Azure Migration Assessment

APAC Client Success Intelligence Hub

Document Version: 1.0
Date: 26 January 2026
Prepared For: Executive Vice President, AI
Classification: Internal - Strategic Planning

Executive Summary

This document assesses the viability, costs, and approach for migrating the APAC Client Success Intelligence Hub from Netlify to Microsoft Azure. The application is a production-grade Next.js platform supporting client success operations across the APAC region.

Key Findings

Dimension	Assessment
Technical Viability	High - No blocking dependencies identified
Strategic Alignment	Strong - Already using Azure AD for authentication
Cost Impact	+95% to +175% increase over current Netlify costs
Migration Effort	4-8 weeks for full migration
Risk Level	Medium - Primarily around scheduled function timing

Recommendation

Proceed with phased migration if Azure consolidation is a strategic priority. The migration is technically feasible and leverages existing Azure AD investment. However, this is a lateral move for operational consolidation, not a cost reduction or capability upgrade.

If cost efficiency is the primary driver, **remain on Netlify**.

Current State Architecture

Production Environment

Component	Technology	Status
Live URL	https://apac-cs-dashboards.com	Production
Hosting Platform	Netlify	Active
Database	Supabase PostgreSQL (AWS ap-southeast-1)	Active
Authentication	Azure AD via NextAuth.js	Active
AI Services	MatchaAI + Anthropic Claude	Active
Email	Resend	Active
Calendar Sync	Microsoft Graph API	Active

Application Statistics

Metric	Value
Lines of Code	500,000+
Custom React Hooks	100+
API Endpoints	230+
React Components	400+
Scheduled Functions	17
Database Tables	24+

Current Monthly Costs (Estimated)

Service	Provider	Monthly Cost (USD)
Hosting + Functions	Netlify Pro/Business	$19-99
Database	Supabase Pro	$25
Email	Resend	Included
AI Services	MatchaAI (Corporate)	Allocated
Total		**$44-124**

Strategic Considerations

Why Consider Azure Migration?

Driver	Relevance
Vendor Consolidation	Single cloud provider for simplified management
Azure AD Integration	Already authenticating via Azure AD; deeper integration possible
Enterprise Compliance	Azure's enterprise certifications and compliance frameworks
Microsoft 365 Synergy	Existing Outlook calendar integration via Graph API
IT Governance	Centralised billing, security policies, and access management
Support Model	Single enterprise support agreement

Why Stay on Netlify?

Factor	Consideration
Cost Efficiency	2-3x lower costs than Azure equivalent
Developer Experience	Git-push deploys, preview environments, zero-config
Production Stability	Currently running without issues
Migration Risk	Zero downtime risk by staying
Time to Value	No migration effort required

Migration Options

Option 1: Hybrid Migration (Recommended)

Migrate hosting to Azure, retain Supabase database

```
Azure App Service (Next.js) + Azure Functions (17 scheduled jobs)
                    │
                    ▼
         Supabase PostgreSQL (AWS ap-southeast-1)
```

Pros	Cons
Lower migration complexity	Cross-cloud latency (~30-50ms)
Retains proven database layer	Two cloud providers to manage
Faster migration timeline	Supabase costs continue
Preserves Supabase realtime features	

Timeline: 4-5 weeks
Monthly Cost: $217-258 USD

Option 2: Full Azure Migration

Migrate everything to Azure ecosystem

```
Azure App Service + Azure Functions + Azure Database for PostgreSQL
                    │
                    ▼
            Azure Blob Storage (replaces Supabase Storage)
```

Pros	Cons
Single cloud provider	Higher migration complexity
Same-region latency	Lose Supabase realtime subscriptions
Unified billing	Database migration risk
Enterprise support	Additional 2-3 weeks effort

Timeline: 6-8 weeks
Monthly Cost: $239-272 USD

Option 3: Azure Static Web Apps (Alternative)

Serverless-first approach closest to Netlify model

```
Azure Static Web Apps + Azure Functions (integrated)
                    │
                    ▼
         Supabase PostgreSQL (retained)
```

Pros	Cons
Closest to current Netlify model	Limited Next.js 16 ISR support
Lower compute costs	Less control over runtime
Integrated Functions	May require code adaptations
Free tier available	

Timeline: 5-6 weeks
Monthly Cost: $150-200 USD

Option 4: Status Quo

Remain on Netlify

Pros	Cons
Zero migration effort	Multiple cloud vendors
Lowest cost option	Less Azure integration
No downtime risk	Separate billing systems
Proven stability	

Timeline: N/A
Monthly Cost: $44-124 USD

Detailed Cost Analysis

Monthly Cost Comparison (USD)

Component	Netlify (Current)	Azure Option 1	Azure Option 2	Azure Option 3
Hosting	$19-99	$138-165	$138-165	$35-50
Serverless Functions	Included	$0 (free tier)	$0 (free tier)	Included
CDN / Edge	Included	$35-45	$35-45	Included
Database	$25 (Supabase)	$25 (Supabase)	$56-72	$25 (Supabase)
Secrets Management	Included	$0.15	$0.15	$0.15
Container Registry	N/A	$5	$5	N/A
Monitoring	Included	$0-10	$0-10	Included
Bandwidth (50GB)	Included	$4-8	$4-8	$4-8
Monthly Total	**$44-124**	**$217-258**	**$239-272**	**$150-200**
Annual Total	**$528-1,488**	**$2,604-3,096**	**$2,868-3,264**	**$1,800-2,400**

Cost with Reserved Instances (1-Year Commitment)

Reserved instances provide 30-40% savings on compute resources.

Option	Pay-As-You-Go	With 1-Year Reserved	Annual Savings
Option 1	$217-258/mo	$165-200/mo	$624-696
Option 2	$239-272/mo	$180-215/mo	$708-684
Option 3	$150-200/mo	$120-160/mo	$360-480

3-Year Total Cost of Ownership

Option	Year 1	Years 2-3	3-Year Total
Netlify (Current)	$1,488	$2,976	**$4,464**
Azure Option 1	$2,400 + migration	$4,800	**$7,200 + migration**
Azure Option 2	$2,580 + migration	$5,160	**$7,740 + migration**
Azure Option 3	$1,920 + migration	$3,840	**$5,760 + migration**

Migration Cost Estimate:

- Developer time: 160-320 hours @ $100-150/hr = $16,000-48,000 (one-time)
- Testing and validation: 40-80 hours = $4,000-12,000 (one-time)
- **Total Migration Investment:** $20,000-60,000

Technical Changes Required

High-Impact Changes

1. Hosting Platform Migration

Current	Target	Effort
`netlify.toml`	`Dockerfile` + GitHub Actions	8-16 hours
`@netlify/plugin-nextjs`	Azure App Service runtime	Included above
Netlify environment variables	Azure Key Vault	4-8 hours

Dockerfile Example:

```
FROM node:20-alpine AS builder
WORKDIR /app
COPY package*.json ./
RUN npm ci
COPY . .
RUN npm run build

FROM node:20-alpine AS runner
WORKDIR /app
ENV NODE_ENV=production
COPY --from=builder /app/.next/standalone ./
COPY --from=builder /app/.next/static ./.next/static
COPY --from=builder /app/public ./public
EXPOSE 3000
CMD ["node", "server.js"]
```

2. Scheduled Functions (17 Functions)

Each Netlify function must be converted to Azure Functions Timer Trigger format.

Function	Current Schedule	Conversion Effort
cse-monday-email	`0 20 * * 0`	2-3 hours
cse-wednesday-email	`0 1 * * 3`	2-3 hours
cse-friday-email	`0 4 * * 5`	2-3 hours
aged-accounts-snapshot	`0 19 * * *`	2-3 hours
aging-alerts-check	`0 21 * * *`	2-3 hours
compliance-snapshot	`0 20 * * 0`	2-3 hours
segmentation-refresh	`0 19 * * *`	2-3 hours
health-snapshot	`0 20 * * *`	2-3 hours
chasen-auto-discover	`0 18 * * *`	2-3 hours
burc-data-sync	`0 30 19 * * *`	2-3 hours
burc-alert-check	`0 30 20 * * *`	2-3 hours
graph-embed	`0 16 * * *`	2-3 hours
proactive-insights	`0 19 * * *`	2-3 hours
changelog-email	`0 1 * * 1`	2-3 hours
(3 additional)	Various	6-9 hours
Total		**34-51 hours**

Azure Function Example:

```
import { app, Timer } from "@azure/functions";

app.timer('aged-accounts-snapshot', {
    schedule: '0 19 * * *', // Same cron syntax
    handler: async (myTimer: Timer, context) => {
        const response = await fetch(`${process.env.SITE_URL}/api/cron/aged-accounts-snapshot`, {
            headers: { 'Authorization': `Bearer ${process.env.CRON_SECRET}` }
        });
        context.log(`Snapshot completed: ${response.status}`);
    }
});
```

Medium-Impact Changes

3. CI/CD Pipeline

Current	Target	Effort
Netlify auto-deploy on git push	GitHub Actions → Azure	8-12 hours

GitHub Actions Workflow:

```yaml
name: Deploy to Azure
on:
  push:
    branches: [main]

jobs:
  build-and-deploy:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@v4
        with:
          submodules: recursive

      - name: Build and push to ACR
        uses: azure/docker-login@v1
        with:
          login-server: ${{ secrets.ACR_LOGIN_SERVER }}
          username: ${{ secrets.ACR_USERNAME }}
          password: ${{ secrets.ACR_PASSWORD }}

      - run: |
          docker build -t ${{ secrets.ACR_LOGIN_SERVER }}/apac-intelligence:${{ github.sha }} .
          docker push ${{ secrets.ACR_LOGIN_SERVER }}/apac-intelligence:${{ github.sha }}

      - name: Deploy to App Service
        uses: azure/webapps-deploy@v2
        with:
          app-name: apac-intelligence
          images: ${{ secrets.ACR_LOGIN_SERVER }}/apac-intelligence:${{ github.sha }}
```

4. Security Headers

Current	Target	Effort
`netlify.toml` [[headers]]	`web.config` or Azure config	2-4 hours

5. Caching Configuration

Current	Target	Effort
Netlify Edge caching rules	Azure Front Door rules	4-6 hours

Low-Impact / No Changes Required

Component	Status	Notes
Azure AD Authentication	No change	Already Azure-native
Microsoft Graph API	No change	Already Azure-native
NextAuth.js	No change	Platform-agnostic
MatchaAI Integration	No change	External SaaS
Resend Email	No change	External SaaS
Application Source Code	No change	Framework-agnostic
Supabase Database	No change (Option 1)	Keep as-is initially

Risk Assessment

Risk Matrix

Risk	Likelihood	Impact	Mitigation
Scheduled function timing errors	Medium	High	Extensive parallel testing; run both environments for 2 weeks
Authentication flow disruption	Low	Critical	Azure AD already in use; minimal change to auth flow
Database migration data loss	Low	Critical	Multiple verified backups; only applies to Option 2
Cold start latency issues	Medium	Medium	Use Premium tier or always-on configuration
Cost overrun	Medium	Medium	Set Azure budgets and alerts; start with reserved instances
Extended timeline	Medium	Low	Build in 2-week buffer; phase the migration
Team learning curve	High	Low	Azure documentation readily available; similar concepts

Critical Success Factors

1. **Parallel Running Period** - Both Netlify and Azure must run simultaneously for

minimum 2 weeks before cutover

2. **Function Timing Validation** - All 17 scheduled functions must fire correctly in staging before production
3. **Performance Baseline** - Establish response time baselines on Netlify; Azure must meet or exceed
4. **Rollback Plan** - Ability to revert to Netlify within 1 hour if critical issues arise

---

Migration Timeline

Phase 1: Foundation (Weeks 1-2)

Task	Owner	Duration
Create Azure Resource Group	Platform Team	Day 1
Provision App Service Plan (P1v3)	Platform Team	Day 1
Set up Azure Container Registry	Platform Team	Day 1
Create Azure Key Vault	Platform Team	Day 1
Migrate all secrets to Key Vault	Platform Team	Days 2-3
Create Dockerfile	Development Team	Days 2-4
Set up GitHub Actions pipeline	Development Team	Days 3-5
Deploy to Azure (staging slot)	Development Team	Days 5-7
Configure custom domain (staging)	Platform Team	Days 7-8
Initial smoke testing	QA Team	Days 8-10

Phase 2: Functions Migration (Weeks 2-3)

Task	Owner	Duration
Create Azure Functions App	Development Team	Day 1
Convert email functions (4)	Development Team	Days 1-3
Convert snapshot functions (5)	Development Team	Days 3-5
Convert sync functions (4)	Development Team	Days 5-7
Convert remaining functions (4)	Development Team	Days 7-9
Function integration testing	QA Team	Days 9-12
Timing validation (full cycle)	QA Team	Days 12-14

Phase 3: Parallel Running (Weeks 3-4)

Task	Owner	Duration
Enable both environments	Platform Team	Day 1
Disable Netlify functions (prevent duplicates)	Platform Team	Day 1
Monitor Azure functions execution	Development Team	Ongoing
Performance comparison testing	QA Team	Days 1-7
User acceptance testing	Business Team	Days 7-10
Security review	Security Team	Days 10-12
Go/No-go decision	Leadership	Day 14

Phase 4: Cutover (Week 5)

Task	Owner	Duration
Final backup of all systems	Platform Team	Day 1
DNS migration to Azure	Platform Team	Day 1
SSL certificate verification	Platform Team	Day 1
Production monitoring (24hr)	Development Team	Days 1-2
Disable Netlify builds	Platform Team	Day 3
Post-migration validation	QA Team	Days 3-5
Documentation update	Development Team	Days 5-7

Phase 5: Optional Database Migration (Weeks 6-8)

Only if proceeding with Option 2

Task	Owner	Duration
Provision Azure PostgreSQL Flexible Server	Platform Team	Day 1
Export Supabase schema and data	Development Team	Days 1-2
Import to Azure PostgreSQL	Development Team	Days 2-3
Update connection strings	Development Team	Day 4
Migrate Supabase Storage to Azure Blob	Development Team	Days 4-6
Data validation testing	QA Team	Days 6-10
Cutover database connection	Platform Team	Day 11
Decommission Supabase	Platform Team	Day 14+

Recommendations

Primary Recommendation

Proceed with Option 1 (Hybrid Migration) if Azure consolidation is a strategic imperative.

Factor	Rationale
Why Option 1	Balances Azure benefits with migration risk
Why not Option 2	Database migration adds complexity without proportional benefit
Why not Option 3	Static Web Apps has Next.js 16 ISR limitations
Timeline	5 weeks with 1-week buffer
Budget	$20,000-35,000 migration + $200-250/month ongoing

Alternative Recommendation

Remain on Netlify if cost efficiency is the primary driver.

The current Netlify setup is:

- Stable and production-proven
- 2-3x more cost-effective
- Requires zero migration effort
- Already integrated with Azure AD

Decision Framework

If your priority is...	Then choose...
Vendor consolidation	Option 1 (Hybrid Migration)
Full Azure compliance	Option 2 (Full Migration)
Cost minimisation	Option 4 (Stay on Netlify)
Serverless architecture	Option 3 (Static Web Apps)
Fastest time to Azure	Option 1 (Hybrid Migration)
Lowest risk	Option 4 (Stay on Netlify)

Next Steps

If Proceeding with Migration

1. **Week 0:** Executive approval and budget allocation
2. **Week 0:** Assign migration team (1-2 developers, 1 platform engineer)
3. **Week 1:** Begin Phase 1 (Foundation)
4. **Week 3:** Migration checkpoint review with stakeholders
5. **Week 5:** Go/No-go decision for production cutover

If Remaining on Netlify

1. Document decision rationale for future reference
2. Review Azure migration option annually or when:
    - Netlify pricing changes significantly
    - Azure offers compelling new features
    - Compliance requirements mandate Azure

Appendices

A. Azure Resource Requirements

```
Resource Group: rg-apac-intelligence-prod
├── App Service Plan: asp-apac-intelligence (P1v3)
│   └── App Service: app-apac-intelligence
├── Function App: func-apac-intelligence
├── Container Registry: cracpacintelligence
├── Key Vault: kv-apac-intelligence
├── Front Door: fd-apac-intelligence
├── Application Insights: appi-apac-intelligence
└── Storage Account: stacpacintelligence (function storage)
```

B. Environment Variables to Migrate

Variable	Sensitivity	Azure Service
NEXTAUTH_SECRET	High	Key Vault
AZURE_AD_CLIENT_SECRET	High	Key Vault
SUPABASE_SERVICE_ROLE_KEY	High	Key Vault
DATABASE_URL	High	Key Vault
MATCHAAI_API_KEY	High	Key Vault
CRON_SECRET	High	Key Vault
RESEND_API_KEY	High	Key Vault
NEXT_PUBLIC_SUPABASE_URL	Low	App Configuration
NEXT_PUBLIC_SUPABASE_ANON_KEY	Low	App Configuration
NEXT_PUBLIC_APP_URL	Low	App Configuration
(15+ additional variables)	Various	Key Vault / App Config

C. Scheduled Functions Detail

Function	Purpose	Schedule (UTC)	Sydney Time
cse-monday-email	Week-ahead focus email	Sun 20:00	Mon 07:00
cse-wednesday-email	Mid-week status	Wed 01:00	Wed 12:00
cse-friday-email	Week wrap-up	Fri 04:00	Fri 15:00
changelog-email	Weekly changelog	Mon 01:00	Mon 12:00
aged-accounts-snapshot	Daily AR snapshot	Daily 19:00	Daily 06:00
aging-alerts-check	Threshold alerts	Daily 21:00	Daily 08:00
compliance-snapshot	Weekly compliance	Sun 20:00	Mon 07:00
segmentation-refresh	Event completion sync	Daily 19:00	Daily 06:00
health-snapshot	Health score tracking	Daily 20:00	Daily 07:00
chasen-auto-discover	AI table discovery	Daily 18:00	Daily 05:00
burc-data-sync	BURC Excel sync	Daily 19:30	Daily 06:30
burc-alert-check	BURC KPI alerts	Daily 20:30	Daily 07:30
graph-embed	Semantic embeddings	Daily 16:00	Daily 03:00
proactive-insights	AI-driven insights	Daily 19:00	Daily 06:00

D. Current vs Azure Architecture Diagram

Current Architecture (Netlify)

```
┌────────────────────────────────────────────────────────────┐
│                          USERS                              │
└─────────────────────────┬──────────────────────────────────┘
                          │
                          ▼
┌────────────────────────────────────────────────────────────┐
│                     NETLIFY EDGE CDN                        │
│               apac-cs-dashboards.com                        │
└─────────────────────────┬──────────────────────────────────┘
                          │
          ┌───────────────┼───────────────┐
          ▼               ▼               ▼
┌─────────────┐   ┌─────────────┐   ┌─────────────┐
│   Next.js   │   │  Netlify    │   │   Static    │
│   App       │   │  Functions  │   │   Assets    │
│   (SSR)     │   │  (17 cron)  │   │   (CDN)     │
└──────┬──────┘   └──────┬──────┘   └─────────────┘
```

```
       │                  │
       └────────┬─────────┘
                │
                ▼
┌──────────────────────────────────────────────────────────────┐
│                    EXTERNAL SERVICES                          │
├─────────────┬─────────────┬─────────────┬────────────────────┤
│  Supabase   │  Azure AD   │  MatchaAI   │  MS Graph API      │
│  (Database) │  (Auth)     │  (AI/LLM)   │  (Calendar)        │
└─────────────┴─────────────┴─────────────┴────────────────────┘
```

Target Architecture (Azure Option 1)

```
┌──────────────────────────────────────────────────────────────┐
│                         USERS                                 │
└─────────────────────────┬────────────────────────────────────┘
                          │
                          ▼
┌──────────────────────────────────────────────────────────────┐
│                   AZURE FRONT DOOR                            │
│               apac-cs-dashboards.com                          │
│           (CDN + WAF + SSL Termination)                       │
└─────────────────────────┬────────────────────────────────────┘
                          │
          ┌───────────────┼───────────────┐
          ▼               ▼               ▼
┌─────────────┐     ┌─────────────┐     ┌─────────────┐
│ App Service │     │   Azure     │     │    Blob     │
│   (P1v3)    │     │  Functions  │     │   Storage   │
│  Container  │     │ (17 timers) │     │  (Static)   │
└──────┬──────┘     └──────┬──────┘     └─────────────┘
       │                   │
       │          ┌────────┴────────┐
       │          ▼                 ▼
       │    ┌─────────────┐     ┌─────────────┐
       │    │  Key Vault  │     │  App Config │
       │    │  (Secrets)  │     │  (Settings) │
       │    └─────────────┘     └─────────────┘
       │
       └────────┬─────────────────────────────┐
                ▼                             ▼
┌─────────────────────────────┐    ┌──────────────────────────┐
│     EXTERNAL SERVICES       │    │     AZURE SERVICES       │
├─────────────┬───────────────┤    ├─────────────┬────────────┤
│  Supabase   │   MatchaAI    │    │  Azure AD   │ MS Graph   │
│  (Database) │   (AI/LLM)    │    │  (Auth)     │ (Calendar)│
└─────────────┴───────────────┘    └─────────────┴────────────┘
```

Document Control

Version	Date	Author	Changes
1.0	26 Jan 2026	Platform Team	Initial assessment

This document contains confidential business information. Distribution should be limited to authorised personnel involved in platform strategy decisions.