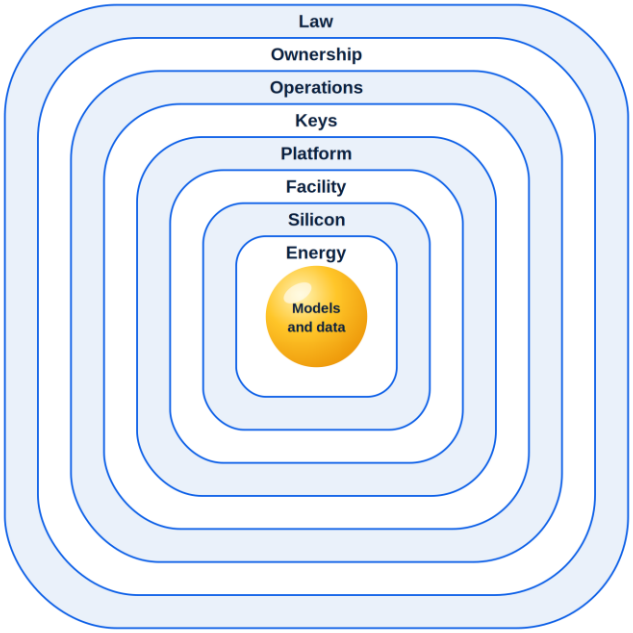
Sovereign AI in the Data Center

Who controls the machines that think, and how to build so that the answer is you.

A field guide for operators, policymakers and buyers.

Published by Anomny. First edition, September 2026.



Contents

Preface 3

Part I: The idea 4

1. What “sovereign” actually means 5
2. The Sovereignty Stack 7
3. Why now 10

Part II: The physical layer 12

4. Anatomy of an AI data center 13
5. Power, cooling and siting 15
6. Silicon and the supply chain 17

Part III: The control layer 19

7. Software, keys and operations 20
8. Data and models 22
9. Security and resilience 24

Part IV: The landscape 26

10. Four regional approaches 27
11. The paradox of dependence 30

Part V: Practice 32

12. The economics of sovereignty 33
13. A build playbook 36
14. Governance and compliance 39
15. Looking ahead 41

Appendices 43

A. Sovereignty scorecard 44
B. Glossary 47
C. Sources and further reading 49
D. About this edition 51

Preface

Why a book about buildings full of chips is really a book about power.

Three years ago, “sovereign AI” was a phrase you met in policy white papers. Today it appears in procurement documents, budget speeches, bank risk registers and the marketing of nearly every infrastructure vendor. That speed of adoption has a cost: the phrase now means so many things that it can be used to justify almost any project, and to sell almost any product.

This book tries to make the phrase precise. Its central claim is simple. **Sovereignty over AI is decided in the data center**, or more exactly in the layers that meet there: the law that governs the operator, the people who hold administrative access, the keys that protect the data, the software that schedules the work, the chips that do it, and the electricity that keeps them running. If someone else controls one of those layers, they hold a lever over everything above it.

The book is written for three kinds of reader. If you *operate* infrastructure, you will find a vocabulary for talking to your regulators and customers, and a set of design principles. If you *make policy*, you will find a way to tell a serious programme from a press release. If you *buy* AI capacity for a bank, a hospital, a ministry or a company, you will find questions to put to every vendor who promises to be sovereign.

How to read it

Part I builds the concepts, including a framework called the Sovereignty Stack that the rest of the book uses. Part II goes into the physical layer: what an AI data center is, why power and cooling dominate its design, and why silicon supply is the hardest dependency to remove. Part III covers the control layer: software, keys, operations, data, models and security. Part IV surveys how Europe, India, the Gulf and others are actually building, and confronts the central paradox of the field. Part V is practical: economics, a build playbook, governance, and a look ahead. The appendices contain a scorecard you can use on your own project, a glossary, and the sources.

You can read straight through, or jump. Chapters 2, 11 and 13 are the ones to read if you only read three.

> **A note on numbers**
>
> This edition reflects public information as of 20 September 2026. Figures for capacity, funding and GPU counts come from government statements, company announcements and press analysis, and they often disagree with one another. Where sources differ, the text says so. Treat every number as an order of magnitude and check it against a current source before you rely on it. This field changes monthly.

Part I: The idea

What sovereignty means, where it lives, and why so many governments and companies want it now.

1. What “sovereign” actually means

A model trained in your country, on your servers, can still be answerable to someone else.

Picture a health ministry that wants to train a diagnostic model on national patient records. It rents space in a data center inside its borders. The data never leaves the country. The GPUs are in a cage the ministry can visit. By the usual definition, that is “sovereign.”

Now look closer. The cluster’s control plane is run from a cloud region abroad. The operator is a subsidiary of a foreign company that can be served with a foreign court order. Firmware updates for the servers arrive from the manufacturer and install automatically. The keys that encrypt the records are held by the cloud provider. The accelerators were exported under a licence that can be revoked. Is the ministry in control?

The honest answer is: *in control of what, against whom?* Sovereignty is not a single property. It is a bundle of separate questions, and a system can pass some while failing others.

Four levers

The simplest way to test any AI infrastructure is to ask who holds four levers over it:

- **Access.** Who can read the data, the model weights, or the prompts and outputs?
- **Continuity.** Who can switch the service off, throttle it, or refuse to renew it?
- **Compulsion.** Who can legally force the operator to hand over data or change behaviour?
- **Change.** Who decides what software, hardware and policies are updated, and when?

An organisation is sovereign over its AI to the extent that all four levers rest with parties it can hold to account. If any lever sits with a foreign government, a foreign corporation, or an unaccountable vendor, sovereignty has a gap there, however many flags the marketing brochure carries.

Five words that are not synonyms

Term	What it answers	What it does not guarantee
Data residency	Where is the data stored and processed?	Who can access it, or which law applies to the operator.
Data sovereignty	Which country’s law governs the data, and who can compel access to it?	Who runs the machines, or whether the service keeps running in a crisis.
Operational sovereignty	Who administers the systems, with what clearance and under what oversight?	That the underlying technology is domestic, or that supply will continue.
Technological autonomy	Could you keep operating, and rebuild, if foreign inputs stopped?	That you can afford to do so, or that your capability is frontier-level.

Term	What it answers	What it does not guarantee
AI sovereignty	Can you develop, run and govern AI across compute, data and models under your own authority?	Anything, unless each layer beneath it holds.

Most projects marketed as sovereign today deliver the first row, some deliver the second and third, and very few deliver the fourth. That is not a criticism. It is a description of where the market is, and it explains why the definition matters.

A ladder, not a switch

Because sovereignty is a bundle, it is more useful to think in rungs. This book uses four. Each rung adds control and each costs more.

Resident

Data and compute sit in-country. The provider, its parent and its tooling are foreign.

Assured

Contracts and technical controls limit foreign access. The provider may still be foreign-owned.

Operated

A domestic entity owns and runs the service under domestic law, using licensed foreign technology.

Autonomous

Domestic control end to end, with diversified supply and the ability to run unassisted.

The ladder is deliberately uneven at the top. Almost no country can reach the fourth rung on every layer, and a later chapter explains why. The useful question is never “are we sovereign?” It is “on which rung are we, layer by layer, and is that enough for this workload?”

> **Sovereignty is a relationship, defined by two things: your ability to say no, and the cost of leaving.**

Sovereignty is not self-sufficiency

A common misreading equates sovereignty with building everything at home. No serious actor pursues that. Even the states with the deepest technology bases depend on international supply chains for chips, memory, tools and materials. What sovereignty asks for is *controlled dependence*: relying on others in ways that are visible, contractual, diversified and reversible.

That reframing changes the engineering. The goal stops being “remove all foreign parts” and becomes “make sure no single foreign party can hurt us, and know what it would cost to switch.”

In short. *Sovereign AI means holding the levers of access, continuity, compulsion and change over your compute, data and models. It comes in degrees, and it is judged layer by layer.*

2. The Sovereignty Stack

Eight layers where someone else can hold the lever, and the weakest one sets the limit.

A data center is not one thing. It is a stack of dependencies, from the law that governs the operator at the top to the power plant at the bottom. This chapter names eight layers. They are the rings on the cover, and every later chapter refers back to them.

1	**Law and jurisdiction**	Whose courts and statutes can compel access or shut the service down?
2	**Ownership and control**	Who owns the operator and the assets, and who can change management or sell them?
3	**Operations and people**	Who administers the systems, and who holds privileged access?
4	**Keys and identity**	Who holds the encryption keys and the identity system, and can they be revoked?
5	**Platform software**	Who controls the orchestration, drivers, schedulers and model-serving stack?
6	**Facility and network**	Who owns the building, the connectivity and the exchange points?
7	**Silicon and hardware**	Who designs, fabricates, ships, updates and licenses the chips and servers?
8	**Energy and materials**	Where do power, cooling water, fuel and spare parts come from?

Above the stack sit the things it exists to protect: **models and data**. Chapter 8 covers those.

Reading the stack

Law and jurisdiction

The top layer decides which government can force the operator's hand. Extraterritorial laws matter here: several countries claim the right to demand data from companies they govern, wherever that data is stored. Data physically in your country can still be reachable if the operator's parent is subject to such a law. Strength at this layer means the operating entity answers to domestic law and has no foreign parent that can be compelled to act against domestic interests.

Ownership and control

Who owns the operator shapes who can override decisions. Joint ventures, licensing arrangements and “golden share” structures all try to give the host country a veto without giving up foreign technology. The test is whether the domestic party can act alone if a dispute arises.

Operations and people

Privileged administrators can do almost anything. Sovereign operations means cleared, in-country staff; supervised, logged and time-limited vendor access; and a real ability to run the site without a call to a foreign help desk. Many projects fail here quietly, because the vendor’s support model assumes remote access.

Keys and identity

If the provider holds your encryption keys and runs your identity system, encryption protects you from outsiders but not from the provider. Strong designs keep keys in hardware modules the customer controls, and tie access to an identity system the customer runs.

Platform software

The scheduler, cluster manager, drivers and serving stack determine how easily you can move. Open standards and open-source components make switching possible. Proprietary layers that only work with one provider make it expensive.

Facility and network

The building, the power feed, the fibre routes and the internet exchange points are physical and local, which makes them the layer countries most often control already. Weakness appears in single-site designs, single-carrier dependence, and reliance on foreign subsea cables.

Silicon and hardware

Accelerators, memory, networking and servers come from a small number of suppliers concentrated in a few places. Firmware inside them can change behaviour. Export licences can restrict what you may buy and where you may run it. This is the hardest layer to make sovereign, and Chapter 6 gives it the space it needs.

Energy and materials

An AI data center is an electrical load first. Grid capacity, fuel imports, cooling water and spare-part supply all sit here. A site with the best legal structure in the world still goes dark when the substation does.

> **Sovereignty is only as strong as the weakest layer. A perfect key-management design does not help if the chips can be cut off.**

Why the stack is useful

It turns a vague debate into a checklist. Instead of arguing whether a project is “really sovereign,” you can score each layer, find the weakest, and decide whether it matters for the workload at hand. A hospital training on de-identified data may accept weakness at the silicon layer. A defence ministry may not. Appendix A gives you a scorecard for doing exactly this.

It also exposes a common trap. Vendors tend to be strong at one layer and silent on the rest. A confidential-computing product speaks to keys. A domestic colocation firm speaks to the facility. A national champion speaks to ownership. Each is real, and none is the whole stack.

In short. *Score every layer, find the weakest, and judge it against what the workload actually needs. That is the whole method.*

3. Why now

Six forces turned a niche concern into a global building programme.

Governments have worried about technological dependence for decades. What changed is that AI made the dependence concrete, expensive and fast-moving, and made it visible to people who had never thought about data centers before.

The six forces

1. Geopolitics and export controls

Advanced accelerators became instruments of foreign policy. Licences to export them can be granted, conditioned or withdrawn. For a country planning a multi-year build, that turns chip supply from a procurement question into a strategic risk. It is the single strongest driver behind the current wave.

2. Regulation and legal risk

Data-protection regimes, sector rules for finance and health, and new AI-specific laws all push toward knowing where data goes and who can touch it. Court rulings that limited transatlantic data transfers showed European organisations that legal exposure can appear overnight. Several large economies have since written localisation or residency expectations into law or procurement.

3. National security and public services

Governments are moving AI into tax, health, policing, defence and welfare. Systems that carry that weight cannot depend on a foreign vendor's goodwill. The reasoning is the same one that led states to build their own power grids and telecom backbones.

4. Economic value capture

Compute is becoming an input to growth, like electricity. Countries that only rent it pay a permanent toll and export the margin. Countries that host it capture construction, operations, energy demand and a base for local start-ups.

5. Language, culture and fit

Models trained mostly on a few languages perform unevenly elsewhere. Local models, trained or adapted on local data, serve local needs better, and that requires local compute under local control.

6. Corporate risk management

Enterprises in regulated sectors face the same concerns in miniature. A bank does not want its risk models on infrastructure it cannot audit. A pharmaceutical firm does not want trial data reachable by a competitor's government.

What the numbers say

The market has responded at speed. Gartner forecast worldwide spending on sovereign cloud infrastructure services at about $80 billion in 2026, up roughly 35 percent on 2025. A tracker of government AI projects run by the Center for a New American Security counted more infrastructure projects announced in the first quarter of 2026 than in all of 2024. Infrastructure makes up the majority of what it tracks, about 59 percent, with model projects at 32 percent and data projects at only 9 percent.

That last split is worth pausing on. Governments are spending far more on buildings and chips than on the datasets and talent that make them useful. The same tracker concludes that rhetoric has run ahead of delivery, and that building national data centers does not by itself remove dependence on American technology. It moves the exposure from one layer of the stack to another. Chapter 11 returns to that point.

> **Sovereign-washing**
>
> Where money flows, labels follow. Be alert to projects that call themselves sovereign because they are located in-country, or because a national flag appears in the partnership announcement. Ask which of the eight layers is actually under domestic control, and how that has been verified. The scorecard in Appendix A is designed for this conversation.

In short. *Export controls, regulation, security, economics, language and risk management all point the same way. Spending is racing ahead of clarity about what is being bought.*

Part II: The physical layer

What an AI data center is made of, what it consumes, and why chips are the hardest thing to make sovereign.

4. Anatomy of an AI data center

A conventional data center stores and serves. An AI data center computes, densely, and around the clock.

To reason about sovereignty you need a working picture of the machine. An AI data center looks like a traditional one from the road and behaves very differently inside. The difference is density: the same floor area now carries several times the electrical load, and almost all of it turns into heat that must be removed.

The building blocks

Compute

The core is a large number of accelerators, chips built for the matrix arithmetic behind neural networks. They sit in servers of typically eight accelerators each, and increasingly in rack-scale systems where dozens of accelerators are wired together as one machine. A training cluster may combine thousands to hundreds of thousands of them.

Networking

Networking is as important as the chips. Inside a server or rack, accelerators talk over very high-bandwidth links (the “scale-up” fabric). Between racks, a second fabric (the “scale-out” network) carries the traffic that lets thousands of chips train one model together. A slow or unreliable network wastes expensive silicon, so training clusters are designed around it. Networking vendors are a dependency of their own.

Storage

Training reads enormous datasets and writes checkpoints, saved snapshots of the model that allow recovery from failure. That needs fast parallel storage. Inference needs less raw storage but low, predictable latency.

Power and cooling

A conventional rack draws single-digit to low double-digit kilowatts. Racks built for the latest accelerators draw well over a hundred. That breaks the assumptions of older facilities: air cannot carry the heat away, so liquid cooling, piped directly to the chips, becomes standard. Chapter 5 covers this.

The control plane

Above the hardware sits the software that allocates jobs, monitors health, manages identity and bills users. Whoever runs the control plane sees and can influence everything below it. This is where much of the sovereignty question hides, and Part III addresses it.

Training and inference

Two workloads shape the design. **Training** builds a model. It is a huge, synchronised, months-long job that wants the largest possible cluster in one place with the best network. **Inference** serves the finished model to users. It is many small, independent requests that want to be close to users and to scale with demand.

The distinction matters for sovereignty. Training is capital-intensive and bursty, and a country may reasonably choose to rent it from a partner while retaining the resulting weights. Inference touches user data and prompts constantly, so it is where residency and access rules bite hardest, and where domestic capacity is often most valuable.

Three classes of site

Class	Typical scale	Typical purpose	Sovereignty angle
Enterprise or sector	Under a few megawatts	Private inference, fine-tuning, regulated workloads	Full control is feasible and affordable; capability is limited.
National or regional	Tens of megawatts, up to around a hundred	Shared public compute, research, national models, government workloads	The main arena for policy choices about ownership, access and operators.
Frontier campus	Hundreds of megawatts to gigawatts	Training very large models, hyperscale inference	Requires partners, capital and chip access few states can supply alone.

Scales are indicative ranges for orientation, not standards.

The vocabulary has shifted with the scale. Vendors and governments now speak of “AI factories,” facilities designed as one integrated machine that turns electricity and data into models and answers, and of “gigafactories” for the largest. The metaphor is useful because it stresses throughput and utilisation, not floor space.

In short. *An AI data center is dense compute, a demanding network and a control plane, wrapped in power and cooling. Training and inference pull the design in different directions, and each has its own sovereignty risks.*

5. Power, cooling and siting

The scarcest input is not the chip. It is a reliable megawatt in the right place.

Ask anyone who has tried to build an AI data center what slowed them down and the answer is rarely the servers. It is the grid connection. Utilities in many regions have multi-year queues for large new loads, and transmission upgrades take longer than a data center does to build.

Power

Scale is now measured in megawatts and gigawatts, not racks. A single gigawatt is the output of a large power station, enough for hundreds of thousands of homes. Announcements in the multi-gigawatt range should be read as ten-year ambitions gated by generation, transmission and permitting, not as buildings that will exist next year.

Operators have four broad levers.

- **Grid supply**, ideally under long-term contracts with diversified feeds and a credible expansion plan from the utility.
- **Power purchase agreements** with renewable or nuclear generators, which fix price and support clean-energy commitments, though they do not by themselves guarantee delivery at the moment of need.
- **On-site generation**, from gas turbines to fuel cells and, increasingly discussed, small modular reactors. This raises resilience and speeds connection, at the price of fuel logistics and emissions.
- **Storage and demand flexibility**, including batteries and the ability to throttle non-urgent training runs during grid stress.

Energy is itself a sovereignty layer. A country that imports the gas its data centers burn has moved its dependence, not removed it.

Cooling

Almost every watt that enters a data center leaves as heat. Older designs blow chilled air through the racks. That stops working at very high density. Modern AI halls use **direct-to-chip liquid cooling**, where coolant circulates through plates mounted on the processors, often supplemented by air for the remainder. **Immersion cooling**, where servers sit in dielectric fluid, is used in some deployments.

The heat then has to go somewhere. Options include evaporative cooling, which consumes water, dry coolers, which consume more electricity, and heat reuse, where warmth feeds district heating or industrial processes. Efficiency is often summarised by power usage effectiveness (PUE), the

ratio of total facility power to IT power. Good modern facilities reach roughly 1.1 to 1.3, depending on climate and design.

> **In practice: hot places and cold places**
>
> Cold, hydro-rich regions offer cheap cooling and clean power, which is why the Nordics attract training clusters. Hot regions, including much of the Gulf and South Asia, face a harder thermal problem and often higher water stress, but they have demand, capital, or both. Neither is better in general. The right site depends on which constraint you can most cheaply relax.

Where to put it

Siting is a trade among six things.

1. **Power** availability, price and expansion room.
2. **Cooling** climate and water access.
3. **Connectivity**, meaning diverse fibre routes and proximity to exchange points and cable landings.
4. **Latency** to users, which matters for inference far more than for training.
5. **Jurisdiction and politics**, including stability, permitting speed and local acceptance.
6. **Hazards**, from flooding and seismic risk to physical security.

Because training tolerates distance and inference does not, many national strategies split the two: a few large training sites where energy is cheap, and distributed inference capacity near population centres. The sovereignty question is whether every one of those sites falls under the same governing authority.

The community problem

Data centers are increasingly contested locally over electricity prices, water use, noise and land. A project with perfect technical sovereignty can still stall for lack of local consent. Serious plans budget time and money for engagement, for transparent reporting of energy and water use, and for sharing benefits with the host community.

In short. *Power and cooling set the pace of every build. Treat energy as a sovereignty layer, and expect community consent to be part of the critical path.*

6. Silicon and the supply chain

The layer where almost every country, including the wealthy ones, depends on someone else.

You can own the land, pay the staff and hold the keys. You still cannot conjure an advanced accelerator. The supply chain behind one runs through a handful of firms and places, and each choke point is a place where a foreign government or company can affect you.

The choke points

Stage	Concentration	Why it matters for sovereignty
Accelerator design	A small number of firms dominate the market for training-class chips.	Software ecosystems create lock-in; roadmaps and allocations are set elsewhere.
Advanced fabrication	The leading-edge nodes are made in very few facilities, chiefly in one region.	A regional crisis or a policy decision can cut global supply.
High-bandwidth memory	Three suppliers make nearly all of it.	Memory availability often limits how many accelerators can ship.
Advanced packaging	A limited number of specialised plants.	Capacity constrains output even when wafers are available.
Servers and networking	Manufacturing concentrated in a few countries; networking in a few vendors.	Additional dependency on firmware and interconnect ecosystems.

Licences: the invisible switch

The most sovereignty-relevant fact about advanced chips is that shipping them abroad is regulated by the exporting country. Licences can be granted, limited by quantity, tied to security conditions, or withdrawn. In late 2025, for example, the United States authorised exports of tens of thousands of its latest accelerators to the UAE and Saudi Arabia after a period of uncertainty that had held up large investments, according to industry reporting. The episode shows the pattern: capacity is built in the expectation of licences, and licences follow diplomacy.

The implication is blunt. **A sovereign compute programme that depends on export licences has a conditional sovereignty.** That does not make it worthless, but the condition should appear in the risk register alongside power and cooling.

What you can actually do

- **Diversify suppliers.** Where workloads allow, qualify a second accelerator family so no single vendor or licence regime can halt you. This is easier for inference than for large-scale training.

- **Invest in portability.** The software layer, meaning frameworks, compilers and kernels, determines how painful it is to switch. Open toolchains reduce lock-in, though real-world performance gaps between ecosystems remain and should be measured, not assumed away.
- **Stock spares and plan refresh.** Accelerators fail and generations turn over quickly. Contractual guarantees for spares, repairs and next-generation access are worth more than headline unit prices.
- **Verify what is inside.** Insist on a bill of materials for hardware and software, on hardware roots of trust and secure boot, and on the ability to inspect and control firmware updates. An update channel you cannot inspect is a lever held by someone else.
- **Buy in tranches.** Staged procurement limits exposure if licences or prices change, and lets you learn between phases.
- **Support domestic capability where it is credible.** Programmes for chip design, packaging and assembly can raise resilience over a decade. They should be funded as long-term options, not counted as near-term compute.

> **The firmware question**
>
> Every server contains management controllers and firmware with deep privileges. A sovereign design treats them as untrusted until proven otherwise: isolate management networks, control who can push updates, verify signatures against keys you can audit, and keep the ability to run for months without any update from the manufacturer.

In short. *Silicon is the layer you can rarely own. Aim to diversify, stay portable, control firmware, and write the licence risk into your plan.*

Part III: The control layer

Software, keys, people, data, models and security: where control is actually exercised.

7. Software, keys and operations

Whoever administers the machines can do anything the machines can do.

Physical sovereignty is visible: you can point at the building. Control-layer sovereignty is invisible, which is why it is so often under-specified. This chapter covers the delivery models on offer, then the three mechanisms that decide who really holds control: keys, people and updates.

Four delivery models

Model	What you gain	What stays foreign	Best fit
Hyperscaler sovereign region or controls	Speed, mature tooling, in-country residency, contractual and technical limits on access.	Ownership, parent-company legal exposure, much of the software and roadmap.	Regulated but not critical workloads that need scale fast.
Licensed technology, domestically owned and operated	Domestic legal entity, local operations, access to hyperscale software under licence.	Intellectual property and update pipelines; dependency on the licensor's continued cooperation.	Government, critical infrastructure and heavily regulated industries.
National build, own and operate	Maximum control over every layer you choose to control.	Silicon and often core software, unless deliberately diversified.	National compute programmes, defence, strategic research.
Private or air-gapped on-premises	Complete isolation for the most sensitive data.	Hardware and software supply, plus the cost of running everything yourself.	Classified, safety-critical and highly confidential workloads.

Several European joint ventures between local firms and global cloud providers follow the second model.

These are points on the ladder from Chapter 1, not competing ideologies. A mature organisation uses more than one: hyperscale services for low-sensitivity work, a licensed domestic cloud for regulated workloads, and an isolated enclave for the crown jewels. The discipline is to *classify workloads first* and choose the model second. Chapter 13 explains how.

Keys

Encryption protects data at rest and in transit. Who holds the keys decides who benefits from that protection. The options form a spectrum:

- **Provider-managed keys.** Convenient, and the provider can read your data.
- **Customer-managed keys inside the provider's service.** Better, but the provider's systems still process the keys.
- **Customer-controlled keys in your own hardware security modules.** The provider cannot decrypt without you. This is the common bar for sovereign designs, sometimes called "hold your own key."
- **Independent key hierarchy and identity provider,** operated in-country, with revocation you have tested.

Confidential computing extends protection to data in use, by running workloads inside hardware-isolated environments that the host operator cannot inspect. Modern accelerators increasingly support it. It is a valuable control, with two caveats: you still trust the silicon vendor's implementation and attestation chain, and it can carry performance and complexity costs. Use it as one layer, not as a substitute for the rest.

People

Administrators hold the most powerful credentials in the building. Sovereign operations therefore start with staffing: in-country personnel, security clearances where the workload warrants them, and clear separation between the people who run the platform and the people who can see customer data.

Vendor support is the classic weak point. Ask specifically: can the vendor connect remotely to production systems? Who approves it? Is every session recorded and reviewable by you? Can you deny it entirely? A workable pattern is **break-glass access**: no standing vendor access, time-limited elevation approved by the customer, fully logged, and reviewed afterwards.

Updates and telemetry

Modern systems phone home. Telemetry, licence checks and automatic updates each create a channel from the vendor into your environment. Each channel can leak information, and any can in principle be used to change behaviour or disable a system. Sovereign design controls them: mirror update repositories inside the boundary, test and sign updates locally, minimise and inspect telemetry, and make sure the platform keeps working if the licence server is unreachable.

The test worth applying is blunt: *if the vendor's connection were cut tomorrow, how long could you keep running, and what would fail first?* Write the answer down. That number is your operational autonomy.

The software stack

Open standards and open-source components, from container orchestration to job schedulers and inference servers, make it possible to change suppliers and to audit what runs. They do not make a system sovereign on their own. You also need the skills to operate them, a plan for security patches, and source escrow or licence guarantees for any proprietary components you keep.

In short. *Control sits with whoever holds the keys, the administrator accounts and the update channel. Hold your own keys, supervise vendor access, and test how long you can run alone.*

8. Data and models

The point of the stack. A sovereign building with an unsovereign model inside is a well-guarded empty vault.

Data

Data is the layer most organisations already understand, and the one they most often under-manage. Sovereign data practice rests on four habits.

- **Classify.** Know which datasets are public, sensitive, restricted or secret, because placement rules follow from classification.
- **Trace lineage.** Record where data came from, what consent or licence covers it, and where copies have travelled. Training on data whose provenance you cannot show creates legal and reputational risk.
- **Keep it resident where required,** including derived data. Embeddings, logs and fine-tuning sets can leak the source.
- **Invest in supply.** Sovereignty means little if you have nothing local to train on. National dataset platforms, such as India's AIKosh, treat data as a public good, giving domestic builders licensed, cleaned material in local languages instead of forcing them to scrape or buy abroad.

Three ways to get a model

Governments and companies use three strategies, and the difference between them is a difference in control and cost.

Strategy	What it is	Control	Cost and skill
Consume	Use a foreign provider's model through an interface, possibly hosted in-country.	Low. The provider controls weights, behaviour and roadmap.	Lowest. Fastest to start.
Adapt	Take an open-weight model and fine-tune it on local data and languages, then run it on your infrastructure.	Medium to high. You hold the weights you run, though the base is someone else's.	Moderate. Needs a capable team and good data.
Build	Train a model from scratch, at sub-frontier or frontier scale.	Highest over the model itself.	Highest. Frontier training is within reach of very few.

The CNAS tracker finds that most countries pursuing model projects choose the middle path, adapting open-weight models on local data to capture language and context at a fraction of the cost, while a smaller number train sub-frontier models from scratch. It counted 21 such from-

scratch government-backed projects by June 2026, more than double the number at the end of 2024. Examples include India's Sarvam models, released in early 2026 with government compute support, and open models from the UAE's Technology Innovation Institute.

The middle path is rational, and it carries a subtle dependency. An adapted model inherits its base model's design, training data and biases. Ask what licence governs it, whether the licence can change, and whether you could reproduce or replace the base if you had to.

Weights are crown jewels

Model weights are the concentrated result of enormous compute and data. Copying them is trivial if security is weak, and a copied model can be used, modified or resold beyond your control. Treat weights as a controlled asset: restrict access to a small, vetted group; keep them in hardened environments; log every access; and match the security level to the threat, from opportunistic criminals up to state-level adversaries. Published security frameworks for AI model weights set out graduated levels of protection and are a useful starting point for thinking about what "enough" means.

Where the prompts go

Sovereignty also covers inference. Every prompt and response is data. If a nominally national service quietly routes requests to a foreign endpoint for some tasks, or logs them where a foreign provider can read them, the sovereignty claim fails at run time, whatever the training story was. Map the actual data path for a request from user to model and back, and check it against the claim.

> **Sovereign model, sovereign infrastructure: two different claims**
>
> A locally trained model running on rented foreign infrastructure is sovereign in one respect and exposed in another. A national data center running only foreign-provided models is the reverse. Sovereignty requires both layers, and it is worth asking of any project which of the two it has actually secured.

In short. *Classify data and trace its lineage. Choose deliberately among consuming, adapting and building models. Guard the weights, and verify the real path of every request.*

9. Security and resilience

A sovereign system that cannot survive an outage or an attack has only moved the failure closer to home.

Sovereignty and security overlap but are not the same. Sovereignty asks who controls the system. Security asks whether anyone can misuse or break it. A design can be fully domestic and easy to compromise. It can also be well defended and foreign-controlled. You need both.

Know the threat

Threat	What they want	Typical route	Main defences
Foreign legal compulsion	Access to data or a service halt	Court orders and statutes directed at the operator or its parent	Domestic operator, customer-held keys, contractual challenge and notification duties
State-level cyber intrusion	Espionage, theft of weights, disruption	Zero-days, phishing, compromised administrators, supply-chain implants	Segmentation, hardware roots of trust, privileged-access management, continuous monitoring
Insider misuse	Theft, sabotage, coercion	Privileged staff and contractors	Vetting, separation of duties, two-person rules, immutable audit logs
Supply-chain compromise	Persistent access	Firmware, drivers, packages, hardware modification in transit	Bills of materials, signed builds, inspection, staged rollouts
Physical attack or sabotage	Destruction, coercion	Intrusion, attack on power, fibre or cooling, and in conflict zones, kinetic strikes	Site hardening, geographic diversity, redundant feeds, rehearsed failover
Accident and failure	None. It just happens.	Grid trips, cooling faults, software bugs, human error	Redundancy, chaos testing, disaster-recovery drills

Design principles

1. **Assume compromise somewhere.** Segment the network so one intrusion does not reach everything. Use zero-trust access, where no connection is trusted merely because of where it comes from.
2. **Protect the management plane hardest.** Out-of-band management networks, administrator consoles and update systems are the highest-value targets.
3. **Separate duties.** No single person should be able to both change a system and erase the record of the change.
4. **Make audit trails tamper-evident,** and keep them where the customer, not only the operator, can inspect them.
5. **Rehearse failure.** A failover plan that has never been run is a hope. Schedule drills, including loss of a site, of a carrier, of a vendor connection and of a key.

Resilience is geography

Concentrating a nation's AI capacity in one campus is efficient and fragile. Resilient designs use multiple sites, on different grid feeds and different fibre routes, with the ability to shift critical workloads between them. Recent conflicts have made the physical protection of digital infrastructure a live planning assumption in some regions, and subsea cables, which carry most international traffic, have shown themselves to be exposed. Plan for degraded modes: what runs if international links are cut, or if a site is lost, and who decides what gets priority.

> **Air-gap realism**
>
> An air gap is a strong control, and it is also a process that decays. Data has to get in and models have to get out, and every transfer path, from removable media to review stations, is a possible route for compromise. Design the transfer process as carefully as the enclave.

Compliance is not security

Certifications and audits establish a baseline and help procurement. They do not prove a system resists a determined adversary. Treat them as a floor and test above it, with red-team exercises, penetration tests and independent review.

In short. *Name your adversaries and defend the management plane first. Use several sites, rehearse failure, and treat compliance as the minimum.*

Part IV: The landscape

How different regions are building, and the paradox that sits at the centre of all of it.

10. Four regional approaches

Europe pools, India subsidises, the Gulf spends, and others partner. Each optimises a different layer.

No two national programmes look alike, because they start from different assets: capital, energy, talent, markets, alliances. This chapter looks at four patterns, with the caveat that the situation is moving quickly and figures come from a mix of official and secondary sources. The aim is to learn the patterns, not to memorise the numbers.

Europe: pool the demand, regulate the terms

Europe's approach is public-private and multinational. The European Commission launched the InvestAI initiative in February 2025, targeting €200 billion of AI investment, with €20 billion set aside for large-scale AI gigafactories. In parallel, a network of smaller "AI Factories" hosted in member states, built around existing supercomputing centres, gives researchers and start-ups access to compute.

The next step arrived in summer 2026. On 30 July, the European High Performance Computing Joint Undertaking opened a call for tenders to select consortia that will build and operate up to seven gigafactories, four medium-scale with at least 75,000 advanced accelerators and three large-scale with at least 100,000. Proposals are due on 12 November 2026. An earlier informal call drew 77 proposals covering 60 sites in 16 member states. Reports put the combined public and private financing in the region of €30 billion, with roughly one public euro expected to draw about two private.

The design includes sovereignty conditions: the amended EuroHPC rules formalise a majority-ownership requirement for gigafactory consortia and exclude suppliers deemed high-risk from their supply chains. The EU and participating states will jointly buy a share of compute time to allocate to European users. Critics ask whether the money will buy strategic autonomy or subsidise the same non-European suppliers it hopes to reduce dependence on, since the chips will largely be imported. That question belongs to every programme in this chapter.

India: subsidise access, build the model layer

India's IndiaAI Mission was approved in March 2024 with an outlay of about ₹10,372 crore (roughly $1.25 billion) over five years, of which the compute pillar is the largest at around ₹4,563 crore. The model is unusual. The government does not own the GPUs. It *empanels* private providers, among them large Indian conglomerates and data-center operators, who make capacity available to start-ups, researchers and public agencies at subsidised hourly rates.

Reported capacity varies by source: figures from about 34,000 to more than 38,000 GPUs appear for mid-2026, with a stated goal of 100,000 public GPUs by December 2026 and national capacity, including private buildouts, expected to exceed that. Reported subsidised prices also

vary, from under ₹100 to about ₹150 per GPU-hour depending on tranche and source. Alongside compute, the mission funds foundation models, including Sarvam's open models released in February 2026, and a national datasets platform.

The debate around it is instructive. One line of criticism observes that only a small fraction of the budget had been disbursed in the first two years, attributed to procurement complexity for advanced hardware. Another asks whether a state-subsidised pool of privately-owned GPUs is a sovereign stack or a subsidised cloud, and argues that the test is whether it produces capability India actually owns: models, data systems and governance. The upshot: cheap access is a real accomplishment, and sovereignty is a separate outcome.

The Gulf: spend at scale, buy the stack

The UAE and Saudi Arabia are funding gigawatt-scale AI infrastructure directly from sovereign wealth. Stargate UAE, developed by G42 with OpenAI, Oracle, NVIDIA, Cisco and SoftBank, envisions a 5-gigawatt campus in Abu Dhabi, with a first 200-megawatt cluster targeted to go live in 2026. Saudi Arabia's HUMAIN, backed by the Public Investment Fund, has announced partnerships for AI factories with capacity targets in the multi-gigawatt range by the mid-2030s.

Two points from the analysts are worth keeping. First, *announced* and *usable* capacity are different quantities. The region's operational third-party data center capacity was reported at about 500 megawatts in early 2026, while headline campuses are largely sovereign self-builds whose capacity may never reach the open market. A chief information officer cannot lease a slice of a national programme. Second, the compute runs on US-approved silicon, so the sovereignty is conditional on export licences, as Chapter 6 described.

Partners and specialists: Canada, Australia and telecom operators

Several countries are using structured partnerships. Canada ran a call for proposals from 15 January to 15 February 2026 for large-scale sovereign AI data centres, using memoranda of understanding with selected proponents to explore how commercial capacity can be built in the country. In Australia, the data-center operator NextDC signed an agreement with OpenAI in December 2025 to develop sovereign AI infrastructure at a Sydney campus. Telecom operators pitch themselves as natural hosts, arguing that their owned real estate, network and regulatory standing provide a ready foundation for national AI capacity.

Region	Financing pattern	What it optimises	Main dependency	Watch item
Europe	EU funds and private capital in consortia	Pooled demand, ownership rules, regulatory terms	Imported accelerators	Selection outcomes after the November 2026 deadline
India	Public subsidy of privately owned capacity	Access and affordability, local models	Imported accelerators, disbursement pace	Whether the 100,000-GPU goal is met and utilised

Region	Financing pattern	What it optimises	Main dependency	Watch item
Gulf	Sovereign wealth funds and global partners	Scale and speed	US export licences, foreign technology partners	Usable capacity against announced capacity
Canada, Australia and others	Partnership and MOU models	Attracting private investment under national conditions	Foreign partners and chips	Which commitments become buildings

Simplified for comparison. Each programme is broader than one row can capture.

The pattern across all of them

Taken together, the CNAS tracker's conclusion is sobering. Governments have made AI sovereignty a priority, and the gap between intent and action is large. Outside China, it is hard to picture any sovereign compute project fully independent of the US technology stack. The programmes are real and useful. They change who hosts and who governs. They mostly do not remove upstream dependence.

In short. *Europe pools, India subsidises, the Gulf spends, and partnership models spread the risk. All of them still import their chips, and all should be judged on delivered, usable, governed capacity.*

11. The paradox of dependence

Every route to sovereignty passes through someone else's technology. The skill is choosing whose, and on what terms.

Here is the uncomfortable centre of the subject. To build sovereign AI infrastructure today, almost everyone must buy accelerators designed by a foreign firm, made in a third country, under licences granted by a fourth. The more sophisticated the ambition, the more true this is. A national data center can reduce your exposure to a foreign cloud provider while increasing your exposure to a foreign chip supplier.

This is not a reason to give up. It is a reason to be precise about what is being achieved, and to stop measuring progress by a yes-or-no test that nobody can pass.

Exposure moves, and that can be a good trade

Consider two designs for a government's health-data AI service. In the first, the data sits in a hyperscaler's in-country region under contractual controls. In the second, it sits in a domestically owned facility using the same chips and an open software stack. Both depend on the same chip vendor. But the second removes exposure to foreign legal compulsion of the operator, foreign control of the platform, and foreign administrative access. It has also added operational burden and cost. Whether that is a good trade depends on the data, and that is the point: **the answer comes from the workload, not the slogan.**

Five principles

1. **Know your exit cost.** For every critical dependency, write down how long and how much it would cost to leave. A dependency with a low exit cost is a convenience. One with a high exit cost is a hostage.
2. **Diversify where diversification is cheap.** Two networking vendors, two key managers, two sites and two power feeds cost far less than a second accelerator ecosystem. Buy independence in the places that are affordable first.
3. **Prefer contractual and technical guarantees to promises.** Step-in rights, source-code escrow, spares commitments, data-return obligations and audit rights turn goodwill into enforceable terms. Combine them with technical controls, such as customer-held keys, that work even if the contract is contested.
4. **Keep option value.** Choose open standards, portable formats and containerised workloads so that switching is possible even if you never do it. The ability to leave is what gives you negotiating power while you stay.
5. **Federate.** Pooling demand and capacity with trusted neighbours, as the European model does, raises purchasing power and resilience. Alliances are a form of sovereignty, provided the partners' interests really align.

> **The goal is not to depend on no one. It is that no single party can hurt you, and that you know what it would cost to switch.**

“Sovereign enough” is a legitimate answer

Not every workload needs the top rung. Public-facing chatbots on published information, internal productivity tools and non-sensitive analytics can run on ordinary commercial services. Spending sovereign-tier money on them wastes resources that should go to the workloads that need it. The discipline is to define *enough* for each class of work, and to write it down so that decisions are consistent and defensible.

The honest ledger

It helps to keep, for any programme, a plain-language ledger with three columns: what we control now, what we depend on and how we would mitigate it, and what we would need to change to control more. A programme that can produce that ledger is in good shape, whatever rung it is on. One that cannot has not yet understood its own position.

In short. *Dependence cannot be eliminated, only managed. Know your exit costs, diversify cheaply, contract for guarantees, keep options open, and define “enough” per workload.*

Part V: Practice

The economics, a build playbook, governance, and what to watch as the field moves.

12. The economics of sovereignty

Sovereign capacity costs more than the cheapest alternative. The question is what you buy for the difference, and whether the machines stay busy.

Sovereignty is a premium product. You pay for domestic operations, duplicated capability, smaller scale and slower access to the newest hardware. Whether the premium is worth it is a business and policy judgement. It should be an informed one, and that means understanding the cost structure.

Where the money goes

Three cost blocks dominate. The **facility**, meaning the building, power distribution and cooling, is commonly estimated in the range of ten million dollars or more per megawatt of capacity, with high-density liquid-cooled designs at the upper end. The **IT equipment**, meaning accelerators, servers, networking and storage, typically costs several times as much as the facility that houses it. And **operations**, chiefly electricity, staff, maintenance and licences, recur every year. Treat all of these as rough orders of magnitude and get current quotes before planning.

Two features make the economics unforgiving. Accelerator generations turn over every year or two, so hardware can lose competitive value faster than its accounting life suggests. And the largest cost, the equipment, is spent up front, while revenue arrives only if the machines are used.

The utilisation trap

The number that decides most projects is utilisation: the share of hours in which the accelerators do paid or valuable work. Idle chips still depreciate. The arithmetic is straightforward. Here is an illustrative calculation, with assumed inputs, not market data.

Assumption (per accelerator)	Illustrative value
Equipment share, all-in (server, network, storage)	$50,000
Facility share, at about 1.5 kW including overhead, $12M per MW	$18,000
Straight-line over five years	$13,600 per year
Electricity, 1.5 kW around the clock at $0.08 per kWh	About $1,050 per year
Staff, maintenance and other running costs	About $1,500 per year
Total annual cost	About $16,150

Utilisation	Useful hours per year	Cost per useful accelerator-hour
90 percent	7,884	About $2.05
70 percent	6,132	About $2.63
40 percent	3,504	About $4.61

Same hardware, same building. Cost per useful hour more than doubles when utilisation falls from 90 to 40 percent.

The lesson is not the specific dollar figures. It is that a sovereign facility running at 40 percent is more than twice as expensive per useful hour as the same facility running at 90, and public projects are especially prone to low utilisation when capacity is built before demand exists.

Building demand before building capacity

The remedies are structural.

- **Secure anchor tenants first.** Government agencies, research programmes and large regulated enterprises that commit volume in advance turn a speculative build into a financeable one. The European plan to buy compute time jointly is this logic applied at multinational scale.
- **Stage the build.** Commission capacity in tranches tied to signed demand and to hardware generations, rather than buying five years of chips at once.
- **Blend workloads.** Mix steady inference with flexible training that can fill gaps, so the machines rarely sit idle.
- **Make access easy.** Subsidised, well-supported access for start-ups and researchers, of the kind India has attempted, creates the demand that justifies the supply, provided the support and skills exist to use it.
- **Contract for refresh.** Negotiate access to next-generation hardware and trade-in terms up front.

Build, buy or rent

Choice	Choose it when	Watch for
Rent from a commercial provider	Workload is low-sensitivity, demand is uncertain, or you need speed.	Exit costs, price changes, jurisdiction.
Rent sovereign-grade capacity from a domestic or licensed operator	Workload is sensitive but a domestic market exists and can meet your controls.	Whether the claim survives the layer-by-layer test in Appendix A.
Co-build with partners	Scale exceeds what you can fund alone and partners' interests align.	Governance disputes, licence dependence.
Build and own	Workloads are strategic and steady, and you can staff and secure the site.	Utilisation, obsolescence, skills.

Overbuild risk

Demand for compute could disappoint. Efficiency breakthroughs, where smaller or cheaper models match larger ones, can reduce the hardware needed per unit of useful work, and a sudden efficiency shock has already rattled expectations once. Grid constraints and permitting delays can

also slow deployment and strand capital. A sovereign programme should plan for a range of demand outcomes, and prefer designs that can be repurposed, resold or extended over designs that only make sense at maximum forecasts.

In short. *Utilisation makes or breaks the economics. Secure demand before capacity, stage the build, contract for refresh, and plan for demand to surprise you in either direction.*

13. A build playbook

Eight steps from mandate to operation, with the questions that expose weak claims.

This chapter is written to be used. It applies equally to a national programme, a regulated enterprise and a vendor preparing a bid. The steps are sequential, but expect to loop back as you learn.

1. **Define the mandate and the threat model.** State in a page what the infrastructure is for, whom it serves, and which adversaries and failures you are defending against. Without this, every later choice is arbitrary. Name the events that would count as failure: a foreign order to disclose data, a chip cut-off, a grid outage, a leaked model.
2. **Classify the workloads.** Sort each workload by data sensitivity and criticality. The table below is a starting point. Most organisations discover that only a minority of workloads need the top tier.
3. **Pick a sovereignty tier per class.** Map each class to a rung on the ladder from Chapter 1, and to a delivery model from Chapter 7. Write down what "enough" means for each and who signed off.
4. **Site and secure power.** Choose sites for energy, cooling, connectivity and resilience together (Chapter 5). Start the utility conversations early, because interconnection is usually the longest lead item.
5. **Design for portability and exit.** Choose open standards, containerised workloads, customer-held keys and documented interfaces. Design the exit before you need it, including how data and models would move.
6. **Procure with sovereignty clauses.** Put the guarantees in the contract: data location, subprocessor lists, handling of government access requests, audit rights, step-in rights, spares and refresh terms, and exit assistance (Chapter 14).
7. **Operate, audit and drill.** Staff in-country, supervise vendor access, monitor continuously, and rehearse failure. Publish what you can about security and performance, and run independent reviews.
8. **Refresh and evolve.** Review the scorecard every year. Hardware generations, regulations and geopolitics all shift, and a design that was right in year one drifts unless someone owns the review.

A classification to start from

Class	Examples	Typical minimum placement
Public	Published information, marketing, open research	Commercial cloud, any region, ordinary controls.

Class	Examples	Typical minimum placement
Internal	Business documents, code, analytics without personal data	Commercial cloud with strong contractual terms and regional residency.
Sensitive	Personal, health, financial and regulated data	Assured or operated tier, customer-held keys, domestic legal entity.
Restricted	Critical infrastructure, government services, high-value model weights	Operated or autonomous tier, cleared in-country staff, supervised vendor access.
Secret	Classified, defence and intelligence	Autonomous tier or air-gapped, national-only supply and staff.

A starting point. Adapt to your legal environment and sector rules.

Ten questions to put to any vendor

1. Which legal entity operates the service, and under whose law? Who owns that entity?
2. Can any government outside our jurisdiction compel you or your parent to disclose our data? How do you handle such requests, and will you tell us and contest them?
3. Who holds the encryption keys, and can you demonstrate you cannot decrypt without us?
4. Who has privileged access to our environment, from where, and how is it approved, logged and reviewed?
5. Which software components would stop working if your connection to us were cut? For how long can we run?
6. Which accelerators and firmware are used, from whom, under which export licences, and what happens if a licence changes?
7. What telemetry leaves our environment, and can we inspect and block it?
8. What are the exit terms: data return, format, timeline, cost and assistance?
9. What independent audits and certifications do you hold, and may we test the controls ourselves?
10. Which failures have you rehearsed in the last year, and what did you learn?

Ten red flags

- “Sovereign” is defined only as “located in our country.”
- A partner is named as a sovereignty guarantee without any description of control rights.
- Vendor remote access is “standard” and cannot be disabled.
- Keys are managed by the provider “for convenience.”
- The capacity is announced in gigawatts, but nobody can say what is energised today.
- There is one supplier for accelerators, one site and one carrier.
- The exit plan is a sentence, not a document.

- There is no funding for operations staff or skills.
- Utilisation assumptions are not stated or are far above comparable facilities.
- Audit rights exist on paper and have never been exercised.

In short. *Start with the mandate and the threat model, classify before you choose, design the exit early, put guarantees in contracts, and review every year.*

14. Governance and compliance

Rules define what you must do. Governance is how you make sure it stays done.

Sovereign infrastructure operates inside a thicket of overlapping rules, and the rules themselves change. This chapter maps the main families and sets out the governance practices that keep a system honest over time. It is a guide to the questions, not legal advice. Anyone making binding decisions should consult qualified counsel in the relevant jurisdictions.

The rule families

- **Data-protection law.** Regimes such as the European GDPR, India's Digital Personal Data Protection Act and the national laws of Gulf states govern personal data, cross-border transfers and individual rights. They set the floor for residency and access.
- **AI-specific regulation.** The European AI Act applies obligations in phases according to risk, with particular duties for general-purpose models and high-risk uses. Other jurisdictions are developing their own approaches. Check the current timetable, because dates and details have been debated and adjusted.
- **Sector rules.** Financial, health, telecoms and critical-infrastructure regulators impose outsourcing, resilience and audit requirements. Financial services regulation in Europe, for example, sets specific expectations for operational resilience and for the management of critical third-party providers.
- **Security and certification schemes.** National cloud-security schemes, such as France's SecNumCloud and Germany's C5, and international standards such as ISO 27001, provide auditable baselines. Some, like SecNumCloud, embed explicit protection against foreign legal access. European-level cloud certification has been the subject of long debate about how far sovereignty criteria should go.
- **Extraterritorial laws.** Several major jurisdictions assert authority over data held by companies they govern, wherever it is stored. Where such a law can reach your provider, residency alone does not protect you. These laws are the reason Chapter 2 places law and jurisdiction at the top of the stack.

Contract clauses that carry weight

Regulation sets the frame. Contracts do the daily work. For sovereign arrangements, the clauses to look for are:

- **Data location and movement,** including a ban on transfers without written approval and a complete list of subprocessors.

- **Government-access handling,** requiring the provider to notify you where lawful, challenge overbroad requests, and disclose the minimum necessary.
- **Audit and inspection rights,** including the right to test controls and to review logs of privileged access.
- **Step-in and continuity rights,** which let you take over or continue operations if the provider fails, withdraws or is compelled to stop.
- **Escrow,** for source code or critical configuration where proprietary components matter.
- **Supply commitments,** for spares, repairs, security patches and hardware refresh.
- **Exit assistance,** with defined formats, timelines and costs.

Governance that lasts

1. **Name an accountable owner** for the sovereignty posture, with authority to stop a change that weakens it.
2. **Keep a register** of dependencies, exit costs and assumptions, and review it on a schedule and after any major event.
3. **Measure it.** Useful indicators include the proportion of workloads placed in the right tier, the share of keys held in customer-controlled hardware, the time you could run without vendor connectivity, the number of standing vendor privileges, spare-parts coverage in months, and the date of the last successful failover drill.
4. **Report transparently.** Publish, where security allows, the requests received from governments, the audits passed and the incidents suffered.
5. **Govern access to compute itself.** For public compute programmes, define who qualifies, how projects are approved, and how misuse is prevented. Know-your-customer checks and usage policies are becoming standard for high-capability systems.

In short. *Read the rule families for your sector and jurisdiction, put the guarantees in the contract, and give one accountable owner a register, a set of indicators and the authority to say no.*

15. Looking ahead

Three futures for sovereign AI infrastructure, and the signals that will tell you which is arriving.

Predicting this field is a good way to be wrong, so this chapter offers scenarios and signals in place of forecasts. The scenarios are not mutually exclusive, and the real world will blend them.

Three scenarios to 2030

Fragmentation

Export controls tighten, regional blocs harden, and each bloc builds its own stack with its own standards. Compute is more expensive and less interoperable, and models diverge in language, values and rules. Sovereignty is stronger on paper and costlier in practice. Smaller countries face a choice of bloc.

Federation

Trusted groups of countries pool demand, share facilities and agree common rules, as the European gigafactories attempt. Sovereignty becomes collective: no member holds every lever, and all hold enough. This works if partners' interests stay aligned and governance is credible.

Convergence

Hyperscalers and national champions merge their offers. "Sovereign" becomes a tier of commercial service, delivered by locally owned entities running licensed technology under strong contractual and technical controls. Costs fall and reach expands. The debate shifts from who builds to who can verify the controls actually work.

Signals to watch

- **Chip policy.** Changes in export licensing, and the emergence of credible non-dominant accelerator ecosystems.
- **Efficiency.** Whether smaller, cheaper models keep closing the gap with the largest, which would ease compute needs and favour distributed, sovereign-scale deployments.
- **Energy.** Progress on grid expansion, on-site generation and new nuclear, since power will pace every build.
- **Delivery.** The ratio of energised, utilised capacity to announced capacity in the big programmes. The gap is the best measure of seriousness.
- **Verification.** The maturity of tools that let customers prove, rather than trust, that provider controls work: attestation, transparent logging, independent audit.

- **Edge and on-device AI.** As capable models run on smaller hardware, part of the sovereignty question shifts from the data center to devices and local servers.
- **Talent.** Whether countries retain the engineers and researchers who operate and improve all this. Buildings without people are shells.

What good looks like

By the end of the decade, the best programmes are likely to share several traits. They will know precisely which layers they control and which they do not. They will have classified their workloads and matched each to a proportionate tier. They will hold their own keys, run their own operations, and rehearse their own failures. They will have diversified where it is cheap and negotiated hard where it is not. And they will measure success by useful, governed capacity delivered, not by megawatts announced.

Sovereignty over AI will not be a state you reach. It is a discipline you keep up: a running, honest account of who could pull which lever, and what you have done about it.

> **Ask of every project: which layers do we control, who can pull the rest, and how would we know?**

Appendices

A scorecard you can use today, a glossary, and the sources behind the numbers.

A. Sovereignty scorecard

Rate your project on each of the eight layers. The weakest layer sets the result.

For each layer, tick the statement that best describes your project today. The four levels match the ladder in Chapter 1: Resident, Assured, Operated and Autonomous. If you do not know, tick "don't know." A layer you have not examined cannot be called sovereign.

Then find the lowest level you ticked on any layer. That is your tier: the weakest layer, not the average, sets the result. Print these pages and tick the boxes by hand, or mark your choice in Word.

1. Law and jurisdiction

Whose law governs the operator, and who can compel access or a shutdown?

- ☐ Don't know or not assessed
- ☐ **Resident:** Data sits in-country, but a foreign provider and foreign law govern the service
- ☐ **Assured:** Contracts and technical limits restrict foreign access requests, though the parent is foreign
- ☐ **Operated:** Domestic entity under domestic law, using licensed foreign technology
- ☐ **Autonomous:** Domestic entity and domestic law, with no foreign parent that can be compelled

2. Ownership and control

Who owns the operator and the assets, and who can override decisions?

- ☐ Don't know or not assessed
- ☐ **Resident:** Foreign-owned operator
- ☐ **Assured:** Foreign-owned, with local governance safeguards or a golden share
- ☐ **Operated:** Domestic majority ownership, foreign technology licensed
- ☐ **Autonomous:** Domestic ownership of operator and assets, with technology rights you can step into

3. Operations and people

Who administers the systems and who can reach production?

- ☐ Don't know or not assessed
- ☐ **Resident:** Foreign staff and remote administration
- ☐ **Assured:** Local staff, with supervised and logged vendor access
- ☐ **Operated:** In-country cleared staff, vendor access only on approval with break-glass
- ☐ **Autonomous:** Fully in-country operations, no standing vendor access, able to run unassisted

4. Keys and identity

Who holds the encryption keys and runs the identity system?

- ☐ Don't know or not assessed
- ☐ **Resident:** The provider holds the keys
- ☐ **Assured:** Customer-managed keys inside the provider's service
- ☐ **Operated:** Keys in customer-controlled hardware modules the provider cannot read
- ☐ **Autonomous:** Own key hierarchy and identity provider, attested hardware, revocation tested

5. Platform software

How portable is the stack, and could you operate it yourself?

- ☐ Don't know or not assessed
- ☐ **Resident:** Proprietary stack tied to one provider
- ☐ **Assured:** Mostly proprietary, with documented export paths
- ☐ **Operated:** Open standards, with portability tested
- ☐ **Autonomous:** Open stack you can operate and rebuild yourself

6. Facility and network

How diverse are your sites, routes and exchange points?

- ☐ Don't know or not assessed
- ☐ **Resident:** Single site and single carrier
- ☐ **Assured:** Domestic site, limited diversity
- ☐ **Operated:** Multiple domestic sites and carriers
- ☐ **Autonomous:** Multi-site with diverse routes, domestic exchange and tested failover

7. Silicon and hardware

How exposed are you to one chip vendor, one licence or one firmware channel?

- ☐ Don't know or not assessed
- ☐ **Resident:** Single vendor, no spares, licence-dependent
- ☐ **Assured:** Single vendor with contractual supply assurances
- ☐ **Operated:** Validated second source, spares stocked
- ☐ **Autonomous:** Multi-vendor, verified firmware, multi-year spares and refresh plan

8. Energy and materials

How secure are power, cooling water and spare parts?

- ☐ Don't know or not assessed
- ☐ **Resident:** Single grid feed, no plan for long outages
- ☐ **Assured:** Grid supply plus backup generation
- ☐ **Operated:** Diverse contracted supply with days of on-site backup
- ☐ **Autonomous:** Diverse supply, on-site generation, secured cooling water

Reading your result

Lowest level ticked	What it means
Don’t know	You have not yet examined at least one layer. Start there.
Resident	Your data and compute are in-country, but at least one layer is controlled from outside. Treat this as a residency solution, and keep truly sensitive workloads elsewhere until the weakest layer improves.
Assured	Contracts and controls limit foreign access on every layer, though some still depend on partners. This suits many regulated workloads. Check that the guarantees are enforceable and tested.
Operated	Domestic control on every layer, with foreign technology under licence. This suits government and critical-infrastructure workloads. Focus on reducing the licence and supply dependencies that remain.
Autonomous	Domestic control end to end with diversified supply. This is rare. Guard against complacency, and keep the annual review going.

First move for your weakest layer

If this is weakest	Do this first
Law and jurisdiction	Move the operating entity, or the legally relevant control, under domestic law. Until then, use customer-held keys so compulsion of the operator yields nothing readable.
Ownership and control	Negotiate domestic majority ownership or explicit veto and step-in rights, and check they work if the partner disagrees.
Operations and people	Remove standing vendor access, require approval and full logging for any elevation, and train in-country staff to run the site alone.
Keys and identity	Move keys into hardware modules you control and test revocation. Consider running your own identity provider.
Platform software	Containerise workloads, adopt open standards and test a migration, so the ability to leave is real.
Facility and network	Add a second site and carrier, and rehearse failover including loss of international links.
Silicon and hardware	Qualify a second accelerator option for at least inference, stock spares, and take control of firmware updates.
Energy and materials	Secure a second power feed or on-site generation, and confirm cooling water and fuel logistics for a prolonged outage.

This is a thinking tool, not an audit. Its value is in the conversation it forces: for each layer, on what evidence did you choose that statement?

B. Glossary

Accelerator
A chip designed for the parallel arithmetic of AI, most often a GPU or a custom AI processor.

AI factory
A data center designed as one integrated system for turning power and data into trained models and answers. Larger versions are called gigafactories.

Air gap
Physical or logical isolation of a system from external networks.

Attestation
A cryptographic proof that hardware or software is in a known, expected state.

Break-glass access
Emergency, time-limited privileged access that requires approval and leaves a full record.

Confidential computing
Hardware-based isolation that protects data while it is being processed, including from the host operator.

Control plane
The software layer that allocates resources, manages identity and monitors a system, as distinct from the workloads it runs.

Data residency
The requirement or fact that data is stored and processed in a particular jurisdiction.

Data sovereignty
The principle that data is subject to the laws of the jurisdiction it belongs to, and who can compel access to it.

Direct-to-chip cooling
Liquid cooling in which coolant flows through plates attached to the processors.

Export controls
Government rules that restrict the sale or transfer of specified technologies across borders.

Extraterritorial law
A law that a state applies to persons, companies or data outside its territory.

Fine-tuning
Further training of an existing model on additional data for a specific purpose.

HBM
High-bandwidth memory, the stacked memory packaged next to accelerators.

Hold your own key (HYOK)

An arrangement in which the customer keeps the encryption keys so the provider cannot decrypt the data.

HSM

Hardware security module, a tamper-resistant device that stores and uses cryptographic keys.

Inference

Running a trained model to produce outputs for users.

Open-weight model

A model whose trained parameters are published for others to run and adapt, usually under a licence that may still restrict use.

PUE

Power usage effectiveness, the ratio of total facility energy to the energy used by IT equipment.

Scale-up and scale-out

The high-bandwidth links inside a server or rack, and the network between racks, respectively.

Sovereign cloud

Cloud services designed so that a defined jurisdiction retains control over data, operations and legal exposure.

Step-in rights

A contractual right to take over or continue operating a service if the provider fails or withdraws.

Training

The process of building a model by adjusting its parameters on large amounts of data.

Zero trust

A security approach that grants no implicit trust based on network location and verifies every request.

C. Sources and further reading

Public sources consulted for the figures and developments in this edition, accessed in September 2026.

Secondary and commercial sources are used where official figures were not available, and they vary in reliability. Where sources conflicted, the text says so.

Market and global overview

- Nokia, “Sovereign AI data center networks,” citing Gartner’s February 2026 forecast for sovereign cloud IaaS spending: nokia.com/data-center-networks/sovereign-ai
- Center for a New American Security, “Sovereign AI Index,” 2026: interactives.cnas.org/reports/sovereign-ai-index

Europe

- European Commission, “AI Gigafactories”: commission.europa.eu
- EuroHPC Joint Undertaking, call for AI Gigafactories, 30 July 2026: eurohpc-ju.europa.eu
- Law and Technology, analysis of the call and its deadline: lawandtechnology.eu
- Open Future, “Steering AI Investment”: openfuture.eu
- European Commission, “AI Factories”: digital-strategy.ec.europa.eu

India

- Overview of the India AI Impact Summit 2026: Wikipedia
- explainx, “India Sovereign AI Status 2026”: explainx.ai
- IFB Trend, on IndiaAI Mission funding and disbursement: ifbtrend.com
- AI Startup Impact, on whether the mission builds sovereignty or subsidised cloud: aistartupimpact.com
- Kapyn, “The IndiaAI Mission, explained”: kapyn.app

Gulf

- Data Center Dynamics, on Stargate UAE and HUMAIN: datacenterdynamics.com
- Data Center World Middle East, on announced versus usable capacity: datacenterworldmiddleeast.com

- Introl, “Middle East AI Revolution”: introl.com
- explainx, “UAE and Saudi AI 2026”: explainx.ai

Other programmes

- Innovation, Science and Economic Development Canada, “Enabling large-scale sovereign AI data centres”: ised-isde.canada.ca
- NextDC, company overview including the December 2025 agreement with OpenAI: Wikipedia

D. About this edition

This book was written by Claude, an AI assistant made by Anthropic, using the public sources listed above and general technical knowledge. Anthropic is a US company, and readers assessing a book about foreign technology dependence should weigh that context. The framework, the analysis and any errors are attributable to this edition, not to the sources.

The economic figures in Chapter 12 are illustrative calculations with stated assumptions, not market data. Regulatory descriptions in Chapter 14 are summaries, not legal advice. Fast-moving items, especially capacity figures, funding and regulatory timetables, should be checked against current official sources before use.

This edition is published by Anomny, a company working across AI, IoT and data centers.

Corrections and additions will improve later editions. The most useful contributions are dated, sourced and specific: which layer, which claim, which primary source.

Sovereign AI in the Data Center

First edition, September 2026

Published by Anomny