CloudForge Cost Optimization Guide

Document ID: PRD-CF-030 **Last Updated:** 2024-03-01 **Owner:** CloudForge Product Team **Classification:** Public

---

Overview

CloudForge provides built-in cost optimization features to help you reduce cloud spending without sacrificing performance. This guide covers cost analysis, recommendations, and optimization strategies.

Cost Dashboard

Accessing Cost Data

1. Go to **Dashboard → Cost Management**
2. View current month spending
3. Analyze trends and forecasts
4. Drill down by service, environment, team

Dashboard Views

Overview: - Total spend (MTD, projected) - Spend by cloud provider - Top spending resources - Cost trends

By Service: - Compute costs - Storage costs - Network costs - Database costs

By Environment: - Production vs non-production - Per-environment breakdown - Environment efficiency scores

By Team: - Cost allocation by team - Budget vs actual - Per-user costs

---

Cost Recommendations

Automatic Recommendations

CloudForge analyzes your infrastructure and provides actionable recommendations:

Rightsizing

What: Resize over-provisioned resources

Example Recommendation:

```
Instance i-abc123 (web-server-1)
Current: m5.2xlarge ($280/month)
Recommended: m5.large ($70/month)
Savings: $210/month (75%)
Reason: Average CPU 15%, Memory 22%
```

How to Apply: 1. Review recommendation details 2. Click **Apply** to resize 3. Or schedule during maintenance window

Reserved Instances / Savings Plans

What: Commit to usage for discounts

Example Recommendation:

```
Savings Plan Opportunity
Current on-demand spend: $5,000/month
With 1-year commitment: $3,500/month
Savings: $1,500/month (30%)
Coverage: 70% of compute
```

How to Apply: 1. Review commitment terms 2. Choose term (1-year or 3-year) 3. Approve purchase through CloudForge

Unused Resources

What: Identify and remove idle resources

Example Recommendation:

```
Unused Resources Detected:
- 3 unattached EBS volumes ($45/month)
- 2 idle load balancers ($36/month)
- 5 stale snapshots ($12/month)
Total savings: $93/month
```

How to Apply: 1. Review unused resources 2. Verify they're truly unused 3. Click **Delete** or **Archive**

Storage Optimization

What: Move data to appropriate storage tiers

Example Recommendation:

```
Storage Tier Optimization
Bucket: logs-archive
Current: S3 Standard ($500/month)
Recommended: S3 Glacier ($50/month)
Savings: $450/month (90%)
Reason: No access in 180+ days
```

Optimization Strategies

Compute Optimization

1. Right-size instances:

```yaml
# cloudforge.yaml
resources:
  - type: instance
    name: web-server
    size: m5.large  # Start small
    autoscaling:
      enabled: true
      min: 2
      max: 10
      target_cpu: 70  # Scale at 70% CPU
```

2. Use spot instances for non-critical workloads:

```yaml
nodeGroups:
  - name: batch-workers
    capacityType: SPOT
    spotMaxPrice: "0.05"
    minSize: 0
    maxSize: 20
```

3. Schedule non-production environments:

```yaml
schedules:
  - name: dev-hours
    environments: [development, staging]
    start: "0 8 * * 1-5"   # 8 AM weekdays
    stop: "0 20 * * 1-5"   # 8 PM weekdays
    timezone: America/Los_Angeles
```

Storage Optimization

1. Lifecycle policies:

```yaml
storage:
  - name: logs
    type: s3
    lifecycle:
      - transition:
          days: 30
          storageClass: STANDARD_IA
      - transition:
          days: 90
          storageClass: GLACIER
      - expiration:
          days: 365
```

2. Compression:

```yaml
storage:
  - name: data-archive
    compression: gzip
    deduplication: true
```

3. Clean up unused snapshots:

```yaml
snapshots:
  retention:
    daily: 7
    weekly: 4
    monthly: 3
  autoCleanup: true
```

Network Optimization

1. Use VPC endpoints:

```yaml
networking:
  vpcEndpoints:
    - service: s3
    - service: dynamodb
    - service: ecr
```

2. Optimize data transfer:

```yaml
cdn:
  enabled: true
  origins:
    - type: s3
      bucket: static-assets
  caching:
    defaultTTL: 86400
```

3. Regional deployment: Deploy resources close to users to reduce latency and data transfer costs.

Database Optimization

1. Use read replicas:

```yaml
database:
  type: postgresql
  size: db.r5.large
  readReplicas:
    count: 2
    regions: [us-west-2, us-east-1]
```

2. Reserved capacity:

```yaml
database:
  reservedCapacity:
    enabled: true
    term: 1year
    paymentOption: partial_upfront
```

3. Storage autoscaling:

```yaml
database:
  storage:
    allocated: 100  # GB
    maxAllocated: 1000
    autoscaling: true
```

Budgets and Alerts

Setting Budgets

1. Go to **Cost Management → Budgets**
2. Click **Create Budget**
3. Configure:
   - Budget amount
   - Time period (monthly/quarterly)
   - Scope (all, environment, team)
4. Set alert thresholds

Alert Configuration

```yaml
budgets:
  - name: monthly-total
    amount: 50000
    period: monthly
    alerts:
      - threshold: 50
        channels: [email, slack]
      - threshold: 80
        channels: [email, slack, pagerduty]
      - threshold: 100
        channels: [email, slack, pagerduty]

  - name: production-compute
    amount: 20000
    scope:
      environment: production
      category: compute
    alerts:
      - threshold: 90
        channels: [slack]
```

Anomaly Detection

CloudForge detects unusual spending patterns:

```yaml
anomalyDetection:
  enabled: true
  sensitivity: medium  # low, medium, high
  notifications:
```

```
    - channel: slack
      webhook: ${SLACK_WEBHOOK}
    - channel: email
      recipients: [finops@example.com]
```

Cost Allocation

Tags

Tag resources for cost allocation:

```
resources:
  - type: instance
    name: web-server
    tags:
      Environment: production
      Team: platform
      CostCenter: eng-123
      Project: api-v2
```

Required Tags

Enforce tagging for cost tracking:

```
policies:
  requiredTags:
    - Environment
    - Team
    - CostCenter
  enforcement: warn  # warn or block
```

Cost Allocation Reports

Generate reports by tag:

1. Go to **Cost Management → Reports**
2. Select dimensions (tags, time, service)
3. Generate report
4. Export CSV/PDF

FinOps Integration

Showback/Chargeback

Allocate costs to teams:

```yaml
costAllocation:
  method: tags  # tags or usage
  tagKey: Team
  untaggedCosts: shared  # shared or specific team
  reports:
    frequency: monthly
    recipients: [finance@example.com]
```

Unit Economics

Track cost per unit:

```yaml
unitMetrics:
  - name: cost-per-customer
    formula: total_cost / active_customers
  - name: cost-per-api-call
    formula: api_infrastructure_cost / api_calls
  - name: cost-per-deployment
    formula: cicd_cost / deployments
```

Best Practices

Immediate Wins

1. **Delete unused resources** - Typically 5-15% savings
2. **Right-size instances** - Typically 20-40% savings
3. **Schedule non-production** - Typically 65% savings on dev/test

Medium-Term

1. **Reserved instances** - 30-60% savings on steady-state workloads
2. **Spot instances** - 60-90% savings on fault-tolerant workloads
3. **Storage tiering** - 50-90% savings on archival data

Long-Term

1. **Architecture optimization** - Serverless, containers
2. **Multi-region strategy** - Balance cost and availability
3. **FinOps culture** - Engineer awareness and accountability

---

Reporting

Standard Reports

Report	Frequency	Audience
Cost Summary	Daily	FinOps team
Budget Status	Weekly	Department heads
Optimization Report	Monthly	Engineering
Executive Summary	Monthly	Leadership

Custom Reports

Build custom reports:

1. Go to **Cost Management → Reports → Custom**
2. Select metrics and dimensions
3. Set filters and grouping
4. Schedule delivery

API Access

Cost Data API

```
# Get current month costs
curl https://api.cloudforge.novatech.com/v1/costs \
  -H "Authorization: Bearer $TOKEN" \
  -d "period=this_month"

# Get recommendations
curl https://api.cloudforge.novatech.com/v1/costs/recommendations \
  -H "Authorization: Bearer $TOKEN"
```

```
# Apply recommendation
curl -X POST https://api.cloudforge.novatech.com/v1/costs/recommendations/rec_123/apply \
  -H "Authorization: Bearer $TOKEN"
```

Related Documents: Getting Started (PRD-CF-001), Architecture Overview (PRD-CF-002), Pricing Plans (PRD-CF-015)