Unified Microsoft Modernization System — Recursive Integration Framework

Core Objective & Program Scope

Objective

Unify Microsoft Partner Success Expanded Benefits with FNGU providers using a DIY local hardware stack. Deliver recursive manageability, scalability, and cost-efficiency across Personal, Family, and Business domains.

Success Criteria:

- Availability (99.x%), latency (p95 targets), MTTR, data quality SLAs, and $/task.
- Demonstrated cost arbitrage via local-first routing and sponsorship/credits utilization.
- Repeatable module patterns across domains using the Module Contract.

Semantic + Workflows

Semantic & Orchestration Backbone

Semantic Kernel planners orchestrate domain agents with grounded context from Microsoft Graph. Power Automate and Logic Apps implement policy-aware workflows and connectors.

Routing Logic:
- Local-first execution with fallback to cloud based on policy (security, data location), performance (SLO), and cost thresholds.
- Recursive delegation: complex tasks decomposed into sub-tasks by planners.

Interfaces: OpenAPI contracts, Graph scopes, event triggers.

Zero Trust + Data Estate

Data, Governance & Security

Data Stores: SharePoint, OneDrive, Dataverse, SQL/Azure Storage, Cosmos DB, and vector indexes for retrieval. Zero Trust: Entra ID, Conditional Access, PIM; Defender XDR/Defender for Cloud; Purview DLP/eDiscovery/Records.

Policies:

- RBAC & least privilege; sensitivity labels and retention.
- Audit evidence via automated queries and workbooks.
- Supply-chain security: SBOM, signed artifacts, secret rotation.

Integration Fabric (FNGU)

Providers & Roles

Microsoft (core) • Apple (MDM/Dev) • Netflix (OSS) • AWS (optional compute) • Google (AI/Firebase/GenAI) • Meta (XR/Marketing) • NVIDIA (GPU) • Broadcom (Infra/Sec) • ServiceNow (ITSM) • CrowdStrike (Telemetry).

Contracts per Provider: APIs (auth, rate limits), program credits/sponsorships, data boundaries, and risk register.

Local-First Compute

DIY Hardware Layer (Recursive)

Primary Node: Ryzen 9/i9, 64–128 GB RAM, 2TB NVMe + 8TB HDD, RTX 3090/A6000.
Secondary Nodes: Raspberry Pi/NUCs; NAS (TrueNAS/Synology); UPS + efficient cooling.

Operating Model:
- Modular agents placed per node.
- Energy-aware scaling and scheduled batch windows.
- Resilience with local failover and cloud fallback.

Ship, See, Save

DevOps, Observability & Cost

DevOps: GitHub Repos/Actions/Environments; IaC (Bicep/Terraform); release rings; SBOM and signed artifacts.

Observability: Azure Monitor, Log Analytics, Sentinel; SLOs with error budgets and alerting.

FinOps: Power BI dashboard tracking unit economics, credits burn-down, and routing thresholds.

Recursive Operating Model (At-a-Glance)

How Recursion Works

Domain agents (Personal, Family, Business) plan and act via Semantic Kernel; workloads flow between cloud and local hardware under policy, performance, and cost constraints.

Cycle: Plan → Build → Operate → Optimize → Plan.

Metrics: p95 latency, success rate, MTTR, $/task; governance evidence auto-collected.