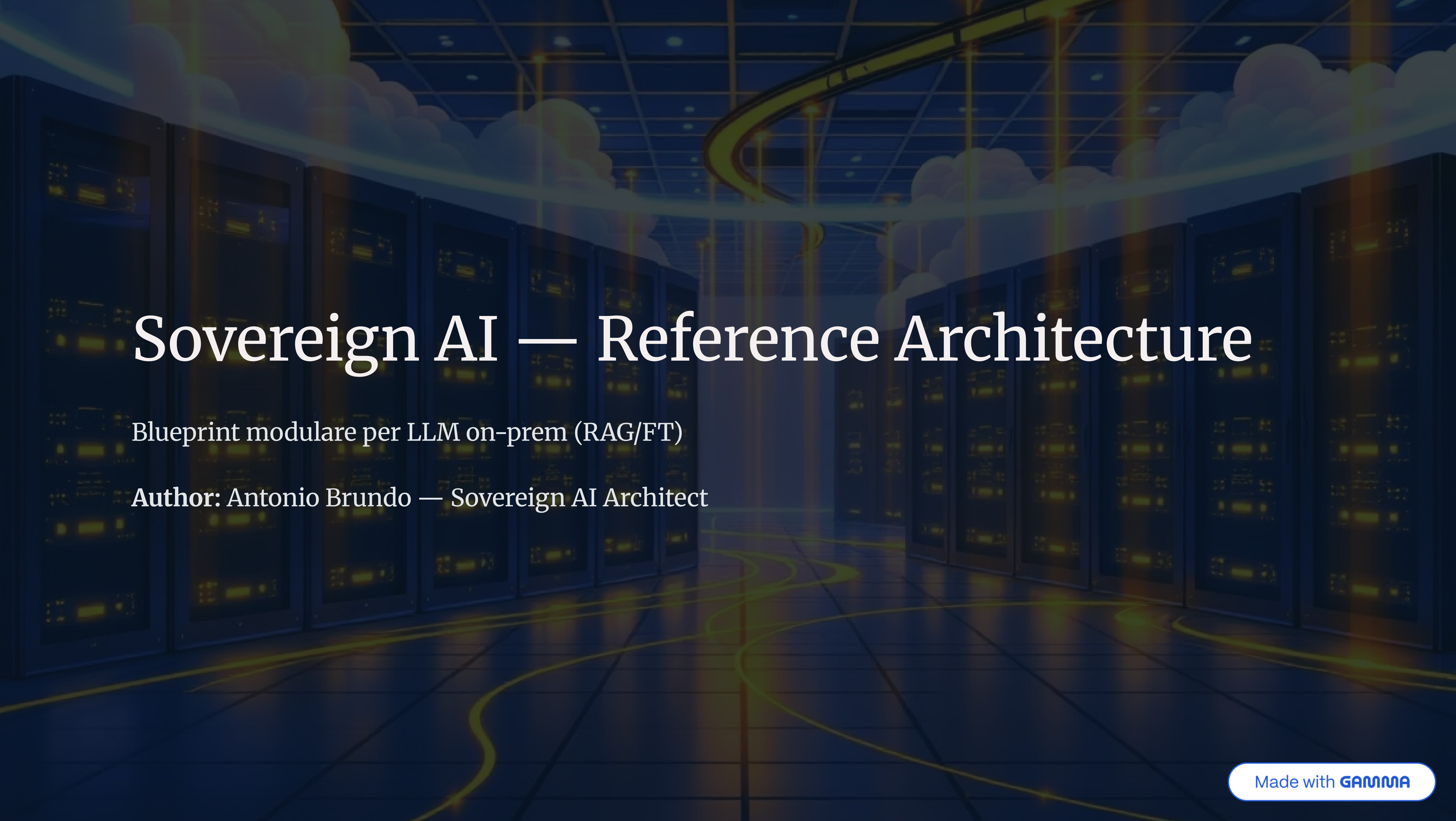


Sovereign AI — Reference Architecture

Blueprint modulare per LLM on-prem (RAG/FT)

Author: Antonio Brundo — Sovereign AI Architect

Architecture Overview & Strategic Vision

This reference architecture represents a paradigm shift towards sovereign artificial intelligence deployment, designed specifically for organisations requiring complete data control, predictable costs, and audit-ready compliance. Unlike cloud-based solutions that introduce latency variability and regulatory uncertainties, this on-premises approach delivers consistent performance whilst maintaining full data sovereignty.

The architecture emphasises three core principles: latency optimisation through intelligent caching and proximity-based processing, comprehensive audit trails for regulatory compliance, and predictable cost structures that eliminate the unpredictability of consumption-based cloud pricing models.

Built around a modular zone-based design, the architecture supports both Retrieval-Augmented Generation (RAG) and Fine-Tuning (FT) workloads across multiple deployment scales. Each zone operates with defined network boundaries and security controls, enabling granular access management whilst facilitating seamless data flow between components.

The design accommodates diverse organisational requirements, from small-scale pilot deployments to enterprise-grade air-gapped environments. This flexibility ensures that organisations can begin their sovereign AI journey with minimal infrastructure investment whilst maintaining a clear path to full-scale production deployment.

Latency Optimisation

Sub-300ms response times with intelligent caching and proximity processing

Audit Readiness

Comprehensive logging and compliance frameworks built-in from day one

Cost Predictability

Transparent unit economics with clear CAPEX/OPEX allocation models

System Architecture Diagram

The architectural diagram illustrates the complete system topology, organised into eight distinct zones (Z0-Z7) each serving specific functions within the sovereign AI ecosystem. The colour-coded zones represent different security contexts and operational boundaries, with clearly defined data flow paths between components.

Edge connections represent the three primary flow types: request processing (blue), data ingestion (green), and operational management (orange). This visual representation enables architects and engineers to quickly understand system dependencies and identify potential bottlenecks or security boundaries.

01

User Interface Layer (Z0)

Web interfaces and API gateways providing secure access points

02

Application Services (Z1)

Business logic and orchestration services managing user requests

03

AI Processing Core (Z2-Z4)

LLM inference engines, vector databases, and processing clusters

04

Data & Storage (Z5-Z6)

Document stores, training data repositories, and backup systems

05

Operations & Monitoring (Z7)

Observability stack, logging systems, and administrative tools

Zone Definitions & Component Architecture

Zone ID	Name	Purpose	Network Boundaries
Z0	DMZ/Edge	Public-facing interfaces and load balancers	Internet-facing, filtered ingress
Z1	Application	Business logic and API orchestration	Internal network, authenticated access
Z2	AI Inference	LLM processing and model serving	Isolated compute cluster
Z3	Vector Store	Embedding storage and retrieval	Database network segment
Z4	Training	Model fine-tuning and adaptation	High-performance compute zone
Z5	Data Lake	Raw data storage and processing	Storage network with encryption
Z6	Metadata	System configuration and catalogues	Administrative network
Z7	Operations	Monitoring, logging, and management	Management network, privileged access

Each zone implements specific security controls and operational procedures designed to minimise attack surfaces whilst enabling efficient data flow. The zone-based architecture allows for granular access control, simplified compliance auditing, and clear separation of concerns across the entire system.

Network Segmentation

Micro-segmentation with software-defined perimeters ensuring zero-trust networking principles

Component Isolation

Containerised services with resource quotas and security context constraints

Data Flow Control

Controlled inter-zone communication with encrypted channels and audit logging

System Flows & Data Processing Patterns

Request Flow

User Query Processing

1. Authentication & authorisation
2. Query parsing & intent classification
3. Context retrieval from vector store
4. LLM inference with RAG augmentation
5. Response post-processing & delivery

Target SLO: p95 ≤ 300ms end-to-end

Ingestion Flow

Data Pipeline Management

1. Document intake & validation
2. Content extraction & parsing
3. Chunking & embedding generation
4. Vector index updates
5. Metadata catalogue updates

Target SLO: <1 hour batch processing

Operations Flow

System Management

1. Health monitoring & alerting
2. Performance metrics collection
3. Security event correlation
4. Capacity planning updates
5. Compliance report generation

Target SLO: Real-time observability

These three primary flows represent the core operational patterns within the sovereign AI architecture. Each flow incorporates failure modes and recovery mechanisms, ensuring system resilience and maintaining service availability even during component failures or maintenance windows.

Request Processing

Real-time user interactions with sub-second response times

Data Ingestion

Batch processing of documents and knowledge base updates

Operations

Continuous monitoring and system health management

Service Level Objectives & Performance Metrics

300ms

Chat Response Time

95th percentile latency for conversational interactions

30ms

Vector Query Speed

Maximum retrieval time for similarity search operations

100

Requests Per Second

Sustained throughput capacity for medium deployment

99.9%

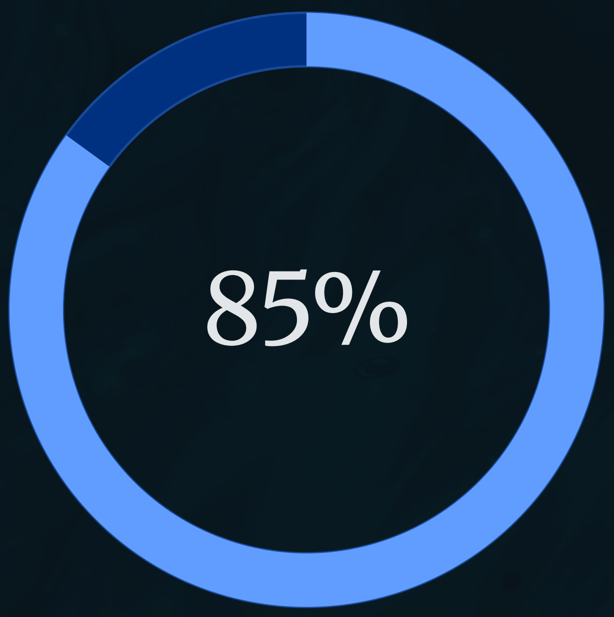
System Availability

Minimum uptime guarantee for production environments

The performance characteristics of sovereign AI systems require careful balance between computational efficiency and response quality. These SLOs represent achievable targets based on current hardware capabilities and optimisation techniques, whilst maintaining room for future improvements as technology evolves.

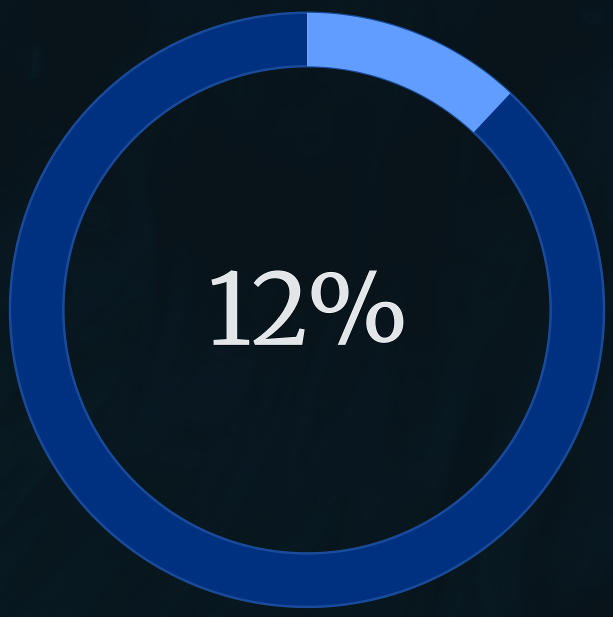
> **Cost Per 1K Tokens Formula:**
> `((CAPEX/hour + OPEX/hour) / tokens_per_hour) * 1000`
> *Where CAPEX includes amortised hardware costs and OPEX covers operational expenses including power, cooling, and maintenance*

This transparent pricing model enables organisations to predict costs accurately and compare the total cost of ownership against cloud-based alternatives. The formula accounts for both capital expenditure amortisation and operational expenses, providing a comprehensive view of true system costs.



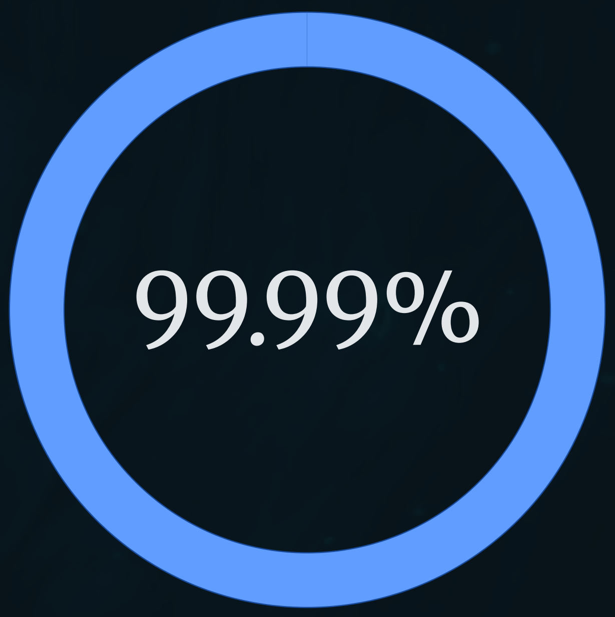
GPU Utilisation Target

Optimal resource allocation for cost efficiency



Memory Overhead

System overhead for optimal performance



Enterprise SLA

Maximum availability for mission-critical deployments

Security & Compliance Framework

The security architecture implements a comprehensive defence-in-depth strategy, addressing both technical controls and compliance requirements. Each zone maintains specific security postures aligned with data classification levels and access requirements, ensuring that sensitive information remains protected throughout its lifecycle.

Security Control	Z0-Z1	Z2-Z4	Z5-Z6	Z7	Compliance Impact
Identity & Secrets	OAuth/SAML	Service Mesh	Vault	Privileged	GDPR Art. 32
Encryption	TLS 1.3	AES-256	At-rest	E2E	NIS2 Directive
Logging & Audit	Access logs	Inference logs	Data lineage	SIEM	SOX Compliance
Egress Control	WAF	Air-gapped	DLP	Monitored	Data Localisation
Prompt Security	Validation	Sanitisation	N/A	Analysis	AI Act
Backup & DR	Config	Models	Full	Logs	BC/DR Standards

Regular security assessments and penetration testing validate the effectiveness of implemented controls. The framework supports continuous compliance monitoring through automated policy enforcement and real-time violation detection, reducing the administrative burden of manual compliance activities.

Data Protection

End-to-end encryption with key rotation and secure enclaves

Access Control

Zero-trust architecture with continuous authentication

Audit Trail

Immutable logging with tamper-evident storage

Compliance

GDPR, NIS2, and AI Act readiness with automated reporting

Deployment Scenarios & Sizing Guidelines

Small Deployment

Target: 10-25 RPS
Models: 7B parameters (Llama2/Mistral)
Hardware: 2x A100 GPUs, 256GB RAM
Use Case: Pilot projects and proof-of-concept

- Single-node deployment with containerisation
- Basic monitoring and logging capabilities
- Development and testing workloads

Medium Deployment

Target: 50-100 RPS
Models: 13-30B parameters
Hardware: 4x A100 GPUs, 512GB RAM
Use Case: Production departmental applications

- Multi-node cluster with load balancing
- Full security controls and compliance
- Business-critical applications support

Large Deployment

Target: 200+ RPS
Models: 70B+ parameters
Hardware: 8x H100 GPUs, 1TB+ RAM
Use Case: Enterprise-wide AI services

- Distributed architecture with auto-scaling
- Advanced monitoring and observability
- Multi-tenancy and resource isolation

Air-Gapped

Target: Variable based on classification
Models: Custom fine-tuned models
Hardware: Hardened infrastructure
Use Case: Classified or highly sensitive data

- Complete network isolation
- Enhanced physical security controls
- Specialised update and maintenance procedures

Each deployment scenario balances performance requirements against infrastructure investment and operational complexity. The modular architecture enables organisations to start with smaller deployments and scale incrementally as usage grows and requirements evolve.

Adoption Checklist & Implementation Roadmap

Pre-Pilot Requirements

Infrastructure Readiness

- Hardware procurement and installation
- Network segmentation implementation
- Security controls deployment
- Monitoring infrastructure setup
- Backup and disaster recovery testing

Organisational Preparation

- Team training and skill development
- Security policy documentation
- Compliance framework alignment
- Change management processes

Go-Live Checklist

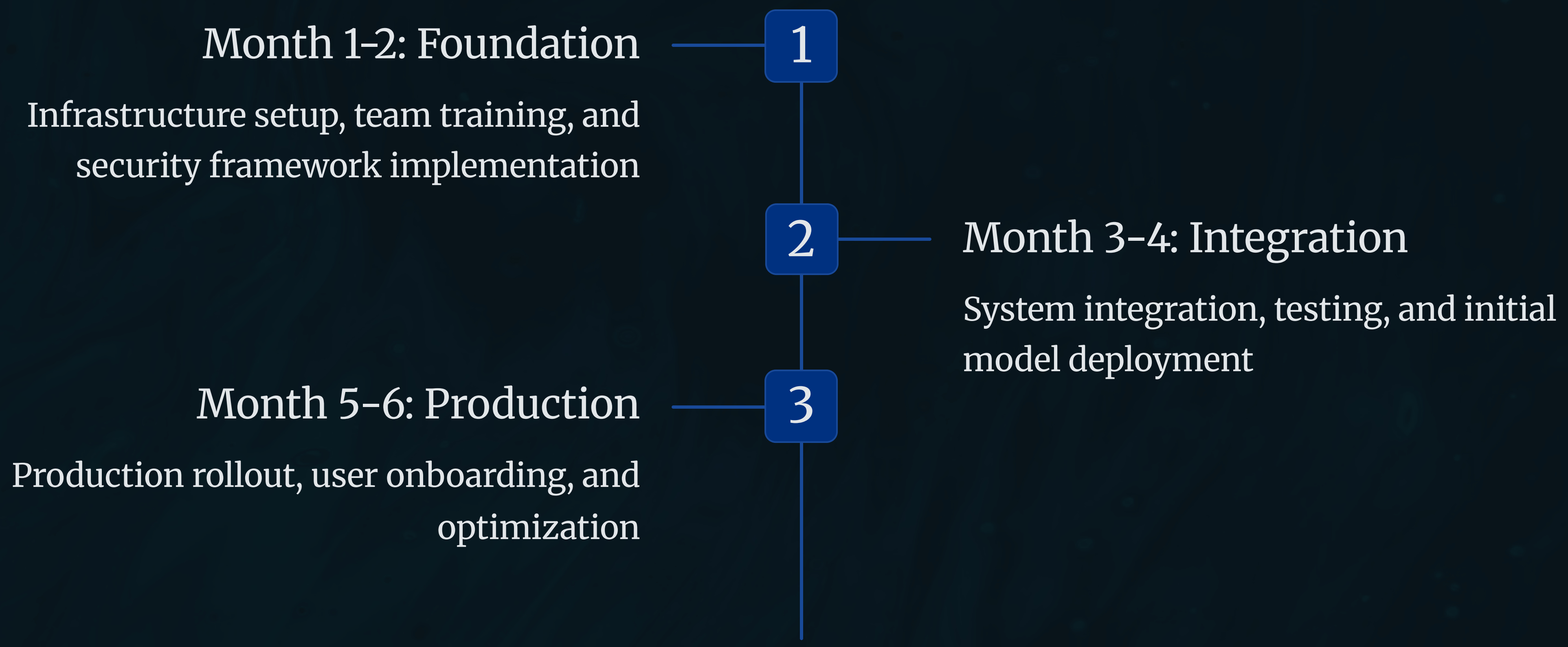
Technical Validation

- End-to-end testing completion
- Performance benchmark validation
- Security penetration testing
- Disaster recovery drill execution
- Compliance audit readiness

Operational Excellence

- 24/7 monitoring activation
- Incident response procedures
- User training and documentation
- Support escalation processes

The adoption process typically spans 3-6 months depending on deployment scale and organisational complexity. Success factors include executive sponsorship, dedicated technical resources, and alignment with existing IT governance frameworks.



1 Month 1-2: Foundation

Infrastructure setup, team training, and security framework implementation

2 Month 3-4: Integration

System integration, testing, and initial model deployment

3 Month 5-6: Production

Production rollout, user onboarding, and optimization

Critical Success Factor: Ensure adequate GPU resources are available before beginning model deployment. Supply chain constraints may impact timeline planning.

Technical Glossary & Key Concepts

RAG (Retrieval-Augmented Generation)

A technique that enhances large language models by retrieving relevant information from external knowledge bases before generating responses, improving accuracy and reducing hallucinations.

KV Cache (Key-Value Cache)

A memory optimization technique that stores previously computed key-value pairs during inference, significantly reducing computational overhead for sequential token generation.

LoRA/PEFT

Low-Rank Adaptation and Parameter-Efficient Fine-Tuning methods that enable model customisation with minimal computational resources and storage requirements.

SLO (Service Level Objective)

Quantitative measures of service performance that define acceptable levels of availability, latency, and throughput for system operations.

Unit Economics

The direct costs and revenues associated with processing individual requests or tokens, enabling precise cost prediction and resource allocation.

Speculative Decoding

An inference acceleration technique that generates multiple token candidates simultaneously, reducing latency whilst maintaining output quality.

Canary Deployment

A progressive deployment strategy that gradually rolls out updates to a subset of users, enabling early detection of issues before full production deployment.

Eval (Model Evaluation)

Systematic assessment of model performance using standardised benchmarks and domain-specific metrics to ensure quality and safety standards.

Understanding these technical concepts is essential for successful implementation and operation of sovereign AI systems. Each term represents critical knowledge areas that technical teams must master to ensure optimal system performance and reliability.

> "Sovereign AI represents not just technological independence, but the foundation for trustworthy artificial intelligence that serves organisational goals whilst respecting regulatory requirements and ethical boundaries."