SOVEREIGN DIGITAL RESILIENCE · PAPER 2 OF 3

Governing Agentic AI

In Systems of Public Record — Where Governance Is the Accelerant

SANJEET KUMAR

Enterprise Architecture & Technology Governance Leader

Critical Infrastructure · Sovereign Digital Resilience · AI Governance

sanjeetkumar.com | linkedin.com/in/sanjeetkumar | July 2026

Governing Agentic AI in Systems of Public Record

Sanjeet Kumar — Enterprise Architecture & Technology Governance Leader *Critical Infrastructure · Sovereign Digital Resilience · AI Governance — July 2026. Views are the author's own.*

The question I'm asked most often about agentic AI is "what can it do?" In systems of public record — registries, core banking, government records — that's the wrong first question. The right one is: **what happens when it's wrong, and who is accountable?** Answer that architecturally, and aggressive AI adoption becomes safe. Skip it, and every impressive demo is a liability being amortized.

I had the opportunity to initiate and shape an agentic-AI modernization MVP inside exactly such an environment — an initiative that ultimately delivered acknowledged multi-million-dollar savings and cut MVP cycles from roughly six months to six weeks. The interesting part isn't the numbers; it's that the acceleration came *because of* the governance, not despite it. Four principles made it work.

1. Draw the authority boundary before the capability boundary. The first architectural artefact wasn't a model selection — it was a decision map: which actions the agent may execute autonomously, which require human confirmation, and which are categorically reserved for humans. In a system of public record, the authoritative *write* is the crown jewel; our guardrail was simple and absolute — agents accelerate everything up to the point of record, and an accountable human owns the point of record. Autonomy is granted per action class, not per system.

2. Treat the agent as an untrusted, talented junior. Agents are given bounded workspaces, least-privilege credentials, and outputs that are *proposals* until validated. "Hallucinated authorization" — an agent convincing itself (or a downstream system) that it holds permissions it doesn't — is a failure mode you prevent structurally, at the permission layer, not behaviourally, in the prompt. Prompts are advisory; credentials are architecture.

3. Extend the architecture repository to cover agents. Traditional EA artefacts describe systems, interfaces, and data flows. Agentic systems require the repository to also carry: agent permission matrices, human-in-the-loop checkpoints, model and prompt lineage, and decision provenance — who or *what* recommended each action, on what context. This is TOGAF-style discipline extended to a new class of actor, and it's what turns "we use AI" into an auditable statement.

4. Sovereignty applies to the whole inference chain. Prompts, retrieval context, embeddings, and outputs inherit the sensitivity of the records they touch. If a citizen's record flows through an inference endpoint, that endpoint sits inside the sovereignty boundary — subject to the same residency, jurisdictional-control, and audit expectations as the database it came from. Regulatory regimes (PDPL in the UAE, Canadian privacy law, sectoral outsourcing rules) are converging on this position faster than most architectures are.

The pattern that emerges: **governance is the accelerant.** Teams move fastest when the safe zone is explicit — when engineers know precisely where agents may act, experimentation stops being negotiated case-by-case and starts compounding. The organizations that will win the agentic era are not those with the fewest rules, but those whose rules are architectural: enforced by design, auditable by default, and calibrated to what the system can afford to get wrong.

In systems societies can't afford to lose, that calibration *is* the job.