help you better track and control when Spot Instances run and terminate (or stop/hibernate).

- **Termination notices** – issued two minutes prior to interruption.
- **Persistent requests** – You can opt to set your request to remain open so that a new instance will be launched in its place when the instance is interrupted. You can also have your Amazon EBS-backed instance stopped upon interruption and restarted when Spot has capacity at your preferred price.
- **Block durations** – If you need to execute workloads continuously for 1–6 hours

Thanks!