Google Cloud

Sovereign AI on Your Terms

Build and scale AI without compromising control.



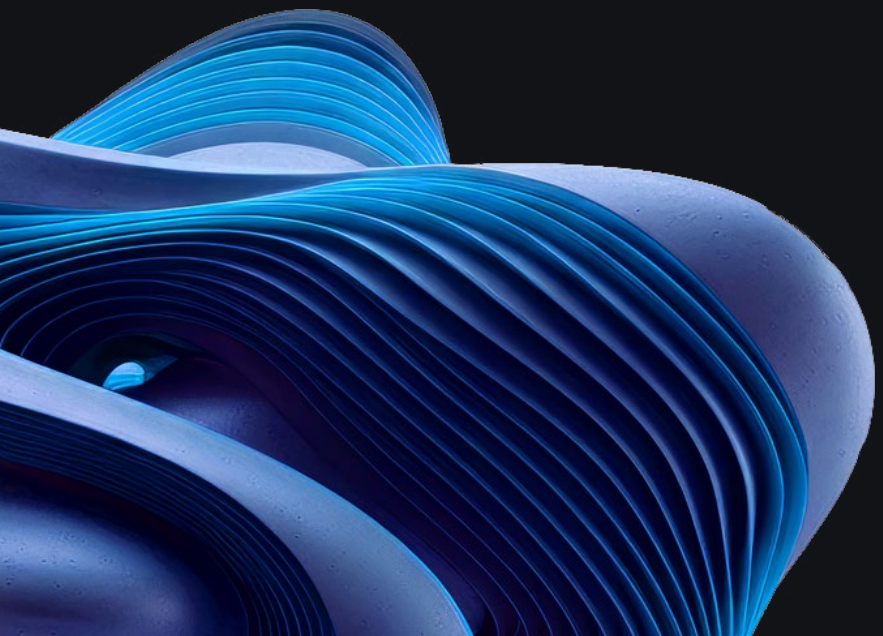
Executive summary

An AI stack is only as sovereign as its weakest layer.

As AI moves into core business operations, leaders need direct control to protect critical services, sensitive data, and intellectual property from sudden geopolitical shifts, export rules, or supplier disruption.

The market offers sovereign chips, sovereign clouds, and sovereign models. Each claim is true, and each covers only one layer. Yet an AI stack is only as sovereign as its weakest layer, the one teams can't control or replace.

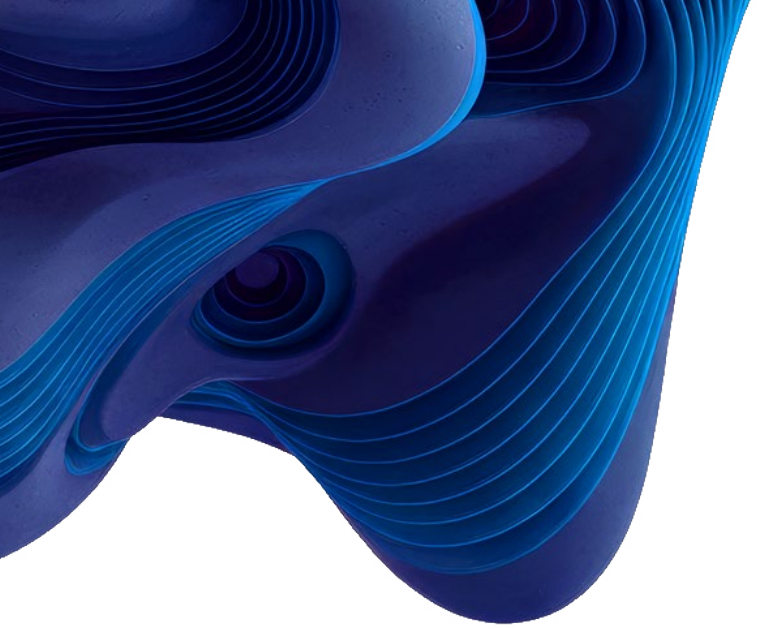
This paper sets out a practical architecture across infrastructure, data, models, and applications and agents, showing what those controls deliver when tuned to workload risk. By keeping the AI stack verifiable at every layer and portable across shifting laws and vendors, organizations build with local data and talent. Sovereign AI, on your terms.



Matt Garr
Group Product Manager
Sovereign AI, Google Cloud

Senanu Aggor
Senior Product Marketing Manager
Sovereign AI, Google Cloud

Michelle Zuanich
AI Regulatory Strategy
and Advocacy Lead, Google Cloud



Contents

01 The sovereign AI stack

02 Platforms for sovereign AI

03 Applications and agents

04 Model choice
Proprietary and open weights

05 Sovereign AI cost, alignment, and portability

06 Conclusion
Sovereignty as an operational mandate



Overview

Building sovereign AI on a foundation of security and choice

Moving AI into core enterprise and public-sector workflows brings sovereign control and cyber resilience together. Organizations want clear authority over where sensitive data lives, who holds the encryption keys, and which jurisdictions govern operations. At the same time, they need confidence that the underlying platform can defend their workloads and stay resilient through external disruptions, including cyberattacks, physical damage to data centers, and rogue agent actions across connected systems.

Keeping data inside a national border offers little protection if an unscreened agent call leaks sensitive records at runtime, and strong security controls fall short if shifting laws or supplier terms interrupt a critical service.

Attackers now use AI to automate reconnaissance and rewrite malware on the fly, shrinking the window between vulnerability disclosure and active exploitation from weeks to days—just as enterprises connect autonomous agents to live databases and external APIs. Rather than relying on static security perimeters alone, leaders look for platform-wide cyber resilience alongside verifiable sovereign outcomes before scaling AI.

Every deployment starts from a hardened security and cyber resilience foundation. This paper focuses on the sovereignty architecture built on top of it, and how to put sovereign controls into practice. Strict custody, continuity, and local value goals do not require walled-off workloads or closed systems.

Sovereign AI from Google Cloud delivers custody, continuity, and local value through a composable stack across infrastructure, data, models, and applications and agents. Built on open standards, our composable stack lets teams set custody per workload, choose freely among vendors, and evolve with confidence as models, threats, and regulations change.



The sovereign AI stack

A stack is only as sovereign as its weakest layer.

Running AI in production exposes organizations to operational risks well beyond the data center. Export rules can cut off model access, vendors can change licensing, regulators may demand audit trails and exit plans, and agents can act on live data.

Where single-layer and closed-stack offers leave teams exposed

Compute vendor

- Agents and apps
- Models
- Data
- **Infrastructure**

Covers local GPUs, leaving data, models, and apps exposed.

Cloud provider

- Agents and apps
- Models
- **Data**
- **Infrastructure**

Covers cloud compute and data, leaving models and agents exposed.

AI lab

- Agents and apps
- **Models**
- Data
- Infrastructure

Covers the model weights alone, leaving customer data, agents, and cloud infrastructure exposed.

One vendor, no exit

- **Agents and apps**
- **Models**
- **Data**
- **Infrastructure**

Covers every layer, yet ties teams to one vendor if policy shifts.

Single-layer approaches leave real gaps across other parts of the stack, while closed full-stack platforms restrict long-term portability and exit options. At the same time, physically air-gapping every workload adds steep infrastructure cost, manual upkeep, and slower adoption across engineering teams.

Without making deliberate architectural choices across the whole stack, engineering teams spend cycles reacting to compliance rules instead of shipping products. Designing infrastructure, data, models, and applications and agents together turns sovereign mandates into verifiable designs and technical controls. Engineering teams can train, fine-tune, and run models in production knowing that external policy shifts won't break live deployments.

Maintaining operational control does not mean giving up advanced AI capability. A modular stack lets teams run proprietary, third-party, or open-weight models across public cloud boundaries, partner-operated clouds, or local hardware. Each tier carries distinct cost and operational trade-offs, and sovereignty comes down to how organizations configure controls across the stack.

Choices and controls across the sovereign AI stack

Applications and agents

Agents query databases, call external tools, and take actions across live systems on a user's behalf. Building on open protocols like Model Context Protocol (MCP) and Agent-to-Agent (A2A) lets teams move agent code across environments without rewrites, while enforcing scoped user permissions and writing every action to tamper-evident logs.

Models

Standardizing model APIs across environments lets teams pair Gemini and third-party frontier models with open-weight models they can inspect and host locally. As workload sensitivity changes, teams switch models through configuration rather than code.

Data

Residency rules, access policies, and encryption keep sensitive data under direct custody. External key management, hardware security modules, and confidential computing can block third-party access to data at rest, in transit, and in use.

Infrastructure

Physical hardware location, operator jurisdiction, and offline survivability set the operational baseline. Teams choose per workload between a public cloud boundary, an independent local partner cloud, or local hardware.

Calibrate control across the stack for each workload

Pick the right balance of capability and control at every layer. Open standards and common APIs let workloads move into stricter isolation using the same codebase.

	Highest capability	Highest control
Apps and agents	Managed cloud agents	Create and deploy custom agents
Models	Proprietary frontier models	Local Gemini and open-weight
Data	Residency and customer-managed encryption keys	Externally held keys / in-rack security
Infrastructure	Public data boundary	Partner-operated or air-gapped

Turning stack controls into real outcomes

Because every production AI workload spans hardware, data, models, and application code, no single layer delivers sovereignty on its own. Isolating physical infrastructure doesn't protect sensitive records if an autonomous agent leaks local context through an unscreened tool call at runtime.

For example, keeping a critical service running when external links go down takes coordinated controls across the stack:

- Portable containers and open agent protocols in the app runtime
- Consistent model APIs to route between cloud and local weights
- Local or partner-run hardware with an independent execution path

Securing sensitive or regulated data requires a layered approach, pairing scoped runtime identities and prompt screening at the agent layer with customer-held encryption keys at the data layer.

Tuning infrastructure, data, model, and agent controls together lets engineering and security teams match technical controls to the legal rules, threat model, and commercial priorities of each individual workload.

That turns sovereignty from a static paper compliance exercise into concrete operational outcomes teams can verify in production.



What a sovereign AI stack delivers

Interoperability and survivability

Critical workloads keep running through network outages, geopolitical disruptions, or supplier changes. Open runtimes and tested failover let teams shift traffic locally or move workloads to another provider on the same codebase.

Security and compliance by design

Access policies, encryption keys, and residency rules sit directly in the runtime request path for models, data, and agents. Extending those controls to autonomous tool calls blocks prompt injection and unauthorized data movement before an action executes. Because the platform logs every identity check, key operation, and tool invocation automatically, security teams can hand regulators continuous runtime and cryptographic proof of compliance instead of static attestations.

Economic competitiveness and independence

Meeting public sector and industry baselines opens regulated markets while building local engineering talent. Keeping proprietary datasets, fine-tuned weights, and domain expertise inside domestic jurisdictions protects commercial IP and avoids dependence on any single outside vendor.

Cultural alignment

Models, applications, and agents reflect local languages, legal frameworks, and institutional norms rather than generic defaults. Teams ground outputs in domestic data sources, tune weights on domain corpora, and set safety and approval rules tailored to local operations.

Few workloads need maximum isolation across every outcome at once. Sovereignty works as a set of independent dials across the stack, built on a secure public cloud baseline. The architect's job is to dial up control only at the layers where regulatory rules or operational risks require it.

For example, a public agency might run frontier models inside a public cloud data boundary, while a defense system runs open-weight models on air-gapped hardware.

Calibrating each layer per workload avoids over-building. Teams use frontier models in a public cloud boundary for general productivity, add external keys and confidential computing for regulated data, and reserve disconnected hardware for classified operations.

The architectural layers stay consistent across the entire portfolio, while the underlying deployment tier shifts to match the real risk of each workload.

Platforms for sovereign AI

Sovereign AI relies on portable architectures that avoid lock-in.

Most enterprises run workloads across public cloud data boundaries, partner-operated clouds where national law dictates local operation, and on-premises hardware where critical workloads need offline survivability.

Because Google Cloud runs the same Kubernetes, model serving, and agent APIs across every tier, teams write application logic once and deploy it wherever regulations require. Workloads can call Gemini, partner, and open-weight Gemma models in the public cloud today, and move onto partner or local hardware tomorrow without rewrites.

Google Sovereign Cloud spans all of these environments so engineering teams can place each workload where its data and legal rules belong.

One codebase across every tier

Google Cloud Data Boundary

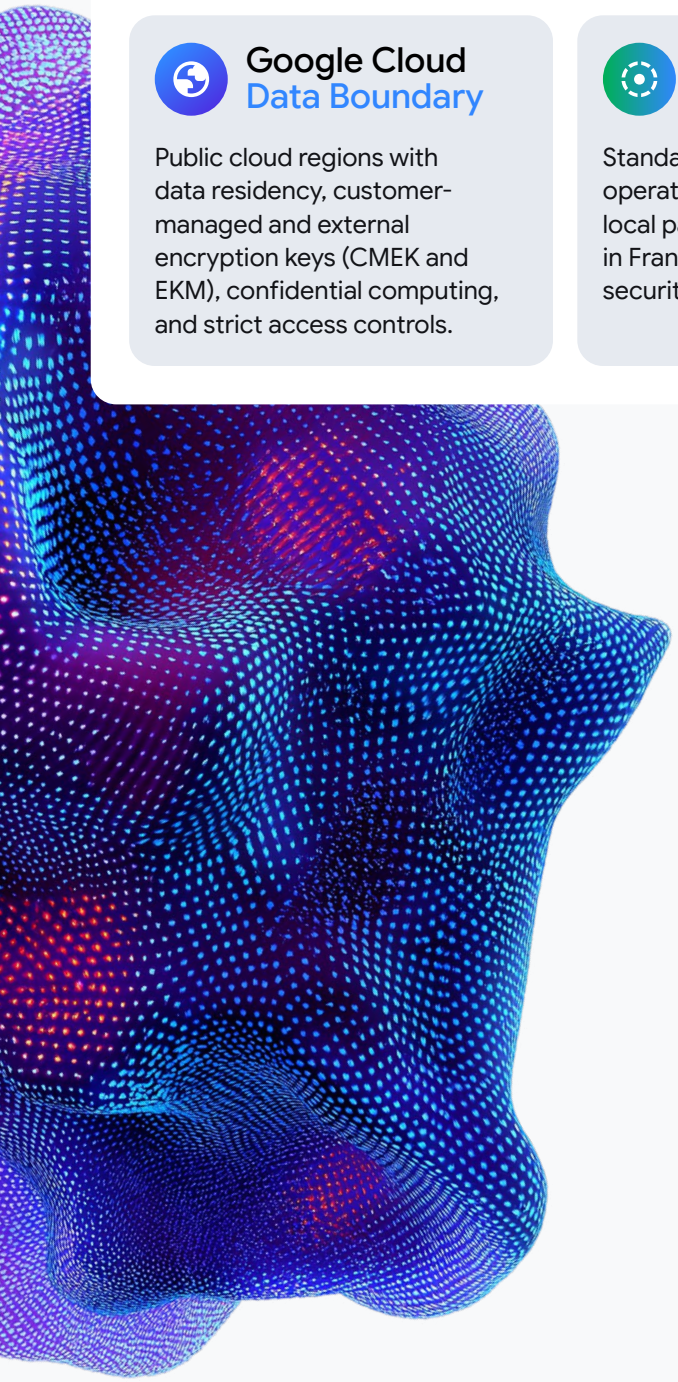
Public cloud regions with data residency, customer-managed and external encryption keys (CMEK and EKM), confidential computing, and strict access controls.

Google Cloud Dedicated

Standalone cloud universes operated by independent local partners (such as S3NS in France) with local hardware security modules (HSMs).

Google Distributed Cloud

Connected or air-gapped hardware inside customer data centers for low latency and high survivability.



Case study • United States

Defense Logistics Agency

Frontier AI for supply chains inside a DoD IL5 / FedRAMP High boundary.

The challenge

Strict IL5 and FedRAMP High rules have historically pushed agencies onto isolated government clouds that trail commercial AI, even as logistics teams need modern models on sensitive supply chain data.

The approach

Run Gailforce inside a Google Cloud Data Boundary, accessing Gemini for Government on DoD's GenAI.mil platform for public-cloud scale under strict IL5 controls.

~60% to 85%
forecast accuracy target

55
AI/ML models

IL5 + High
kept on public cloud

Multi-region
US failover

Control layer	What DLA chose	What it delivered
Apps and agents	Gailforce Sitrep and assistants on open standards, portable to the edge. Every action logged.	→ **Cultural alignment** Grounded in internal data classification and handling norms.
Models	Gemini for Government plus 55 mission-specific AI/ML models, routed dynamically inside IL5.	→ **Economic benefit** Forecast accuracy toward 85% cuts waste and frees working capital.
Data	Supply chain data stays in the boundary, with US-held keys and automated least-privilege logs.	→ **Compliance by design** Continuous evidence that data stays in the perimeter preserves the ATO.
Infrastructure	Software-defined controls on shared public cloud, operated at scale by US personnel.	→ **Survivability** Fails over across multiple US regions during outages and surges.

Google Cloud Data Boundary



Applications and agents

Since autonomous agents call external tools and modify live systems, the security model must govern tool execution paths at runtime.

Once an AI agent can query databases, call external APIs, execute code, or spin up sub-agents, it acts on a user's behalf across organizational boundaries. Giving models direct tool access inside regulated environments introduces operational security, auditability, and compliance risks that data residency rules may fail to catch.

Tool egress and boundary integrity

An agent running inside a sovereign perimeter can still leak sensitive records if it passes local context into an external search connector, webhook, or third-party API. Keeping that data boundary intact means restricting model-selected tool calls strictly to approved endpoints and network paths.

Multi-agent delegation and identity

When an orchestrator delegates subtasks across specialist sub-agents, broad backend service accounts create paths for privilege escalation. Rather than relying on static backend keys, each sub-agent should inherit short-lived, downscoped credentials tied directly to the originating human user.

Indirect prompt injection

Agents routinely read untrusted text like incoming emails, tickets, and shared documents. Hidden instructions inside that content can hijack which tools the agent calls next. Because the agent holds active credentials, every tool parameter should be screened before execution.

Memory lifecycle and data retention

To carry context across sessions, agents store conversation history and vector embeddings in persistent memory. Under rules like GDPR and the EU AI Act, teams must set explicit retention limits on contextual memory and demonstrate that erased records cannot be recovered.

Runtime controls for sovereign agents

Governing agentic workflows means enforcing security checks directly inside the agent execution runtime rather than relying on perimeter controls alone.

1 Scoped, non-human IAM identities

Agents execute under short-lived IAM identities linked to the originating user. Whenever an agent invokes a sub-agent, IAM automatically downscopes the token so downstream tasks never exceed the permissions of the initial request.

2 In-flight gateway inspection

Tool calls built on open standards like MCP route through Agent Gateway inside a VPC Service Controls perimeter. Model Armor inspects prompts, completions, and tool arguments in flight to block prompt injection and sensitive data leaks before execution.

3 Customer-managed encryption

Agent memory, vector indexes, and intermediate session states can be encrypted using customer-managed encryption keys (CMEK) or external keys. Revoking key access immediately blocks further access to stored context and agent state.

4 On-behalf-of audit trails

Agent tool calls, delegations, and reasoning steps can write to Cloud Audit Logs with on-behalf-of user attribution, giving security teams clear visibility into live actions.

Sovereign outcome	Agent layer delivery	Implementation in platform
Security and compliance by design	Scoped IAM credentials and runtime tool screening prevent unintended data movement and privilege expansion across multi-agent workflows.	Model Armor and Agent Gateway enforce policy checks within the VPC Service Controls perimeter before tool calls execute.
Interoperability and survivability	Standardized tool protocols and agent discovery formats keep applications portable across deployment environments with one codebase.	Open standards like MCP and A2A let agents run unchanged across public cloud, partner-operated Google Cloud Dedicated, and Google Distributed Cloud racks.
Economic competitiveness and independence	Shared tool interfaces decouple business logic from underlying model weights so teams aren't tied to a single agent framework.	Unified runtime APIs let teams build an agent workflow once and route calls dynamically across Gemini, partner, and local open-weight Gemma models.
Cultural alignment	Explicit tool allowlists and human-in-the-loop approvals keep automated agents operating safely within local policy and risk limits.	Teams can configure high-impact actions like financial transfers, database writes, or access changes to require explicit human sign-off before execution.

Evolving regulatory frameworks

Regulators across major jurisdictions are building these and similar runtime controls into AI governance frameworks.

EU AI Act

Mandates automatic runtime event logging, built-in human oversight controls, and clear transparency disclosures for user-facing AI system interactions.

US NIST AI Standards

Focuses on authenticated non-human identities, strict credential scoping, and end-to-end provenance tracking for autonomous agents.

Singapore IMDA

The Model AI Governance Framework for Agentic AI sets practical baselines for scoped API keys, tool-call observability, and runtime risk controls on autonomous agents.

Building identity downscoping, in-flight gateway screening, key-backed memory, and on-behalf-of audit logs directly into the agent runtime lets teams deploy autonomous workflows without losing control of regulated data or compliance posture.



Model choice

Proprietary and open weights

Run AI workloads at every isolation tier without giving up model quality or vendor independence.

Google Cloud supports a broad ecosystem of model choices, spanning frontier Gemini models, third-party frontier models, and open-weight Gemma and partner models that teams can inspect, tune, and host. Developers build on one set of APIs and tooling across every tier.

Open-weight control with Gemma 4

Open-weight models are foundational to a survivable AI stack. Under the Apache 2.0 license, Gemma 4 grants organizations full autonomy over model weights, fine-tuning, and local execution.

- Fully inspectable and auditable model weights
- No external telemetry
- Independent from third-party cloud APIs

Gemma 4 is built for constrained and disconnected environments.

Architectural efficiency

Gemma 4 spans 2 billion to 31 billion parameters across dense and mixture-of-experts (MoE) architectures. The 26-billion MoE variant activates only 3.8 billion parameters per token, delivering 31-billion-class reasoning at a fraction of the compute cost. Compact variants like Gemma 4 E2B fit inside a sub-3 GB memory footprint for disconnected edge devices and local workstations.

Air-gapped multimodality

Gemma 4 handles text, images, video, and audio locally on enterprise hardware. Teams can run document extraction, visual inspection, and speech workflows offline with no outbound API calls.

Agentic control

With up to a 256K context window, native function calling, and configurable reasoning budgets, Gemma 4 runs multi-step autonomous agent workflows inside a cloud boundary, sovereign partner region, or local rack.

Proprietary, partner, and open models

Pairing Gemini with Gemma, third-party, and partner models avoids capability drop-offs as isolation rises. Teams route each workload to the right model mix.

Broad model choice on public infrastructure

Run Gemini, third-party, and Gemma models inside a cloud boundary, including Gemini for Government at FedRAMP High and DoD IL5.

Model choice through local partners

Partner-operated clouds, such as S3NS in France under SecNumCloud qualification, provide local operational control. Organizations can deploy open-weight Gemma and partner models through Model Garden in this perimeter today, while adopting managed Gemini models as they launch.

Air-gapped and local deployment

On Google Distributed Cloud, teams can run open-weight Gemma models, partner open-weight models like Mistral, or Google-signed Gemini Flash models directly on customer-operated hardware inside local data centers or edge sites.

Case study • European Union

Thales

Hybrid sovereign AI across on-premises hardware and S3NS, under SecNumCloud.

The challenge

Engineering AI must meet France's SecNumCloud and evolving European digital rules, while weighing fixed on-premises capex against cloud elasticity, and local open models against frontier reasoning.

The approach

Combine existing local hardware with PREMI3NS from S3NS, an independent French partner cloud on Google Cloud Dedicated.

SecNumCloud 3.2
S3NS PREMI3NS, qualified Dec 2025

Gemma + Mistral
open-weight models today

Hybrid
on-prem + S3NS

Open standards
Consistent APIs, no rewrites

Control layer	What Thales chose	What it delivered
Apps and agents	Thales' engineering builds on open standards and one common API so teams can switch models without rewrites.	→ **Cultural alignment** Grounds models in French engineering and legal terms.
Models	Runs Gemma and Mistral locally; advanced tasks route to Gemini, once SecNumCloud-qualified.	→ **Economic benefit** Baseline work on existing hardware; elastic cloud for complex reasoning.
Data	Sensitive data runs on S3NS, under French control. Secret data stays on-premises.	→ **Compliance by design** Audits every model run and data access.
Infrastructure	Keeps local hardware for some workloads; adds elastic capacity via S3NS on Google Cloud Dedicated.	→ **Survivability** S3NS is designed to be operated by the partner for up to 12 months if its connection to Google is severed.

Google Cloud Dedicated (S3NS)



Sovereign AI cost, alignment, and portability

Sovereignty comes with a cost curve. Defaulting to maximum isolation for every workload adds hardware and operating overhead that stalls AI roadmaps.

Under-protecting regulated data creates real legal exposure, while over-isolating everyday workloads drives up infrastructure and operating costs. Matching each workload to the right deployment tier on open standards keeps compliance spending predictable.

Understand the real cost drivers of isolation

Stronger isolation requires physical, operational, and supply-chain investment because dedicated hardware and local staffing cannot pool scale like a global public cloud. Moving workloads into higher-assurance tiers adds direct overhead across hardware, staffing, and key custody.

Dedicated infrastructure

Dedicated partner clouds, on-premises racks, and disconnected edge sites require isolated accelerators, power, and networking. Because capacity isn't shared across tenants, organizations fund dedicated headroom and longer hardware refresh cycles.

Sovereign operations

Operating high-assurance environments requires vetted personnel with local citizenship and active security clearances, backed by dedicated in-country support and SRE rotations independent of global engineering pools.

Supply chain and cryptographic autonomy

Maintaining verifiable custody adds ongoing hardware supply chain audits, local HSM operations, external key infrastructure, and signed root-of-trust checks.

Because dedicated silicon and cleared local operations carry fixed overhead, treating every application like a classified system drains AI budgets fast. Grouping workloads into clear assurance tiers reserves sovereign partner clouds and disconnected racks for the sensitive data that legally requires them, while keeping standard enterprise workflows inside lower-cost cloud boundaries so teams ship faster.

Protect engineering work with open APIs and portability

Model capabilities and national regulations change faster than enterprise hardware cycles. Tying AI applications to single-environment APIs creates vendor dependency and forces costly code rewrites whenever rules tighten.

Decoupling application logic from infrastructure safeguards software investments. Containerizing workloads on Kubernetes and using open agent protocols (MCP and A2A) keeps application code portable. The same container image and tool definitions run unchanged across public cloud regions, partner-operated clouds, and air-gapped racks.

At the model layer, a unified inference gateway abstracts the model endpoint from the calling application. Engineering teams build an agent once and switch model backends through configuration rather than code.

When a workload moves into a disconnected enclave or partner facility, the router shifts calls from cloud endpoints to local Gemini models or Gemma and Mistral weights served on vLLM, while preserving the same tool-calling and audit schema.

Standardizing on open protocols and portable runtimes preserves clear exit strategies. Engineering teams manage fewer codebases while compliance teams can relocate workloads cleanly as rules change.

Case study (field demonstration) • United States

U.S. Air Force and GDIT

Demonstrating air-gapped tactical AI at Exercise Mobility Guardian 2025.

The challenge

Tactical military units frequently operate in denied, degraded, intermittent, and limited-bandwidth (DDIL) environments without core cloud links, yet still need real-time AI and collaboration up to the Secret level.

The approach

At Exercise Mobility Guardian 2025 in Guam, GDIT and Google Public Sector demonstrated the ruggedized GDC air-gapped appliance at the tactical edge.

Up to Secret
Zero Trust security

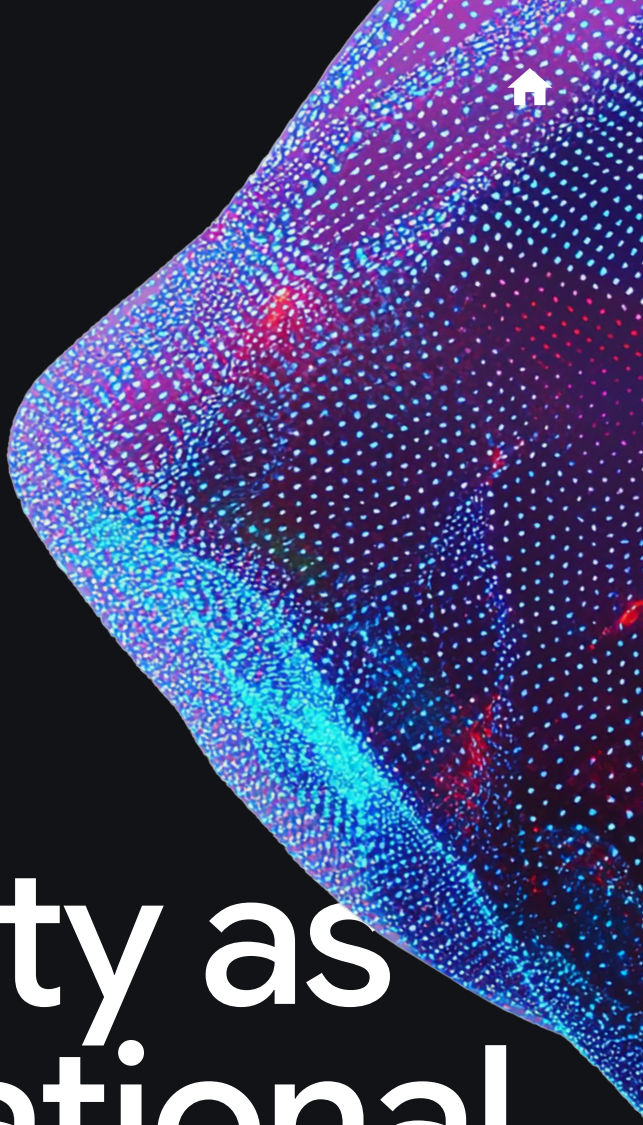
Fully offline
DDIL tactical edge demo

Hours
to integrate mission apps

Local AI
OCR, translation, and Luna AI

Control layer	What was demonstrated	What it delivered
Apps and agents	Containerized Mattermost collaboration, GDIT's Luna AI for air defense, and local Jupyter notebooks for field updates.	→ **Cultural alignment** Grounded tactical troubleshooting in ingested Air Force technical manuals.
Models	Google Cloud AI and computer vision models run locally for translation, transcription, and OCR.	→ **Economic benefit** Brought commercial cloud and AI tools into live tactical workflows in hours.
Data	Technical manuals, air defense data, and logs processed locally up to the Secret level.	→ **Compliance by design** Kept sensitive exercise data on the local appliance with no outbound network traffic.
Infrastructure	Ruggedized, transportable GDC air-gapped "cloud-in-a-box" appliance in Guam.	→ **Survivability** Operated autonomously without internet or cloud links, syncing when connectivity returned.

Google Distributed Cloud • Demonstrated live at Exercise Mobility Guardian 2025.



Conclusion

Sovereignty as an operational mandate

Because enterprise AI agents now query data and act on live systems directly, sovereignty requires deliberate engineering across every layer.

Most enterprises still govern AI sovereignty one project at a time, slowing engineering teams whenever regulations or threats evolve.

Moving faster starts with tiering workloads by real data and operational risk instead of applying one blanket isolation rule.

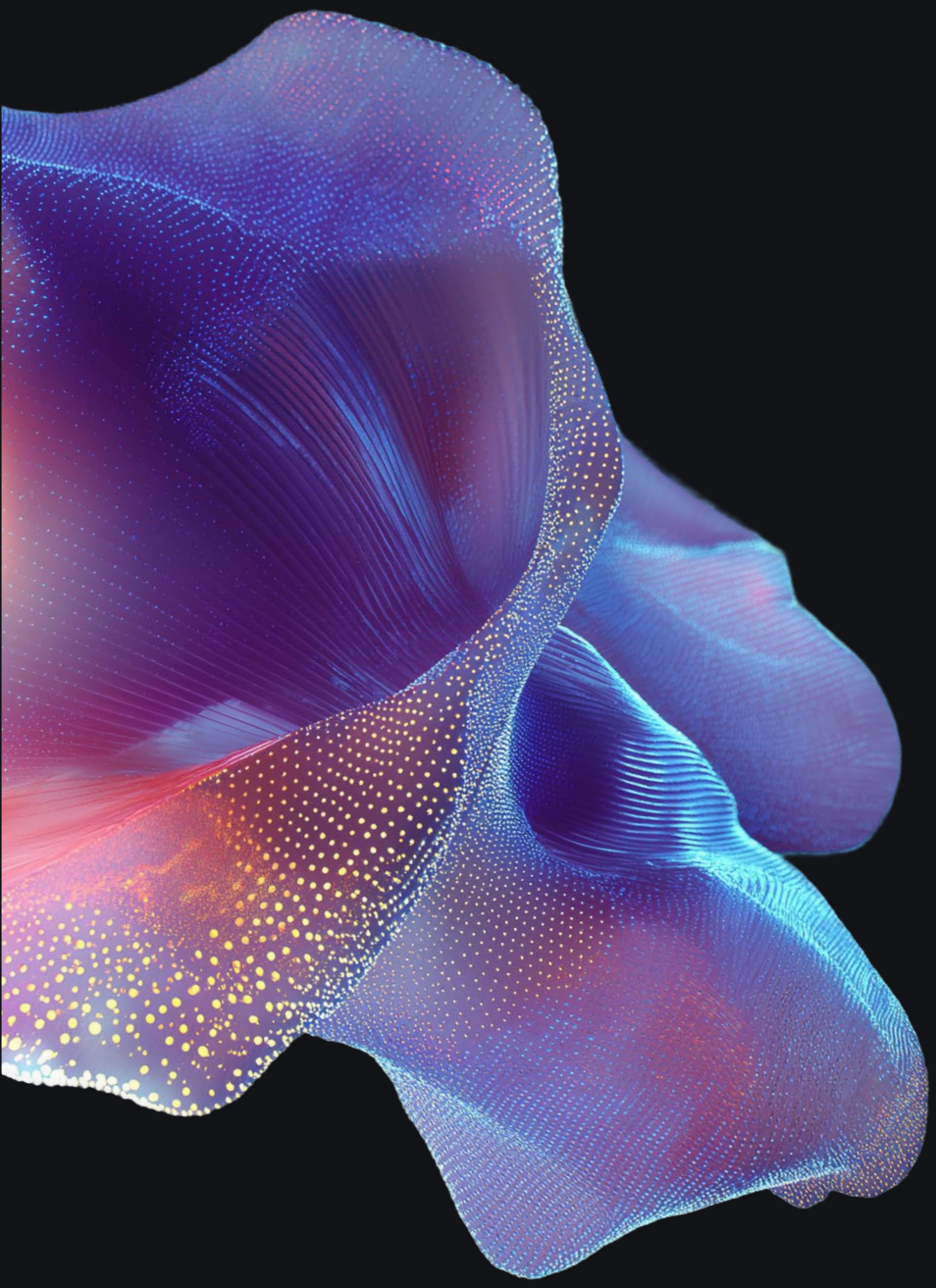
Keep everyday productivity and analytics inside a public cloud data boundary with customer-held keys, reserving partner clouds and air-gapped hardware for regulated workloads.

Build new applications on Kubernetes, open agent protocols like MCP and A2A, and consistent APIs from day one so workloads can move across deployment tiers without code rewrites.

Run Gemini across sovereign boundaries and local deployments, complemented by open-weight Gemma and partner models where workloads need direct weight custody. As agents act on live systems, enforce scoped user identities, in-flight prompt screening, and on-behalf-of audit logs on every tool call.

Put together, those practices turn sovereignty into a practical operating advantage. Security and compliance teams gain continuous runtime audit evidence for regulators, while engineering teams ship new capabilities faster and keep clear exit options open.

Delivering modern AI capability does not require giving up verifiable control. With an open, modular stack verified at every layer, you can build and scale AI across core operations on your own terms.



Google Cloud

Build your verifiable sovereign AI architecture

Learn more

