Well-Architected Bootcamp 2020 Taipei
Cost Optimization Pillar

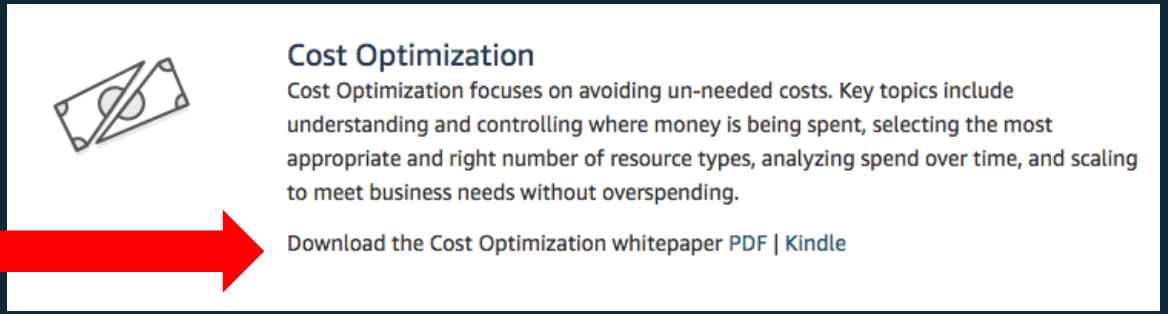
Bob Yeh, Solutions Architect, Amazon Web Services

Cost Optimization focuses on avoiding un-needed costs. Key topics include understanding and controlling where money is being spent, selecting the most appropriate and right number of resource types, analyzing spend over time, and scaling to meet business needs without overspending.

Resources:

Website:
https://aws.amazon.com/architecture/well-architected/

Whitepaper



Resources:

HTML Version of Performance Pillar:

https://wa.aws.amazon.com/wat.pillar.costOptimization.en.html

Design Principal:

- Adopt Consumption Model
- Measure Overall Efficiency
- Stop spending on On-Premise-DC
- Analyze and Attribute Expenditure
- Adopt Managed Service

Definitions:

- Cost Effective
- Match Supply and Demand
- Expenditure Awareness
- Optimizing Over Time

Definitions:

- Cost Effective
- Match Supply and Demand
- Expenditure Awareness
- Optimizing Over Time

Cost Effective

- Appropriate Provisioning
  - Over-estimated? Under-estimated?
- Right Sizing
  - Instance Type? Instance Family? Database Type? Storage Type?
- Purchase Options
  - On Demand? Reserved Instance? Spot Instance?
  - All Upfront? No Upfront? Partial Upfront?
- GEO Selection
  - Centralized? Distributed?
- Managed Services
  - Self managed cost? Human resource cost?

aws

Cost Effective

- Appropriate Provisioning
  - Based on Cloudwatch and historical data
- Right Sizing
  - Cloudwatch, Logs, Trusted Advisor
  - Monitoring reflect end-users' experience
- Purchase Options
  - Cost Explorer
  - Hybrid combination: On Demand + Reserved Instance + Spot Instance
- GEO Selection
  - S3, CloudFront
- Managed Services
  - RDS, Dynamodb…
  - SES, SQS, SNS…
  - Lambda, APIGW…

Definitions:

- Cost Effective
- Match Supply and Demand
- Expenditure Awareness
- Optimizing Over Time

Match supply and demand

- Demand-Based:
    - How quickly you need to provision
    - Understand the size of margin between supply and demand
- Buffer-Based:
    - Components run at different rates over time
- Time-Based:
    - Align the resource capacity to predictable timeframe in real-world
    - Office hours, Weekdays, Campaign, Super Hot Selling Holiday…

Match supply and demand

- Demand-Based:
  - Auto-Scaling, Pre-built AMI
  - Load-Balacing
  - CloudWatch, CloudWatch Alert/Event/Trigger
- Buffer-Based:
  - SQS
  - Kinesis
  - Sport Instances
  - Lambda
- Time-Based:
  - Auto-Scaling
  - CloudFormation

Definitions:

- Cost Effective
- Match Supply and Demand
- Expenditure Awareness
- Optimizing Over Time

Expenditure Awareness

- Stakeholders:
  - CFO/Financial Controllers
  - BU Owners
  - Tech Lead
  - 3rd Parties
- Visibility and Controls:
  - I want to estimate & forecast billing
  - I want to receive Alert if exceed threshold
  - I want to analyze spending/usage
  - I want to know RI Utilization/Coverage

Expenditure Awareness

- Cost Attribution
    - Account Structure (Multi-Account) varies by business reasons
    - Set specific limit to particular workloads (Sub Account? IAM user?)
    - Specific reservation for certain workloads (Central DB?)
- Tagging
- Entity Lifecycle Tracking

Expenditure Awareness

- Stakeholders/Visibility and Controls:
  - Cost Explorer
  - Billing Alarm, CloudWatch Alarm, SNS
- Cost Attribution
  - Consolidated Billing *Quota Discount
  - IAM
- Tagging
  - Resource Tag
- Entity Lifecycle Tracking
  - CloudTrail
  - Config
  - IAM

Definitions:

- Cost Effective
- Match Supply and Demand
- Expenditure Awareness
- Optimizing Over Time

Optimizing Over Time

Optimizing Over Time

- Measure, Monitor, and Improve:
  - Establish a Cost Optimization Function
  - Establish Goals and Metrics
  - Gather Insight and Perform Analysis
  - Report and Validate
- Stay Ever Green
  - Stay up-to-date with AWS
  - Check with managed services

Optimizing Over Time

- Measure, Monitor, and Improve:
  - TAM (Technical Account Manager)
  - Utilization, RI Coverage
  - Trusted Advisor
- Stay Ever Green
  - Trusted Advisor
  - AWS Blog
  - What's New At AWS

Q: How do you make sure your capacity matches but does not substantially exceed what you need?

Provision for Peak

Demand-based Approach

Buffered-Based Approach

Time-Based Approach

Q&A

Thank you!