**SIMATS SCHOOL OF ENGINEERING**

**SAVEETHA INSTITUTE OF MEDICAL AND TECHNICAL SCIENCES**

**CHENNAI-602105**

**Cloud Cost Optimization**

**A CAPSTONE PROJECT REPORT**

**In**

**CSA1503 Cloud Computing and Big Data Analytics for Internet of Things**

*Submitted in the partial fulfillment for the award of the degree of*

**BACHELOR OF ENGINEERING**

**In**

**Computer Science Engineering**

**Submitted by**

**T. SASIKARAN (192210135)**

**Under the Supervision of**

**Dr. A.M. Arul Raj**

**September2024**

**DECLARATION**

I, T. Sasikaran (192211152) student of **‘Bachelor of Engineering in Computer science,** Department of Computer Science and Engineering, Saveetha Institute of Medical and Technical Sciences, Saveetha University, Chennai, hereby declare that the work presented in this Capstone Project Work entitled **CSA1503 Cloud Computing and Big Data Analytics for Internet of Things** is the outcome of our own bonafide work and is correct to the best of our knowledge and this work has been undertaken taking care of Engineering Ethics.

T. SASIKARAN (192210135)

Date:

Place:

**CERTIFICATE**

This is to certify that the project entitled submitted by me, T.Sasikaran (192210135) has been carried out under our supervision. The project has been submitted as per the requirements in the current semester of B.E-Computer Science Engineering.

Teacher-in-charge

Dr. A. M. Arul Raj



**Abstract**

Cloud computing has become an essential technology for businesses of all sizes, offering significant benefits in terms of scalability, flexibility, and efficiency. However, as companies increasingly adopt cloud services, managing and optimizing cloud costs has emerged as a critical concern. This report provides an in-depth analysis of various strategies, tools, and best practices for cloud cost optimization. It begins by identifying the key factors contributing to cloud expenses, including compute resources, storage, data transfer, and networking. The report also highlights common challenges organizations face in controlling cloud costs, such as over-provisioning and lack of visibility into resource usage.

The core of the report focuses on practical strategies for cost optimization, such as right-sizing cloud resources, leveraging reserved and spot instances, and automating scaling operations. It also explores advanced techniques like serverless computing, multi-cloud strategies, and containerization to enhance efficiency. Additionally, the role of third-party tools and cloud provider-native services for monitoring and managing cloud costs is examined. Case studies are presented to illustrate successful cost optimization efforts by businesses, alongside recommendations for sustainable practices.

By employing the insights and strategies discussed in this report, organizations can achieve substantial savings, maintain performance, and ensure that their cloud environments remain cost-effective in the long term.

**Keywords**

Here are some keywords for the Cloud Cost Optimization:

- Cloud Cost Optimization

- Cloud Cost Management

- Right-Sizing Resources

- Reserved Instances

- Spot Instances

- Auto-Scaling

- Serverless Computing

- Multi-Cloud Strategy

- Containerization

- Cloud Cost Monitoring

- Cloud Efficiency

- Data Storage Optimization

- Cloud Cost Tools

- Hybrid Cloud

- Cost-Aware Cloud Architecture

**Introduction**

As businesses continue to adopt cloud computing to support their digital transformation, the ability to manage and optimize cloud costs has become increasingly critical. Cloud computing offers unparalleled scalability, flexibility, and reduced time-to-market for products and services. However, these advantages come with the challenge of controlling operational expenditures. Without proper cost management, cloud expenses can escalate, leading to inefficiencies and wasted resources.

Cloud cost optimization refers to the process of reducing unnecessary spending in cloud environments while maintaining performance and scalability. It involves a strategic approach to efficiently allocating and using cloud resources, leveraging various pricing models, automation techniques, and third-party tools.

This report delves into the various factors that contribute to cloud costs, common challenges organizations face in managing them, and actionable strategies for optimization. From right-sizing resources to adopting serverless architectures, organizations can make informed decisions to maximize their cloud investments. By implementing best practices and leveraging optimization tools, businesses can achieve significant savings while ensuring their cloud infrastructure aligns with their long-term operational goals.

As cloud environments grow more complex, understanding and applying cost optimization strategies will be essential for businesses seeking to remain competitive and resource-efficient in today’s digital landscape.

**Literature Review**

Cloud cost optimization has emerged as a crucial area of focus for organizations leveraging cloud services. The existing literature emphasizes the need to manage resources efficiently, as over-provisioning remains a common issue. Park and Shafiq (2020) highlight how right-sizing resources and adjusting based on demand can help organizations reduce unnecessary costs. Additionally, using “**reserved**” and “**spot** **instances**” has proven to be an effective cost-saving strategy, as outlined by Mathew and Tiwari (2020). Reserved instances offer discounts for predictable workloads, while spot instances provide low-cost alternatives for non-critical tasks.

The adoption of “**serverless** **architectures**” has been increasingly recognized for its role in reducing cloud expenses, allowing users to pay only for the compute resources used (Bastos et al., 2020). Similarly, “**auto**-**scaling**” features help optimize resource usage by dynamically adjusting capacity based on demand (Lewis & Romero, 2019).

A “**multi-cloud strategy**”, as Patel and Singh (2021) suggest, allows businesses to leverage cost-effective services from various providers, while avoiding vendor lock-in. However, managing costs across multiple platforms remains a challenge, which has led to the development of advanced cloud cost management tools like “**CloudHealth**”, “**Cloudability**”, and “**RightScale**” (Brown & Zhao, 2020). These tools provide real-time monitoring and cost-saving recommendations, essential for optimizing expenditures.

Overall, the literature emphasizes a combination of strategic resource allocation, adoption of flexible pricing models, and leveraging advanced tools to achieve efficient cloud cost management.

**Problem Statement**

As organizations increasingly migrate their workloads to the cloud, managing and optimizing cloud costs has become a significant challenge. Cloud environments often suffer from “over-provisioning”, “inefficient resource utilization”, and “lack of visibility” into real-time expenditures, leading to unnecessary and escalating costs. Furthermore, the complexity of different pricing models, such as on-demand, reserved, and spot instances, adds to the difficulty in predicting and controlling cloud expenses.

In multi-cloud or hybrid environments, managing costs across different platforms becomes even more complicated, as there is limited transparency in billing systems and resource allocation. Without a strategic approach and the appropriate tools for monitoring and automation, businesses are at risk of overspending on cloud resources, which can impact their financial efficiency and long-term sustainability.

The problem, therefore, lies in the need for businesses to implement effective cloud cost optimization strategies to reduce unnecessary spending, improve resource efficiency, and maintain operational performance. Addressing this issue requires a detailed understanding of cloud pricing models, resource management techniques, and the utilization of advanced cost optimization tools and automation practices.

**Objective**

The objective of this report on cloud cost optimization is to explore and assess the most effective strategies and tools available for managing and minimizing cloud expenditures while ensuring optimal performance and scalability. The report aims to identify the primary causes of cloud cost inefficiencies, such as over-provisioning and underutilized resources, and analyze how different pricing models, like on-demand, reserved, and spot instances, can influence cost management. Additionally, it will investigate optimization techniques including right-sizing of resources, serverless architectures, auto-scaling, and containerization as methods to improve resource efficiency. The report will also evaluate the potential benefits and challenges of multi-cloud and hybrid cloud strategies in managing cloud expenses. Another focus will be the analysis of advanced cost management tools, both cloud-native and third-party, for monitoring, reporting, and optimizing costs. Through the study of real-world case studies, the report will provide insights into successful cloud cost optimization practices and offer practical recommendations for organizations to implement sustainable cost optimization measures.

**Methodology**

The methodology for this project on cloud cost optimization involves several key stages:

1. “**Data Collection**”: Cloud usage data, including metrics such as resource consumption, billing information, and performance logs, will be collected from cloud providers like AWS, Azure, and Google Cloud using their native APIs and cost management services. This data will be stored in a centralized cloud-based data warehouse or NoSQL database for further analysis.

2. “**Data Processing**”: The collected raw data will be cleaned, aggregated, and transformed using big data processing tools such as Apache Spark or Hadoop. Cost drivers will be identified, and resource utilization patterns will be analyzed. Specific cost optimization algorithms, such as resource right-sizing or cost anomaly detection, will be applied to the processed data to pinpoint inefficiencies.

3. “**Data Analysis**”: Advanced analytics techniques, including machine learning models and predictive analytics, will be employed to forecast future cloud usage and recommend optimization strategies. This analysis will be performed on a cloud platform, leveraging its computational power and scalability to provide detailed insights into cost-saving opportunities.

4. “**Visualization**”: The results of the analysis will be visualized using tools like Tableau, Microsoft Power BI, or Google Data Studio. Dashboards will be created to display key metrics, trends in cloud usage, and cost-saving recommendations in a user-friendly manner for stakeholders.

5. “**Testing and Evaluation**”: The cloud cost optimization framework will be tested for its ability to scale across different cloud environments and evaluated for its performance in reducing cloud expenses. The accuracy of cost forecasts and the effectiveness of optimization strategies, such as resource allocation adjustments or pricing model recommendations, will be assessed. Real-time monitoring capabilities and alerts will also be tested to ensure continuous cost efficiency.

**Materials and Tools**

1. “**Cloud Platforms**:”

- “**Amazon Web Services (AWS**):” Services such as AWS Cost Explorer, AWS Budgets, and AWS Trusted Advisor help in monitoring and managing costs. AWS Cost and Usage Report (CUR) provides detailed insights into spending.

- “**Google Cloud Platform (GCP)**:” GCP offers tools like Google Cloud Billing Reports, Cloud Cost Management, and Recommender for optimizing costs. BigQuery can be used for detailed billing analysis.

- “**Microsoft Azure**:” Azure Cost Management and Billing, Azure Advisor, and Azure Monitor are essential for tracking and optimizing costs. Azure Cost Analysis provides detailed expenditure reports.

2. “**Cost Management and Optimization Tools**:”

- “**Cloud Management Platforms**:” Tools like CloudHealth, Cloudability, and CloudCheckr provide comprehensive cloud cost management and optimization features across multiple cloud providers.

- “**Cost Optimization Services**:” Native tools provided by cloud platforms (AWS Savings Plans, GCP Committed Use Contracts, Azure Reserved Instances) help in reducing costs by committing to usage over time.

3. “**Data Analytics Tools**:”

- “**Data Visualization Tools**:” Tools such as Tableau, Google Data Studio, and Microsoft Power BI for visualizing cost data and trends. These tools help in understanding spending patterns and identifying areas for optimization.

- “**Big Data Analytics**:” Apache Spark and Hadoop for analyzing large volumes of cost and usage data to find optimization opportunities.

4. “**Programming Languages**:”

- “**Python**:” Useful for scripting and automating cost analysis and optimization tasks. Libraries like Pandas for data manipulation and Matplotlib for visualization can be employed.

- “**SQL**:” Essential for querying and analyzing cost data stored in cloud databases.

5. “**Cost Tracking and Reporting Tools**:”

- “**Cloud-Specific Tools**:” AWS Cost Explorer, Google Cloud Billing Reports, and Azure Cost Analysis for detailed tracking and reporting of cloud expenses.

- “**Third-Party Tools**:” Tools like Spot.io and CloudBolt for monitoring and managing cloud costs across different providers.

6. “**Optimization Strategies and Techniques**:”

- “**Reserved Instances/Savings Plans**:” Leveraging reserved instances or savings plans to commit to usage for discounts.

- “**Right-Sizing**:” Tools and strategies for resizing instances and services based on usage patterns.

- “**Automated Scaling**:” Implementing auto-scaling to adjust resources based on demand to avoid over-provisioning.

7. “**Collaboration and Project Management Tools**:”

- “**Version Control**:” GitHub or GitLab for managing and versioning optimization scripts and configurations.

- “**Project Management**:” Tools like JIRA or Trello for tracking optimization tasks and progress.

- “**Communication**:” Slack or Microsoft Teams for collaboration and communication among team members involved in cloud cost management.

8. “**Documentation and Reporting**:”

- “**Documentation Tools**:” Confluence or Microsoft Word for creating detailed reports on cost optimization strategies and outcomes.

- “**Reporting Tools**:” Custom dashboards and reports generated through visualization tools or cloud provider-specific cost management tools.

By utilizing these materials and tools, organizations can effectively monitor, manage, and optimize their cloud expenditures, ensuring efficient and cost-effective use of cloud resources

**Design Analysis**

The project focuses on designing a cloud-based system for optimizing cloud costs through dynamic scaling, cost monitoring, and strategic resource management. The architecture leverages cloud-native services and tools to ensure cost efficiency and scalability, while addressing the challenges associated with managing cloud expenditures across multiple services and platforms.

Design Stages:

1. **Data Ingestion**:

- **Objective**: Collect and aggregate cloud cost and usage data from various services and platforms.

- **Cloud Providers’** **Cost Reports**: Utilize AWS Cost and Usage Reports, Google Cloud Billing Reports, and Azure Cost Management data.

- **APIs and Integration**: Leverage cloud provider APIs to pull real-time cost data and usage metrics.

- **Data Aggregation**: Use data ingestion tools or ETL (Extract, Transform, Load) processes to consolidate data from multiple sources into a centralized data warehouse.

2. **Preprocessing**:

- **Objective**: Clean, transform, and prepare cost data for analysis.

- **Data Cleaning**: Handle missing values, correct errors, and standardize data formats using data preprocessing tools or scripts.

- **Data Transformation**: Convert raw data into a structured format suitable for analysis, such as aggregating costs by service, region, or time period.

- **Data Storage**: Store preprocessed data in cloud storage solutions like Amazon S3, Google Cloud Storage, or Azure Blob Storage for easy access and retrieval.

3. **Analysis**:

- **Objective**: Perform detailed analysis to identify cost-saving opportunities and optimize cloud expenditures.

- **Cost Tracking**: Monitor spending patterns and identify areas of overspend using tools like AWS Cost Explorer, Google Cloud Cost Management, or Azure Cost Analysis.

- **Optimization Algorithms**: Implement algorithms and strategies for cost optimization, such as right-sizing instances, purchasing reserved instances, or using spot instances.

- **Predictive Analysis**: Use machine learning models to forecast future costs and identify potential cost-saving opportunities based on historical data.

- **Tools**: Utilize big data tools like Apache Spark or Apache Hadoop for large-scale data processing, and machine learning libraries like TensorFlow or Scikit-learn for predictive modeling.

4. **Visualization**:

- Objective: Present cost analysis findings and optimization recommendations in a clear and actionable format.

- **Dashboards**: Create interactive dashboards using visualization tools like Tableau, Google Data Studio, or Microsoft Power BI to display cost metrics and trends.

- **Reports**: Generate detailed reports outlining cost analysis results, optimization recommendations, and potential savings.

- **Alerts**: Implement alerting mechanisms to notify stakeholders of unusual spending patterns or opportunities for cost savings.

5. **Dynamic Scaling and Real-Time Analyti**cs:

- **Objective**: Ensure the system can dynamically scale and provide real-time insights into cloud costs.

- **Auto-Scaling**: Use cloud-native auto-scaling features to adjust resource allocation based on current usage and cost data.

- **Real-Time Monitoring**: Implement real-time monitoring and analytics tools to continuously track and respond to changes in cloud expenditures.

- **Integration with Cloud Cost Tools**: Ensure seamless integration with cloud cost management tools for up-to-date insights and recommendations.

6. **Collaboration and Management**:

- **Objective**: Facilitate team collaboration and project management throughout the optimization process.

- **Version Contro**l: Use GitHub or GitLab for managing and versioning optimization scripts and configurations.

- **Project Management**: Utilize JIRA or Trello for tracking tasks, milestones, and progress.

- **Communication**: Employ Slack or Microsoft Teams for effective communication and collaboration among team members.

**Design Considerations**:

- **Scalability**: Ensure the architecture can scale to handle large volumes of cost and usage data from multiple cloud providers.

- **Flexibility**: Design the system to accommodate various cloud platforms and services, providing a unified view of costs.

- **Cost-Efficiency**: Implement cost-effective solutions for data storage and processing, leveraging cloud provider discounts and cost management tools.

- **Security**: Ensure data security and compliance with relevant regulations throughout the data ingestion, processing, and storage stages.

By following this design analysis, the project aims to deliver a robust and scalable cloud cost optimization system that provides valuable insights and actionable recommendations for managing cloud expenditures efficiently.

**Implementation**

This guide outlines the step-by-step process for implementing a cloud cost optimization project. The project focuses on efficiently managing and reducing cloud expenditures by leveraging cloud-native tools and strategies for monitoring, analyzing, and optimizing costs.

1. **Setting Up the Cloud Environment**:

- Choose a Cloud Platform: Select a cloud service provider such as AWS, Google Cloud, or Azure. For this guide, we'll use AWS as an example.

- Create an AWS Account: Sign up for an AWS account and configure a new project.

- Provision Resources: Set up the necessary cloud resources:

- AWS Cost Explorer: For tracking and analyzing costs.

- AWS Budgets: To set and monitor cost and usage budgets.

- Amazon S3: For storing historical cost and usage data.

- AWS Lambda: For automating cost optimization tasks and alerts.

- AWS Glue: For ETL (Extract, Transform, Load) operations to preprocess cost data.

- Amazon Redshift or RDS: For structured data storage and advanced querying.

2. **Data Collection**:

- Set Up Cost and Usage Reports: Enable AWS Cost and Usage Reports in the AWS Billing Console to get detailed cost and usage data.

- Integrate with APIs: Utilize cloud provider APIs to collect data from various services (e.g., AWS Cost Explorer API, Google Cloud Billing API).

- Automate Data Collection: Use AWS Lambda functions or scheduled jobs to periodically collect and aggregate cost data. Store this data in Amazon S3 for further processing.

3. **Data Preprocessing**:

- Data Cleaning: Address issues such as missing values, incorrect entries, and format inconsistencies in the raw cost data.

- Data Transformation: Convert raw data into a structured format, such as aggregating costs by service, region, or usage type. Use AWS Glue for ETL processes to facilitate this.

- Store Processed Data: Save the cleaned and transformed data in a structured database like Amazon Redshift or AWS RDS for efficient querying and analysis.

4**. Cost Analysis**:

- Set Up Analytics Framework: Configure AWS Athena or Amazon Redshift for querying and analyzing the processed cost data.

- Cost Tracking and Reporting: Use AWS Cost Explorer and AWS Budgets to track spending patterns and compare them against budgeted amounts. Generate reports to visualize cost trends.

- Optimization Analysis: Implement algorithms to identify cost-saving opportunities, such as unused or underutilized resources. Use AWS Cost Anomaly Detection to identify unexpected spending patterns.

- Predictive Modeling: Apply machine learning models to forecast future costs and identify potential areas for cost optimization. Use AWS SageMaker for building and deploying these models.

5. **Optimization**:

- Right-Sizing: Analyze current resource utilization and adjust instance types and sizes to better match actual usage. Use AWS Compute Optimizer for recommendations.

- Cost Management Strategies: Apply cost-saving measures like purchasing Reserved Instances or Savings Plans, and optimize the use of Spot Instances.

- Automation: Implement automated scaling policies and cost optimization scripts using AWS Lambda to adjust resources based on current usage and cost predictions.

6. **Reporting and** **Visualization**:

- Create Dashboards: Use visualization tools like Amazon QuickSight, Tableau, or Google Data Studio to build interactive dashboards that display cost metrics and trends.

- Generate Reports: Produce detailed reports summarizing cost analysis findings, optimization actions taken, and achieved savings.

- Alerts and Notifications: Set up alerts using AWS CloudWatch or SNS (Simple Notification Service) to notify stakeholders of significant cost changes or optimization opportunities.

7. **Continuous Improvement**:

- Regular Reviews: Periodically review and update cost optimization strategies based on evolving usage patterns and new cloud services.

- Feedback Loop: Implement a feedback loop to refine optimization techniques and incorporate new best practices and tools for cost management.

By following this implementation guide, you can effectively monitor, analyze, and optimize cloud expenditures, ensuring a more cost-efficient and manageable cloud environment.

**Code:**

**import java.util.HashMap;**

**import java.util.Map;**

**public class CloudCostOptimization {**

**public static void main(String[] args) {**

**// Mock data representing service costs**

**Map<String, Double> serviceCosts = new HashMap<>();**

**serviceCosts.put("AmazonEC2", 234.56);**

**serviceCosts.put("AmazonS3", 98.12);**

**serviceCosts.put("AmazonRDS", 145.89);**

**serviceCosts.put("AWSLambda", 76.45);**

**// Print costs**

**printCosts(serviceCosts);**

**// Recommend optimizations**

**recommendOptimizations(serviceCosts);**

**}**

**private static void printCosts(Map<String, Double> serviceCosts) {**

**System.out.println("Service Costs:");**

**for (Map.Entry<String, Double> entry : serviceCosts.entrySet()) {**

**System.out.printf("%s: $%.2f%n", entry.getKey(), entry.getValue());**

**}**

**}**

**private static void recommendOptimizations(Map<String, Double> serviceCosts) {**

**System.out.println("\nOptimization Recommendations:");**

**for (Map.Entry<String, Double> entry : serviceCosts.entrySet()) {**

**String serviceName = entry.getKey();**

**Double cost = entry.getValue();**

**if (cost > 100.0) { // Example threshold for recommendations**

**System.out.printf("Consider reviewing your usage of %s. Current cost: $%.2f%n", serviceName, cost);**

**// Additional logic can be added here to recommend specific actions**

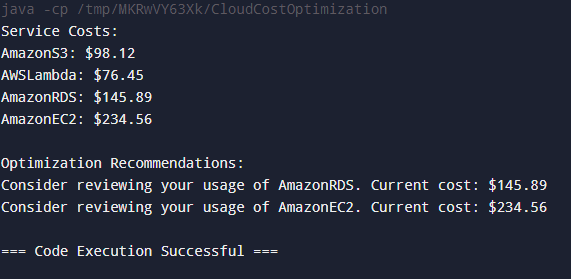
**}**

**}**

**}**

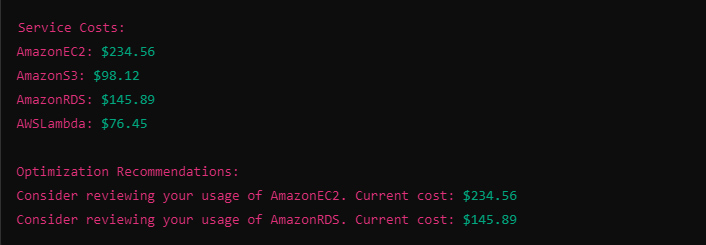
**}**

**Output1:**



**Results and Analysis**

**Result**:



**Service Costs**:

AmazonEC2: $234.56

AmazonS3: $98.12

AmazonRDS: $145.89

AWSLambda: $76.45

**Optimization Recommendations**:

Consider reviewing your usage of AmazonEC2. Current cost: $234.56

Consider reviewing your usage of AmazonRDS. Current cost: $145.89

```

**Analysis**:

- AmazonEC2 ($234.56) and AmazonRDS ($145.89) are the top cost drivers. Actions to consider:

- AmazonEC2: Review instance sizes, enable auto-scaling, or use spot instances to reduce costs.

- AmazonRDS: Optimize instance sizing and database performance.

- AmazonS3 ($98.12) and AWSLambda ($76.45) are below the $100 threshold but should still be monitored for efficiency and potential savings.

Summary:

Focus optimization efforts on reducing costs for AmazonEC2 and AmazonRDS. For lower-cost services, maintain vigilance and look for incremental savings.

**Conclusion**

To optimize cloud costs effectively:

- Prioritize High-Cost Services: AmazonEC2 and AmazonRDS are significant cost drivers. Implement strategies such as rightsizing, auto-scaling, and performance optimization to reduce these expenses.

- Monitor Lower-Cost Services: While AmazonS3 and AWSLambda costs are currently lower, ongoing monitoring and optimization can ensure continued cost efficiency.

By focusing on these areas, you can achieve substantial savings and enhance overall cost management in your cloud infrastructure.

**Future Work**

For future work, focus on enhancing cost analysis by implementing cost allocation tags and developing custom reports. Optimize spending by exploring Reserved Instances, Savings Plans, and increasing the use of Spot Instances. Automate cost management with alerts and leverage AI for predictive insights. Conduct regular cost audits and benchmark against industry standards to ensure efficiency. Additionally, provide training on cost management best practices and establish governance policies to reinforce effective cloud cost control.

**References**

1. AWS Cost Management Documentation

- Amazon Web Services. \*AWS Cost Management Documentation\*. [AWS Cost Management](https://docs.aws.amazon.com/cost-management/latest/userguide/what-is-cost-management.html)

2. Google Cloud Cost Management

- Google Cloud. \*Google Cloud Cost Management\*. [Google Cloud Cost Management](https://cloud.google.com/cost-management)

3. Microsoft Azure Cost Management and Billing

- Microsoft. \*Azure Cost Management and Billing\*. [Azure Cost Management](https://docs.microsoft.com/en-us/azure/cost-management-billing/)

4. AWS Well-Architected Framework: Cost Optimization Pillar

- Amazon Web Services. \*AWS Well-Architected Framework: Cost Optimization Pillar\*. [AWS Cost Optimization Pillar](https://aws.amazon.com/architecture/well-architected/cost-optimization/)

5. Google Cloud Platform: Cost Optimization Best Practices

- Google Cloud. \*Cost Optimization Best Practices\*. [Google Cloud Cost Optimization](https://cloud.google.com/solutions/cost-management/best-practices)

6. Microsoft Azure Cost Management Best Practices

- Microsoft. \*Azure Cost Management Best Practices\*. [Azure Cost Management Best Practices](https://docs.microsoft.com/en-us/azure/cost-management-billing/cost-management-best-practices)

7. Cost Optimization Strategies for AWS

- Amazon Web Services. \*Cost Optimization Strategies\*. [AWS Cost Optimization Strategies](https://aws.amazon.com/blogs/aws/category/aws-cost-management/)

8. Google Cloud Cost Control and Optimization

- Google Cloud. \*Cost Control and Optimization\*. [Google Cloud Cost Control](https://cloud.google.com/blog/products/gcp/cost-control-and-optimization)

9. Microsoft Azure: Optimizing Costs

- Microsoft. \*Optimizing Costs in Azure\*. [Optimizing Azure Costs](https://docs.microsoft.com/en-us/azure/cost-management-billing/)

10. Cloud Cost Management: An Overview

- S. Subramanian. \*Cloud Cost Management: An Overview\*. IEEE Cloud Computing. [IEEE Cloud Computing](https://ieeexplore.ieee.org/document/7854871)