Cost Analysis and Optimization Opportunities

Northwind Health — Azure Environment
Assessment Period: January 2026

Current State Overview

The Northwind Health Azure environment spans three subscriptions with a combined monthly spend of approximately **$18,450**. This analysis identifies optimization opportunities totaling **$4,200–$5,800 per month**, representing a 23–31% cost reduction with minimal operational risk.

Spend Distribution by Service

Service	Monthly Cost	% of Total
Azure SQL Database	$4,850	26.3%
App Service Plans	$3,920	21.2%
AKS (Kubernetes)	$2,680	14.5%
Storage Accounts	$2,150	11.7%
Service Bus	$1,890	10.2%
App Insights / Log Analytics	$1,420	7.7%
Virtual Machines (Legacy)	$890	4.8%
Other (Network, DNS, etc.)	$650	3.5%
Total	**$18,450**	**100%**

Spend Distribution by Environment

Environment	Monthly Cost	% of Total
Production	$14,200	77.0%
Development	$3,450	18.7%
Shared Services	$800	4.3%

Optimization Opportunities Summary

Priority	Opportunity	Monthly Savings	Effort	Risk
1	Delete orphaned resources	$85	Very Low	Very Low

Priority	Opportunity	Monthly Savings	Effort	Risk
2	Remove temporary load test resources	$490	Very Low	Low
3	Reduce Log Analytics retention	$280	Low	Low
4	Downgrade Production App Service Plan	$650	Low	Medium
5	Right-size AKS cluster	$890	Medium	Medium
6	Migrate backup storage to Cool tier	$420	Low	Low
7	Implement dev environment auto-shutdown	$1,100	Medium	Low
8	Downgrade Service Bus tier	$700	Medium	Medium
9	Reserved Instances for SQL	$600	Low	Low
10	Optimize App Insights sampling	$400	Medium	Low

Total Identified Savings: $4,200–$5,800/month
Potential Annual Savings: $50,400–$69,600

Detailed Recommendations

1. Delete Orphaned Resources

Savings: $85/month
Effort: Very Low (< 1 hour)
Risk: Very Low

The following resources are orphaned (not attached to any active workload):

Resource	Type	Estimated Cost
`disk-nwh-legacy-001`	Managed Disk	$45/month
`pip-nwh-legacy-001`	Public IP	$15/month
`snap-nwh-backup-20250115`	Snapshot	$25/month
`nic-nwh-legacy-001`	Network Interface	$0 (no cost)

Action: Move to quarantine resource group for 7-day observation period, then delete after confirming no dependencies.

---

2. Remove Temporary Load Test Resources

Savings: $490/month
Effort: Very Low (< 1 hour)
Risk: Low

The resource group `rg-nwh-test-temp-eastus` contains load testing resources that are no longer needed:

Resource	Type	Estimated Cost
`app-nwh-loadtest-001`	App Service	—
`plan-nwh-loadtest-001`	App Service Plan (P1v2)	$490/month

These resources are tagged with `lifecycle=temporary` and testing was completed over 30 days ago.

Action: Confirm with owner (a.wong@northwindhealth.fake) that testing is complete, then delete the entire resource group.

---

3. Reduce Log Analytics Retention

Savings: $280/month
Effort: Low (< 2 hours)
Risk: Low

Current State	Recommended
`log-nwh-prod-001` retention: 90 days	30 days operational
Cost: ~$680/month	Cost: ~$400/month

Action:

1. Reduce retention to 30 days
2. Configure export to Storage Account (Cool tier) for long-term archival if compliance requires
3. Use Archive tier for data older than 90 days

---

4. Downgrade Production App Service Plan

Savings: $650/month
Effort: Low (1–2 hours)
Risk: Medium (requires validation)

Current	Recommended
`plan-nwh-portal-prod-001`	Same plan
SKU: P2v3 (4 vCPU, 16 GB)	SKU: P1v3 (2 vCPU, 8 GB)
Cost: ~$1,500/month	Cost: ~$850/month
Utilization: 25% CPU, 30% memory	Adequate headroom at P1v3

Action:

1. Review 30-day performance metrics to confirm utilization
2. Test workload on P1v3 in staging environment
3. Schedule scale-down during low-traffic window
4. Monitor for 48 hours post-change

Alternative: Implement autoscaling rules to scale between P1v3 (baseline) and P2v3 (peak).

---

5. Right-Size AKS Cluster

Savings: $890/month
Effort: Medium (4–8 hours)
Risk: Medium

Current	Recommended
`aks-nwh-prod-001`	Same cluster
Nodes: 3 × Standard_D4s_v3	Nodes: 2 × Standard_D4s_v3
Utilization: ~15%	Enable autoscaler (min: 2, max: 4)
Cost: ~$2,680/month	Cost: ~$1,790/month

Action:

1. Review pod resource requests and limits
2. Analyze peak utilization patterns over past 30 days
3. Enable cluster autoscaler
4. Scale node pool to 2 nodes
5. Test pod scheduling and failover scenarios

---

6. Migrate Backup Storage to Cool Tier

Savings: $420/month
Effort: Low (2–4 hours)
Risk: Low

Current	Recommended
`stnwhbackups001` tier: Hot	Tier: Cool
Access pattern: Monthly or less	Cool is optimal for infrequent access
Cost: ~$650/month	Cost: ~$230/month

Action:

1. Analyze blob access patterns via Storage Analytics
2. Create lifecycle management policy to automatically tier blobs:

    - Move to Cool after 30 days
    - Move to Archive after 180 days
3. Set default tier to Cool for new uploads

---

7. Implement Dev Environment Auto-Shutdown

Savings: $1,100/month
Effort: Medium (4–6 hours)
Risk: Low

Development resources currently run 24/7 (168 hours/week) despite actual usage of ~45 hours/week during business hours.

Affected resources:

- `plan-nwh-portal-dev-001` (B2)
- `sql-nwh-main-dev-001` and databases
- Associated services

Action:

1. Document required uptime hours with development team
2. Implement Azure Automation runbook for scheduled stop/start
3. Configure SQL auto-pause (if using Serverless) or scheduled scale-down
4. Target schedule: 7 AM – 7 PM weekdays only

---

8. Evaluate Service Bus Tier

Savings: $700/month
Effort: Medium (4–8 hours)
Risk: Medium

Current	Potential
`sb-nwh-prod-001` tier: Premium	Tier: Standard
Message volume: ~500 messages/day	Standard handles this easily
Cost: ~$1,890/month	Cost: ~$1,190/month

Important consideration: Premium tier is required for private endpoints. If network isolation is mandatory, this recommendation may not apply.

Action:

1. Confirm whether private endpoint requirement is firm
2. If not, test Standard tier in non-production
3. Plan migration during maintenance window

---

9. Reserved Instances for SQL Databases

Savings: $600/month
Effort: Low (1–2 hours)
Risk: Low (financial commitment only)

Production SQL databases are stable, long-term workloads suitable for Reserved Capacity pricing.

Current	With Reserved Capacity
Pay-As-You-Go pricing	1-year Reserved Capacity
SQL spend: ~$4,850/month	~40% discount on vCore costs
	Estimated savings: ~$600/month

Action:

1. Confirm SQL databases will remain in use for 12+ months
2. Calculate exact reserved capacity requirements
3. Purchase via Azure Portal or Enterprise Agreement

---

10. Optimize App Insights Sampling

Savings: $400/month
Effort: Medium (4–6 hours)
Risk: Low

Current	Recommended
`ai-nwh-prod-001` sampling: 100%	Adaptive sampling: 25–50%
Ingestion: ~15 GB/day	Target: ~5–8 GB/day
Cost: ~$680/month	Cost: ~$280/month

Action:

1. Review telemetry data to identify high-volume, low-value events
2. Enable adaptive sampling in Application Insights SDK
3. Add filtering for common dependency calls (e.g., health checks)
4. Adjust log levels in application code

Implementation Roadmap

Week 1 (Immediate)

- [ ] Delete orphaned resources (after validation)

- [ ] Remove load test resources
- [ ] Reduce Log Analytics retention

Quick win savings: $855/month

Week 2–4 (This Month)

- [ ] Downgrade App Service Plan
- [ ] Migrate backup storage to Cool tier
- [ ] Evaluate Reserved Instance pricing

Month 1 savings: $1,925/month

Month 2–3 (This Quarter)

- [ ] Right-size AKS cluster
- [ ] Implement dev auto-shutdown
- [ ] Evaluate Service Bus tier
- [ ] Optimize App Insights sampling

Full implementation savings: $4,200–$5,800/month

Ongoing Cost Governance

To maintain cost efficiency, we recommend:

1. **Monthly cost reviews** — Review Azure Cost Management reports monthly
2. **Azure Advisor** — Review and action recommendations quarterly
3. **Budget alerts** — Configure budget alerts at 80% and 100% thresholds
4. **Tag enforcement** — Ensure cost center tags are applied for accurate allocation
5. **Reserved Instance review** — Re-evaluate RI coverage annually

---

All cost estimates are based on Azure Pay-As-You-Go pricing as of January 2026. Actual costs may vary based on usage patterns and regional pricing.