ACTIAN™
a division of HCLSoftware

WHITE PAPER

Data Sovereignty by Design

Why control, not location,
is the foundation of trust in the AI era.

Emma McGrattan
Chief Technology Officer

Table of Contents

Executive Summary

Data sovereignty has rapidly evolved from a narrow concern about where data is stored into a broader question of who controls data, under which rules, and how that can be proven. Enterprises now operate in a world where data flows across regions, partners, and platforms, where cloud and SaaS providers concentrate infrastructure and control planes, and where regulators and customers expect credible assurances that their data is handled in line with jurisdictional, contractual, and sector-specific requirements.

Over the past few years, interest in data sovereignty has spiked because a series of very concrete issues have exposed how fragile traditional assumptions really are. Cross-border data transfer mechanisms have been challenged or overturned, forcing organizations to rethink long-standing patterns for moving data between jurisdictions. New and evolving regulations with extra-territorial reach now specify not only where data may reside but also who may access it and for what purposes. Governments and regulated enterprises are imposing data localization and "sovereign cloud" mandates that require in-country processing, local key management, and clear separation from foreign legal reach. High profile breaches, SaaS misconfigurations, and uncontrolled use of external services, including AI powered tools, have shown how quickly sensitive operational, customer or intellectual property data can cross boundaries or leave governed environments entirely. Together, these developments have turned data sovereignty into a board-level concern rather than a niche compliance topic.

At the same time, modern analytics and AI are putting existing data architectures under strain. Even when organizations keep data nominally "in region", they often rely on centralized control planes, global monitoring, and shared services that blur jurisdictional lines. Data can be resident without being truly sovereign. Real sovereignty requires more than location. It requires control - clear decision rights over who may use which data and for what; enforcement - technical mechanisms that ensure policies are applied consistently across platforms and workloads; and proof - lineage, audit trails, and observability that demonstrate what actually happened.

This white paper argues that data sovereignty must therefore be engineered by design, not bolted on as an afterthought. That means shifting from document-centric governance to platform-native enforcement. It requires using federated architectures, rich metadata, and data contracts to express sovereignty rules, embedding those rules into data products, pipelines, and access layers, and ensuring that downstream usage, whether traditional analytics or newer AI-powered retrieval and summarization, operates within those constraints.

In this context, vector databases and Local AI are important not because they redefine sovereignty, but because they increasingly sit where data is accessed and interpreted. They become part of the control plane that determines what governed data can be semantically discovered and used, and under which regional or environmental boundaries.

Enterprises that take this approach, treating data sovereignty as a core architectural property rather than a compliance checklist, will be best positioned to adapt to regulatory change, enable safe adoption of AI and advanced analytics, and turn sovereignty from an impediment to innovation into a foundation for trusted growth.

The Sovereignty Wake-Up Call

For years, data sovereignty was treated as a specialized compliance topic. Teams would document where certain systems were hosted, map a few data flows, and rely on contractual language and certification badges to satisfy regulatory expectations. That era is ending.

Several trends have converged to push data sovereignty into the center of enterprise data strategy:

- **Global operating models.** Even traditionally regional businesses now depend on cross-border supply chains, distributed workforces, and multinational customer bases.
- **Platform concentration.** A small number of hyperscale providers and SaaS vendors host an increasing share of critical workloads, often with globally shared control planes.
- **Regulatory fragmentation.** Privacy, financial, healthcare, and sector-specific regulations are proliferating, often with jurisdictional nuance and extra-territorial effects.
- **Data-intensive analytics and AI.** Organizations are reusing the same data for multiple purposes, such as analytics, personalization, risk scoring, and AI-driven insight, which magnifies the impact of any sovereignty misstep.

In practice, sovereignty problems surface long before anyone receives a formal notice from a regulator. The early warning signals look like:

- Projects stalled or blocked because teams cannot clearly answer "Where is this data processed, and under whose control?"
- Regional carve-outs for certain customers or jurisdictions that require bespoke pipelines and platforms.
- Internal disagreements between legal, security, and data teams on what is actually allowed.
- Business units working around the governance framework when the official path is slow or unclear.

These are not just compliance failures; they are architectural failures. They reveal that sovereignty has not been built into the way data is collected, integrated, governed, and consumed.

Why location-based sovereignty fails

The most common working assumption about sovereignty is deceptively simple: if data is stored in the right geography, often "in region", then sovereignty obligations are met. In reality, most modern data architectures no longer behave in ways that make this assumption true.

Data residency without meaningful control

Data can be physically resident in a particular country or region while:

- Identity, access control, and policy enforcement are managed from another region.
- Operations and support staff access systems from different jurisdictions.
- Telemetry, logs, and backups flow to global observability or disaster recovery services.
- Upstream applications, integration pipelines, BI tools, and AI applications pull data into other regions for processing and consumption.
- Shared platform services, including catalog and governance tooling, reach into regional systems.

From a sovereignty perspective, where the data resides is only one part of the story. Equally important are:

- Who controls the systems that manage and process the data.
- Where those systems run and which laws apply to them.
- How access is granted, limited, monitored, and where possible, revoked.
- How consistently policies are enforced across technologies.

Revocation is particularly challenging in practice. Once data has been legitimately accessed, no architecture can fully prevent private copies from being created. Sovereignty by design therefore focuses on limiting unnecessary access, constraining what can be accessed in the first place, and maintaining traceability for governed copies, rather than assuming perfect reversibility.

Location is necessary context, but it is not a sufficient safeguard.

Analytics and AI blur jurisdictional lines

Even without making AI the star of this paper, it is impossible to ignore its effect on data use. Modern analytics and AI workloads:

- Combine data from multiple sources and regions into shared pipelines.
- Use semantic and vector search to find relevant information, not just rows in a table.
- Surface insights through dashboards, interactive analysis, and AI-driven interfaces.

In such environments, a user in one jurisdiction may gain insight derived from data in another, even if the underlying records never moved. Derivative data, training data, aggregates, and embeddings all become part of the sovereignty equation.

A location-only view simply cannot capture this complexity.

Redefining Data Sovereignty: Control, Enforcement, and Proof

To make sovereignty actionable, organizations need a definition that maps directly onto architectural and operational decisions. A practical definition of data sovereignty can be framed around three capabilities.

Control

Who decides how data can be used, by whom, from where, and for what purposes?

Control includes:

- Clear ownership and stewardship for datasets and domains.
- Explicit data classifications and usage categories.
- Jurisdictional and contractual constraints tied to specific datasets or customers.
- Access rights and boundaries for AI models and assistants, including where they may run and which data they may use for inference.
- Business-level decisions about what is acceptable use, not just technical permissions.

Without clear control, every new use case becomes a negotiation.

Enforcement

How are sovereignty rules actually applied?

Enforcement mechanisms include:

- Access control policies that factor in user identity, role, purpose, and location.
- Network and infrastructure boundaries which reflect jurisdictional considerations.
- Data masking, tokenization, and minimization applied by default where appropriate.
- Query routing and processing locality that align with residency and control requirements.
- Consistent use of shared identity and policy services so that heterogeneous systems, such as operational databases, data warehouses, pipeline tools, BI tools, and AI applications, apply the same rules to enforce consistent governance across services and avoid out-of-sync situations.

In practice, few enterprises run on a single platform. Enforcement therefore has to be native within each system and also coordinated across systems through shared identity, metadata, and policy. It must also respect transitive effects. For example, the fact that a user can run a pipeline or open a report does not automatically mean they should see every underlying data source. Those dependencies have to be modeled and enforced deliberately. If sovereignty rules exist only in documents or in isolated administration consoles, rather than in the systems and services that actually grant and evaluate access, sovereignty remains aspirational.

Proof

How do you demonstrate that you are doing what you say?

Proof encompasses:

- Lineage from source systems through transformations, data products, and downstream use, including where data was stored and processed at each step.
- Audit trails showing who accessed what, when, from where, and under which policy.
- Evidence that specific regulations, contractual terms, and internal policies were honored.
- The ability to respond quickly and credibly to regulator, customer, or internal inquiries.

Proof is what converts sovereignty from a belief into a defensible position.

Architecture as the Sovereignty Enforcement Layer

Sovereignty lives or dies in the architecture. Policies define intent. Platforms and integration patterns determine what is actually possible.

From monolithic platforms to federated data architectures

Historically, enterprises tried to centralize data into a single warehouse, lake, or platform. That approach is under pressure for three reasons:

- Regulatory constraints make it difficult to bring all data into a single environment.
- Organizational reality means data is created and owned in distributed domains.
- Performance and agility suffer when everything must route through a central bottleneck.

A sovereignty-aware architecture instead favors federation with coordination:

- Domains and regions own and manage their data and local platforms.
- A central layer provides shared semantics, governance policies, identity and access control services, and discovery so that heterogeneous systems apply the same rules.
- Global use cases are implemented by orchestrating data products and services, not by centralizing raw data.

This model dovetails naturally with data mesh, data products, and modern data intelligence platforms.

Separating data, metadata, and compute

Sovereignty requirements change faster than physical infrastructure can. New regulations, markets, and customer expectations emerge regularly.

Organizations can adapt to new constraints without constant refactoring by cleanly separating:

- Data, where it is stored and processed.
- Metadata, what it is, where it came from, what rules apply.
- Compute, which workloads run where and against what.

Sovereignty rules should live primarily in metadata and be enforced by compute and access layers. That way, you can:

- Move workloads instead of data where appropriate.
- Adjust which datasets participate in specific use cases.
- Reconfigure boundaries as jurisdictions or customer requirements evolve.

Hybrid and edge sovereignty

Many of the most sensitive sovereignty scenarios involve hybrid and edge deployments:

- National infrastructure operating under strict in-country requirements.
- Industrial and IoT systems generating data that must remain on premises.
- Highly regulated workloads that cannot be moved into generic public cloud services.

An architecture that assumes "everything in one cloud region" will fail in these cases. Sovereignty-aware design assumes heterogeneity from the start and treats:

- On-premises environments,
- Private clouds, including self-hosted Kubernetes and similar platforms,
- Sovereign or sector-specific clouds, and
- Edge locations

As first-class locations for storage and processing, It is also important not to assume that a "sovereign cloud" label automatically solves the sovereignty problem. If ultimate control of the infrastructure, control plane, or keys remains with a provider that is subject to another jurisdiction, there is still a dependency and a residual risk. True sovereignty-by-design looks beyond where the hardware sits and asks who can operate it, who can switch it off, and which laws apply when difficult questions are asked.

Governing Sovereignty by Design

Governance is where many sovereignty strategies falter. If governance relies primarily on committees, documents, and manual checks, it will not keep up with the volume and velocity of modern data use.

Metadata as the backbone of sovereignty

For sovereignty to be enforceable, data must carry its rules with it. That means investing in rich, connected metadata that captures:

- Data classifications, such as sensitivity levels, PII flags, or sector-specific categories.
- Jurisdiction and residency requirements.
- Contractual and customer-specific obligations.
- Usage constraints, for example internal analytics only, no sharing with partners, or no export.
- Retention, deletion, and minimization expectations.

This metadata must be:

- Consistent across platforms and tools.
- Available at query and pipeline runtime, not just in catalogs.
- Governed and auditable in its own right.

Data contracts with sovereignty clauses

Data contracts formalize expectations about data between producers and consumers. To support sovereignty, contracts should extend beyond schema and SLAs to include:

- Allowed use cases, such as analytics, reporting, or modeling.
- Disallowed behaviors, such as export outside certain regions, combination with specific datasets, or use in certain external services.
- Jurisdictional scope and residency commitments.
- Dependencies on other datasets with their own constraints.

When data contracts are treated as part of platform configuration, rather than purely as documentation, they become powerful enforcement tools. They also make sovereignty rules understandable and actionable for non-IT users, so that self-service analytics and AI do not accidentally bypass obligations that were only ever visible to technical teams.

Governance-by-design

Rather than positioning governance as a gate at the end of a process, sovereignty-aware organizations embed governance into:

- Data contracts/data product templates and provisioning processes.
- Pipeline orchestration and CI/CD workflows.
- Access management and self-service data tools.

The goal is to make the compliant path the default path, with:

- Automatic policy evaluation and enforcement for common operations.
- Clear, exception-based processes for genuinely unusual scenarios.
- Continuous feedback loops from observability and audit findings back into design.

In reality, no organization starts from a clean slate. Governance by design usually begins with forward engineering new data products and pipelines with contracts and policies from day one, while incrementally wrapping and upgrading existing assets. That can mean prioritizing high risk domains, inferring initial contracts from lineage and usage, and using patterns such as gateways or shared services to bring older systems under the same controls. The aim is not a big bang rewrite, but a steady shift so that over time more of the estate behaves as if it had been designed for sovereignty from the start.

Governance-by-design makes sovereignty compatible with speed, instead of in tension with it.

The Role of Vector Databases and Local AI

Even in a paper primarily about data sovereignty, it is important to acknowledge where the landscape is moving. Semantic search, vector databases, and Local AI are becoming standard ways for users and systems to access data. They can either weaken sovereignty or strengthen it.

Vector databases as a new access layer

Vector databases and semantic layers index the meaning of content, not just its structure. They power capabilities such as:

- Natural-language search over documents and records.
- Retrieval-augmented generation, where AI systems ground responses in enterprise data.
- Recommendation and similarity-based discovery.

In practice, their primary use in enterprises is to augment AI applications with context drawn from proprietary, customer specific, or even PII-heavy data. That combination of powerful retrieval and highly sensitive content makes the vector layer one of the most critical components for enforcing sovereign storage and access.

From a sovereignty perspective, this matters because whatever is embedded and indexed becomes part of the effective access surface for data.

If the vector layer is:

- Global and undifferentiated,
- Blind to jurisdiction and classification, and
- Loosely governed,

then sovereignty rules can be bypassed in practice, even if they are respected at the raw data layer.

Governing semantic and vector layers

To align with data sovereignty principles, vector databases and semantic layers should:

- Store and expose metadata and policy alongside embeddings, for example region, owner, classification, or allowed jurisdictions.
- Enforce query-time filters so retrieval respects residency and usage constraints.
- Integrate with existing access control and governance systems, rather than operating separately.
- Emit lineage and audit events indicating which items were retrieved and under which policies.

In other words, semantic and vector layers must be treated as governed data products, not sidecars.

Local AI as an access pattern, not a separate problem

Local AI, meaning AI workloads that run in the same jurisdiction and sphere of control as the governed data they use, whether on premises, in a private cloud, or in a clearly scoped regional cloud, should be viewed as a deployment pattern built on sovereign data foundations, not an independent domain.

It is useful to think about two distinct components of an AI application:

- The workflows and pipelines that orchestrate data preparation, retrieval, prompting, and post processing.
- The model endpoints that are invoked over API calls from those workflows.

Both components need to respect sovereignty requirements.

For workflows and pipelines, the earlier architectural principles already apply. Orchestration must run in environments that align with residency and control obligations. It must use governed data products, semantic and vector layers that are deployed into appropriate regions, and access controls that reflect jurisdiction and contractual constraints.

For model endpoints, sovereignty raises additional questions. True local control means that at least some models need to be hosted in environments that are under the same jurisdiction and operational control as the data they are using.

In practice this is challenging. Large language models require significant compute and memory, and cloud-based LLM services provide more than a base model. They bundle system prompting, guardrailing, monitoring, autoscaling, load balancing, memory, and other orchestration capabilities that are difficult to recreate from scratch in a purely local environment.

As a result, many organizations will use a mix of patterns, for example:

- Hosting smaller or domain specific models locally for highly sensitive use cases and data.
- Using regional deployments of larger models for less sensitive scenarios, combined with strict controls on what data is sent.
- Routing different classes of prompts and retrieval results to different models, based on jurisdiction and sensitivity.

Open-source models, when self-hosted and integrated into a carefully controlled environment, can reduce the risk that conversations or retrieved context are sent to external providers. They are not a guarantee on their own, but they do make it more feasible to build AI services where both the data and the model execution remain under a single, clearly defined sphere of control.

Typical Local AI patterns in a sovereignty aware design include:

- Deploying semantic and vector services, and the AI workflows that use them, into specific regions or environments that align with sovereignty requirements.
- Exposing consistent APIs and user experiences that internally route to different local backends and model endpoints, depending on jurisdiction, customer, and data classification.
- Ensuring that any summarization, transformation, or generation performed by AI is subject to the same constraints and logging as direct analytical access to the underlying data.

When Local AI, model hosting, and vector databases are designed in this way, they reinforce data sovereignty by ensuring that new AI access methods remain bound by the same rules as traditional access methods, rather than creating a separate, less governed path into sensitive data.

From Compliance Burden to Strategic Advantage

Organizations that treat data sovereignty as a static checklist will continue to experience it as friction:

- Projects slowed or stopped late in their lifecycle.
- Complex exception processes that incentivize workaround behavior.
- Fragmented architectures created to satisfy narrow interpretations of rules.

By contrast, organizations that treat sovereignty as a core architectural property gain several advantages:

- Speed with safety. Teams can move faster because guardrails are built into platforms, not negotiated case by case.
- AI readiness without the risk. Enterprises can enable AI adoption, including semantic search and assistants, without creating new compliance gaps or regulatory exposure, because AI access is bound by the same sovereignty rules as any other access.
- Resilience to change. New regulations, markets, and customer requirements can be absorbed by adjusting policies and deployment patterns rather than rebuilding systems.
- Stronger trust. Customers, partners, and regulators gain confidence from credible, evidence-backed answers to "Where is my data, who can see it, and under what conditions?"
- Strategic flexibility. Enterprises can choose where to run workloads and store data based on business and performance needs, knowing that sovereignty rules will follow.

Sovereignty is no longer just about restricting what cannot be done with data. Done well, it is how organizations expand what they can do, safely, at scale, and with confidence.

About The Author

Emma McGrattan
CTO, Actian

Emma McGrattan is the Chief Technology Officer at Actian, where she leads the strategy behind some of the industry's most innovative data solutions. With over two decades of experience delivering mission-critical data technologies, she has helped organizations across industries design and modernize their data architectures, embed governance by design, and become AI-ready.

About Actian

Actian empowers enterprises to confidently manage and govern data at scale. Organizations trust Actian data management and data intelligence solutions to streamline complex data environments and accelerate the delivery of AI-ready data. Designed to be flexible, Actian solutions integrate seamlessly and perform reliably across on-premises, cloud and hybrid environments. Learn more about Actian, The data and AI division of **HCLSoftware**, at **actian.com**.

Scan to read the white paper online.

ACTIAN™
a division of HCLSoftware

Appendix A: Data Sovereignty Checklist

Use this checklist as a practical assessment tool.

Scoring (optional):

- **Yes** = in place and enforced
- **Partial** = inconsistent or manual
- **No** = absent or policy-only

Definition and Ownership

□ We have a shared, enterprise-wide definition of data sovereignty beyond residency.

□ Control, enforcement, and proof are explicitly defined for data usage.

□ Ownership for sovereignty decisions spans data, platform, security, and legal.

Architectural Foundations

□ Our data architecture supports federation, not just centralization.

□ We can process data locally and share insights globally in compliant ways.

□ Hybrid, on-prem, and sovereign environments are treated as first-class locations.

Metadata and Policy Enforcement

□ Metadata captures classification, jurisdiction, and usage constraints.

□ Policies are machine-readable and enforced in platforms, not only documented.

□ Policies apply consistently across structured, unstructured, and semantic or vector layers.

Data Contracts and Governance-by-Design

□ Data contracts include sovereignty clauses such as allowed use cases, jurisdictions, and limits.

□ Governance is embedded into data product and pipeline design, not added afterwards.

□ Policy inheritance allows regional or local adaptation without breaking global standards.

Access Control and Key Management

□ Access can be restricted by region, purpose, and workload type.

□ Encryption keys and key management align with sovereignty requirements.

□ Third-party and operational access, including support, monitoring, and SaaS integrations, is explicitly governed..

Auditability, Lineage, and Proof

□ We can trace data from sources through transformations into products and consumption.

□ Access and usage logs are complete, retained appropriately, and queryable.

□ We can respond rapidly and confidently to regulator or customer questions.

Semantic, Vector, and Local AI Access

□ Vector databases and semantic layers are governed with the same rigor as primary stores.

□ Retrieval respects residency and usage constraints via policy-aware filters.

□ Local AI deployments, where they exist, operate within the same sovereignty framework.

If many answers are **"Partial"** or **"No"**, sovereignty risk will surface first as friction, with projects delayed, customers asking harder questions, and architecture becoming more fragmented.

ACTIAN™
a division of HCLSoftware

Tel +1 512 231 6000 Fax +1.512.231.6010 710 Hesters Crossing Road, Suite 250, Round Rock, TX 78681 **Actian.com**