Machine Translated by Google
Revista Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Evidence-driven governance and governance drift in cloud-native infrastructures

Evidence-oriented governance and governance drift in cloud-native infrastructures

Evidence-driven governance and governance drift in native cloud infrastructures

Obed Bessa Rocha da Silva

Independent specialist in cloud solution architecture, DevOps, and digital infrastructure governance.

SUMMARY

This article proposes a conceptual model of evidence-driven governance to analyze the divergence between approved institutional intent and operational state in cloud-native infrastructures. A structured, non-exhaustive narrative review of norms, specifications, peer-reviewed research, and technical reports was conducted, organized into five categories: representation of intent, automation of change, identity and provenance, state observation, and deviation handling. The synthesis delimits governance drift in relation to configuration deviations and isolated policy violations, describes a six-step cycle linking intent, declared state, delivery, observed state, materiality classification, and response, and presents a taxonomy of deviations and a model of...

Maturity in five levels. It is argued that persistent governance depends on verifiable evidence, stable identities, semantic comparison, and responses proportionate to risk. The work is conceptual: it does not validate scale, universal thresholds, or organizational effectiveness. Its contribution lies in making explicit the relationships and criteria necessary for future operationalization and empirical evaluation of governance drift.

Keywords: digital governance; governance drift; cloud-native; DevSecOps; provenance; configuration management.

ABSTRACT

This article proposes an evidence-oriented governance model for analyzing divergence between approved institutional intent and operational state in cloud-native infrastructures. A structured, non-exhaustive narrative review of standards, specifications, peer-reviewed research, and technical reports was organized into five categories: intent representation, change automation, identity and provenance, state observation, and deviation treatment. The synthesis distinguishes governance drift from configuration drift and isolated policy violations, describes a six-stage cycle linking intent, declared state, delivery, observed state, materiality classification, and response, and presents a deviation taxonomy and a five-level maturity model. Persistent governance is argued to depend on queryable evidence, stable identities, semantic comparison, and risk-proportionate responses. The work is conceptual: it validates neither a scale, universal thresholds, nor organizational effectiveness. Its contribution is to make explicit the relationships and criteria required for future operationalization and empirical evaluation of governance drift.

Keywords: digital governance; governance drift; cloud-native; DevSecOps; provenance; configuration management.

1 INTRODUCTION

Cloud-native architectures replace relatively stable sets of servers with

ecosystems composed of microservices, containers, controllers, integration pipelines and

Continuous delivery and declarative interfaces. This change increases speed and elasticity, but...

It also expands on the points at which an authorized decision may fail to correspond to the state.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

effectively implemented. The governance challenge is no longer just about approving a change: it's about getting it done.
to reconstruct who authorized it, what artifacts were produced, how they came to be produced, what
The result was determined by whether the divergence remained within the accepted risk.
Traditional configuration management provides baseline principles, change control, and...
Monitoring, while Infrastructure as Code and GitOps are part of the versionable intent.
and reconcilable (NIST, 2019; RAHMAN; MAHDAVI-HEZAVEH; WILLIAMS, 2019; BEETZ;
(HARRER, 2022). These capabilities, however, do not guarantee that the declared configuration has
institutional authority, that the supply chain remains intact or that the observed state
Maintain a verifiable link with approvals and exceptions. The literature on provenance and safety.
The software supply chain partially fills this gap, but it does not replace the governance decision regarding...
materiality, legitimacy and response (W3C, 2013; TORRES-ARIAS et al., 2019; SLSA
COMMUNITY, 2025).
In this article, governance drift refers to the material divergence between authorized institutional intent.
and observable operational reality. The intention may be expressed in policies, decisions
Architectural aspects, declared configurations, regulatory requirements, and approved exceptions. The reality.
Operational includes resources in operation, identities, relationships, permissions, and evidence produced.
during delivery and operation. The divergence only acquires governance significance when it is interpreted
with regard to authority, risk, scope, duration and responsibility; isolated technical difference not
That's enough.

The study answers three questions: RQ1 — how to distinguish governance drift from technical deviations
Adjacent? RQ2 — What evidence links approved intent, delivery, and observed state? RQ3 —
How to classify and handle discrepancies without confusing automation with absolute control?
The contributions are: a well-defined construct; a six-step analytical cycle; and a taxonomy of...
deviations and responses; maturity model driven by verifiable capabilities; and agenda of
empirical validation.
The practical relevance stems from the scale achieved by the cloud-native substrate. In the annual survey of
In the 2023 CNCF survey, 66% of potential or current consumer respondents stated that they use Kubernetes.
in production and 18% said they were evaluating it (CNCF, 2024). These percentages do not demonstrate
Governance maturity does not universally represent all organizations, but it does indicate that
Automated decisions and ephemeral states have become an ordinary part of the digital infrastructure.

2 METHODOLOGY

A structured, non-exhaustive narrative review was conducted, appropriate for the purpose of integrating concepts.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

stemming from software engineering, security, governance, and operations, without estimating effects. aggregates. The quality of the review was guided by the six domains of SANRA: relevance, objective, Description of the search, referencing, scientific reasoning, and presentation of relevant data. (BAETHGE; GOLDBECK-WOOD; MERTENS, 2019). SANRA was employed as critical instrument, not as a systematic review protocol.

The targeted search was completed on August 25, 2026, in IEEE Xplore, ACM Digital Library. ScienceDirect and USENIX, in addition to the official repositories of NIST, ISO, W3C, CNCF, DORA, and SLSA. The terms governance drift, configuration drift, and policy as were combined in Portuguese and English. code, infrastructure as code, GitOps, cloud-native governance, continuous monitoring, software provenance, SLSA, DevSecOps, and software supply chain. The search covered publications up to the date of cut, without a rigid lower limit, because foundational norms of provenance and monitoring They remain technically valid.

Documents with traceable authorship and origin that contributed directly to were included. at least one of the five analytical categories: representation of intent; automation of change; Identity and provenance; observation of the state; and handling of deviation. Norms and standards were prioritized. Primary specifications, peer-reviewed research, and reports with described methodology. They were Excluding promotional materials, duplicates, texts without identifiable authorship, and sources that only They mentioned tools without discussing mechanisms, evidence, or risks. The final corpus compiled 21 documents.

The synthesis was carried out through conceptual comparison. For each source, control objects were recorded. presumed authority, observed artifact, offered guarantee, residual risk, and anticipated response. In Next, the mechanisms were organized into the proposed cycle and subjected to three consistency tests: distinction between availability of evidence and semantic correctness; separation between authorization and configuration; and preservation of human responsibility. There was no evaluation of the organization. experiment, interview or incident measurement.

As methodological limitations, the search does not aim for exhaustive bibliometric coverage, nor did it perform... Double independent screening combines scientific literature with standards and technical evidence. institutional. Consequently, the synthesis supports conceptual plausibility and traceability of propositions, but not prevalence, causality, universal thresholds, or model effectiveness in production.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

3. CONCEPTUAL FOUNDATION AND DELIMITATION

3.1 Governance as a socio-technical capability

Digital governance defines decision-making rights, limits of autonomy, risk acceptance criteria, and...
forms of accountability. In cloud-native environments, these elements cross over.
Repositories, service catalogs, pipelines, registries, clusters, and observability platforms.
NIST SP 800-53 Rev. 5 and ISO/IEC 27001:2022 place controls within processes.
Organizational risk management strategies, avoiding reducing them to a checklist applied after implementation.
(NIST, 2020; ISO, 2022a).
The socio-technical nature stems from the interdependence between code, infrastructure, and coordination.
human. A technically correct policy can fail without an owner, remediation path, or
Documented exception; a well-designed process can be undermined by excessive credentials.
Manual changes and competing sources of truth. DevOps studies link performance
sustainable through the combination of technical, cultural and learning capabilities, and research of
Infrastructure as code shows that adoption does not eliminate the challenges of testing, maintenance, and support.
(FORSGREN; HUMBLE; KIM, 2018; GUERRIERO et al., 2019).
Declarative controllers explicitly state a desired state and execute reconciliation loops.
However, distributed systems allow for transient divergences, partial failures, and interventions.
emergencies. The experience gained in Borg, Omega, and Kubernetes demonstrates the centrality of
continuous control and convergence mechanisms, without implying that every state difference is a
Violation of governance (BURNS et al., 2016). The analysis requires authority, context, and materiality.

3.2 Governance drift and related concepts

Table 1 outlines the proposed concept. Configuration drift describes the difference between baseline or
Desired configuration and observed configuration. A policy-as-code violation occurs when a rule
The executable requirement is not satisfied. A provenance gap appears when it is not possible to demonstrate
Origin, process, or identity. Governance drift can include any of these conditions, but
This requires an additional element: a significant divergence between operational reality and intention.
authorized institution.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Table 1 – Delineation between governance drift and adjacent concepts

Concept	Object compared	central question	Typical response
Configuration drift	Baseline/desired configuration × observed configuration	Did the technical situation diverge?	Reconcile or update the baseline.
Policy violation	State or artifact × enforceable rule	Was a coded condition violated?	Block, alert, or create an exception.
Provenance gap	Artifact/state × origin, process and identity	Can the chain be verified?	Preserve, correlate, or reconstruct evidence.
Governance drift	Authorized intention vs. operational reality	Does the divergence alter risk, legitimacy, or accountability?	To classify, contain, reconcile, accept, or investigate.

Source: compiled by the author based on NIST (2019, 2022a, 2024), W3C (2013), Rahman, Mahdavi-Hezaveh and Williams (2019), Beetz and Harrer (2022) and SLSA Community (2025).

The distinction prevents two errors. The first is classifying any emergency change as a failure, even when authorized, registered, and temporary. The second is to presume conformity because a The configuration matches the repository, although the repository does not represent the current decision or the The delivery chain cannot be verified. Governance drift is therefore a relationship between states and authority, not an isolated attribute of a file.

4. Evidence-Driven Governance Model

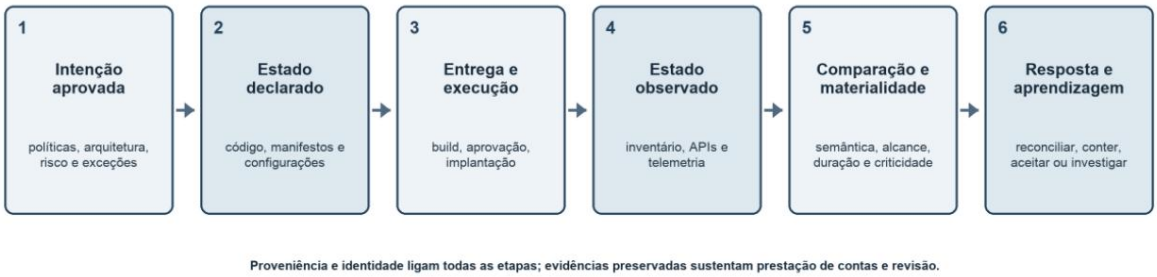
The model organizes persistent governance into six interconnected steps. Each step produces or consumes. evidence, but no single source proves the overall accuracy. Verifiability depends on stable identity, temporal semantics, and the ability to relate decisions, artifacts, activities, and states.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Figure 1 – Evidence-driven governance cycle



Source: prepared by the author.

4.1 Intention approved and status declared

The approved intent covers policies, architectural decisions, risk criteria, and requirements.
regulatory and exception procedures. The first requirement is to declare which artifact has authority for each.
The decision and for how long. ISO/IEC 27001:2022 requires that the safety management system be
appropriate to the context and risks of the organization, while the NIST SSDF recommends integrating
Safe development practices throughout the life cycle (ISO, 2022a; NIST, 2022b).
The declared state translates part of this intention into infrastructure code, manifestos, catalogs, and
Executable rules. Infrastructure as code improves versioning and repeatability, but studies
They point to limitations in testing, maintenance, and tool support (RAHMAN; MAHDAVI-HEZAVEH;
WILLIAMS, 2019; GUERRIERO et al., 2019). GitOps adds reconciliation based on
repositories, without single-handedly resolving authority, conflicts between sources, or the legitimacy of exceptions.
(BEETZ; HARRER, 2022).

4.2 Delivery, execution and observed status

Between declaration and execution there are builds, verifications, signatures, approvals, mutations of
Admission and controller actions. Partial failure, unavailable dependency, or manual intervention.
This can produce a different result than planned. Therefore, comparing only the repository and configuration is not recommended.
The generated data is insufficient: the observed state must include inventory, control plane APIs,
telemetry, runtime identities and relationships.
Policy as code allows for anticipating controls, but codified rules remain software: they can
Contain errors, become outdated, and produce systemic effects. NIST SP 800-204C describes policy
Observability as code and observability as code as components of cloud-native DevSecOps, and NIST SP 800-
204D integrates software chain controls into pipelines (NIST, 2022a, 2024). Governance should

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Test the rules, version them, monitor their impact, and preserve exception procedures.

4.3 Provenance as a verifiable link

Provenance describes the entities, activities, and agents involved in the production of an artifact or...
state. The W3C PROV model provides an interoperable vocabulary for representing these relationships.
(W3C, 2013). In the software chain, in-toto, SLSA and Sigstore demonstrate mechanisms
supplementary steps for declaration, attestation, signature linked to identities, and transparency.
(TORRES-ARIAS et al., 2019; NEWMAN; MEYERS; TORRES-ARIAS, 2022; SLSA
COMMUNITY, 2025).

Metadata, by itself, does not guarantee reliability. If the same actor can alter both the artifact and the record...
Without detection, the evidence will have limited independence. Stronger controls use separation of
functions, short-lived identities, controlled builders, verifiable signatures, and storage
with proper retention. The chain also needs to be searchable: immutable identifiers, such as
digests reduce manual correlations and allow for the location of artifacts affected by vulnerability or
incident.

Data collection must be focused on minimization and purpose. Evidence should not reveal secrets or data.
Unnecessary personnel. Evidence-driven governance is not the same as total surveillance;
It seeks the necessary elements to support decision-making, accountability, and reconstruction, with access and
retention proportional to the risk.

5. DETECTION, CLASSIFICATION AND RESPONSE

Detecting drift requires a comparison function and a materiality model. Syntactic differences can be...
irrelevant, as a minor semantic change — for example, expanding a network policy or
Changing the execution identity can substantially alter the risk. Comparators should
normalize representations, know standard values, consider relationships between resources, and record the
the moment of validity of each observation.

Continuous monitoring of NIST SP 800-137 provides applicable principles: visibility on
assets, threats, vulnerabilities, and effectiveness of controls (NIST, 2011). In ephemeral environments,
The frequency of data collection should keep pace with the rate of change. Control events, snapshots
Signed documents and regular consultations provide complementary evidence; periodic checks are slow.
They may not capture resources that existed for minutes.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Table 2 – Operational taxonomy of divergences

Class	Condition	Standard treatment	Minimum evidence
Authorized and temporary	Change approved, with deadline and owner.	Monitor and reconcile at maturity.	Approval, scope, validity and status
Operational benign	Expected variation without material change in risk.	Register or normalize comparison	Rule of tolerance and observation
Control drift	Missing, outdated, or bypassed control.	Correct control and evaluate coverage.	Policy version, result, and exception.
Safety drift	Disclosure of material confidentiality, integrity, or availability.	Contain, preserve evidence, and investigate.	Timeline, identity, scope, and artifacts
Undetermined	Insufficient or conflicting evidence	Refrain from destructive correction and collect context	Gaps and stated confidence level

Source: prepared by the author.

The indeterminate class is necessary to avoid artificial certainty. Automatic correction based on Incomplete evidence can obscure useful signals, disrupt services, or broaden interpretation. mistaken. For low risks and well-tested rules, automatic reconciliation is plausible; for High risks or low confidence, preservation of context, containment, and human review should to prevent irreversible changes.

Useful indicators include time between deviation and detection, time to reconciliation, and the proportion of changes outside the approved workflow, resource coverage by policy, rate of overdue exceptions and completeness of provenance links. These indicators should support learning, not Ranking individuals. Isolated goals encourage local optimization; accurate reading considers this. Incidents, workload, developer experience, and reliability results.

6 controls incorporated into the DevSecOps cycle

Integrating governance into pipelines anticipates issues and reduces the cost of remediation. Controls can... Verify dependencies, secrets, infrastructure code, images, signatures, and policies before... Promotion, maintaining results and approvals. Continuous delivery already handles build, test, and deployment. as an automated flow; the distinguishing feature of evidence-driven governance is preserving the connection.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

between each outcome, the decision that authorized it and the state that actually arose (HUMBLE; FARLEY, 2010).

Shift left does not transfer all responsibility to the developers. Security and platform teams. They should provide understandable rules, reliable tools, and exception handling paths. Controls with Opaque messages and high false positive rates encourage bypass. The 2024 DORA report associates this with... Internal platforms lead to gains in productivity and organizational performance, but also record throughput reductions and stability in certain contexts; user experience, independence and Adoption as a product are design conditions, not cosmetic details (GOOGLE CLOUD; DORA, 2024).

Preventive controls need to be combined with detective and responsive mechanisms. Vulnerabilities emerge after deployment, dependencies change, and identities can be... Committed. Runtime, observability, and response policies cover residual risks. A Segregation of duties should be proportionate: critical services may require independent review. while low-risk changes can follow automation with clear boundaries.

The central principle is not to multiply approvals, but to prevent a single identity from altering them. Intention, artifact, evidence, and production without the possibility of detection. Signatures and logs of Transparency reduces this risk, but guaranteeing it depends on a threat model, verification, and... protection of identities (NEWMAN; MEYERS; TORRES-ARIAS, 2022).

7 Evidence-Driven Maturity Model

Maturity is treated as demonstrable capability, not as a quantity of tools. The Framework 3 presents five cumulative levels. Progression should be evaluated by services and material risks; An organization can be at different levels depending on the domain being analyzed.

Table 3 – Maturity levels and verifiable criteria

Level	Dominant capability	Expected evidence	Progression criteria
1 — Fragmented	Incomplete inventory and responsibilities	Scattered records and manual changes	Identify critical assets, responsible parties, and sources of truth.
2 — Declared	Versioned intent and change review	Repositories, baselines, and approvals	Observe the actual state and relate it to the declared version.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS – Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Level	Dominant capability	Expected evidence	Progression criteria
3 — Traceable	Provenance across review, build, artifact, and deployment	Attestations, signatures, identities and digests	Consult the supply chain under the deadline and detect gaps.
4 — Responsive	Drift classified and treated as a risk.	Prioritized alerts, exceptions with validity periods, and reconciliations.	Demonstrate timeliness, coverage, reliability, and results.
5 — Adaptive	Controls learn from incidents and metrics.	Efficacy reviews, controlled trials, and transparent changes.	Reduce recidivism without increasing friction or residual risk.

Source: prepared by the author.

Progression does not presuppose complete automation. Simple environments, with clear responsibilities and
Reliable evidence can be more manageable than extensive and disconnected data stacks.
Assessments should test whether the organization can rebuild after change, identify exposure,
to restore status and explain decisions to an independent party.
At the adaptive level, policies are versioned and tested, exceptions expire, and changes are made to controls.
They go through a transparent process. The platform offers approved paths, but allows...
Challenge and evolution. Automation collects, compares, and executes predictable responses; people define.
They value things, accept risks, and consider the consequences.

8 DISCUSSION

8.1 Theoretical and practical implications

The main theoretical implication is to separate governance from momentary conformity. A snapshot.
as it may not explain how the state was produced; a deviation may be legitimate when
authorized and temporary. The model shifts the unit of analysis from the isolated file to the chain.
between decision, artifact, activity, and observation. This brings governance, provenance, and operation closer together.
without stating that one of them replaces the others.
In practice, the organization can start with verifiable questions: which artifact has authority?
Who approved the current version? Does the deployment derive from the expected build? What is the observed state?
Is it within acceptable limits? Can the evidence be retrieved in the required timeframe? When the answer
Since the information is unknown, the gap should be recorded as uncertainty, not filled by presumption.
The model also reduces the false dilemma between speed and control. Approved paths, policies

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google
RCMOS - Multidisciplinary Scientific Journal The Knowledge.
ISSN: 2675-9128. São Paulo-SP.

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Tests and evidence produced during the workflow prevent subsequent manual reconstruction. However, Poorly designed controls increase cognitive load and encourage parallel routes. Governance needs It must be consumable, observable, and adjustable, as well as technically rigorous.

8.2 Limitations and research agenda

The work does not validate the construct, define weights, or establish thresholds of materiality. Categories may overlap, and different reviewers may disagree on the matter. Legitimacy, scope, and response. The selected literature focuses on security, engineering, and... Software and international standards; legal, sectoral, and cultural aspects require separate studies. The research agenda includes: operationalizing dimensions without concealing their components; measuring Inter-rater agreement; testing whether completeness of provenance and evidence time predict. reconstruction quality; compare syntactic and semantic detection; analyze costs and adverse effects. of automatic reconciliation; and to conduct multi-organizational studies that separate correlation from causality.

Refutation protocols are also necessary. The model loses its usefulness if the classification is not... Improve prioritization for simple technical alerts if the provenance does not reduce time or Uncertainty of investigation, whether incorporated controls merely shift work, or whether levels of maturity levels cannot be assessed with acceptable agreement. These results should be treated as evidence against the model, not as a failure of adoption.

9 CONCLUSION

Cloud-native infrastructures render governance based solely on assessments insufficient. periodicals. The relevant divergence is not any textual difference, but the material separation between authorized intent and operational reality, interpreted according to risk, legitimacy, duration and Responsibility. Governance drift offers an analytical category for examining this drift. without presuming that all disagreement constitutes a flaw.

The synthesis showed that versioned intention, observed state, searchable provenance, comparison Semantics and proportional response form complementary capabilities. Policy as code and pipelines. DevSecOps enhances prevention, but does not eliminate residual risks nor replace judgment. The desired maturity is the ability to explain, verify, and correct changes with evidence. sufficient and explicit uncertainty.

The proposed model remains conceptual. Its contribution is to provide boundaries, stages, classes, and...

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

criteria that can be empirically challenged. The next scientific step is to test whether these capabilities improve reconstruction, prioritization, and recovery across different organizations, without Confusing a higher volume of records with better governance.

STATEMENTS

Funding: There was no external funding for this study.

Conflicts of interest: the author declares no conflicts of interest.

Data availability: the study did not produce a dataset; the sources analyzed are identified in the references.

Use of generative artificial intelligence: OpenAI's generative AI tool was used as

Editorial, linguistic, and structural support. The author reviewed the content, verified the references, and assumes responsibility.

Full responsibility for the submitted version.

REFERENCES

BAETHGE, Christopher; GOLDBECK-WOOD, Sandra; MERTENS, Stephan. SANRA—a scale for the quality assessment of narrative review articles. Research Integrity and Peer Review, vol. 4, art. 5, 2019. DOI: 10.1186/s41073-019-0064-8.

BEETZ, Florian; HARRER, Simon. GitOps: The Evolution of DevOps? IEEESoftware, v. 39, n. 4, p. 70–75, 2022. DOI: 10.1109/MS.2021.3119106.

BURNS, Brendan; GRANT, Brian; OPPENHEIMER, David; BREWER, Eric; WILKES, John. Borg, Omega, and Kubernetes. Communications of the ACM, vol. 59, n. 5, p. 50–57, 2016. DOI: 10.1145/2890784.

CNCF — Cloud Native Computing Foundation. CNCF Annual Survey 2023. San Francisco: CNCF, 2024. Available at: https://www.cncf.io/reports/cncf-annual-survey-2023/.
Accessed on: August 25, 2026.

FORSGREN, Nicole; HUMBLE, Jez; Kim, Gene. Accelerate: The Science of Lean Software and DevOps. Portland: IT Revolution, 2018.

GOOGLE CLOUD; DORA. Accelerate State of DevOps 2024. Mountain View: Google Cloud, 2024. Available at: https://dora.dev/research/2024/dora-report/. Accessed on: August 25, 2026.

GUERRIERO, Michele; GARRIGA, Martin; TAMBURRI, Damian A.; PALOMBA, Fabio. Adoption, Support, and Challenges of Infrastructure-as-Code: Insights from Industry. In: IEEE INTERNATIONAL CONFERENCE ON SOFTWARE MAINTENANCE AND EVOLUTION, 2019. Proceedings [...]. IEEE, 2019. p. 580–589. DOI: 10.1109/ICSME.2019.00092.

HUMBLE, Jez; FARLEY, David. Continuous Delivery: Reliable Software Releases through Build,

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

Test, and Deployment Automation. Boston: Addison-Wesley, 2010.

ISO — INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC 27001:2022: Information security, cybersecurity and privacy protection — Information security management systems — Requirements. Geneva: ISO, 2022a.

ISO — INTERNATIONAL ORGANIZATION FOR STANDARDIZATION. ISO/IEC 27002:2022: Information security, cybersecurity and privacy protection — Information security controls. Geneva: ISO, 2022b.

NEWMAN, Zachary; MEYERS, John Speed; TORRES-ARIAS, Santiago. Sigstore: Signing Software for Everybody. In: ACM SIGSAC CONFERENCE ON COMPUTER AND COMMUNICATIONS SECURITY, 2022. Proceedings [...]. New York: ACM, 2022. p. 2353–2367.
DOI: 10.1145/3548606.3560596.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-137: Information Security Continuous Monitoring (ISCM) for Federal Information Systems and Organizations. Gaithersburg: NIST, 2011. DOI: 10.6028/NIST.SP.800-137.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-128: Guide for Security-Focused Configuration Management of Information Systems. Update 1.
Gaithersburg: NIST, 2019. DOI: 10.6028/NIST.SP.800-128.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-53 Rev. 5: Security and Privacy Controls for Information Systems and Organizations. Gaithersburg: NIST, 2020.
DOI: 10.6028/NIST.SP.800-53r5.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-204C: Implementation of DevSecOps for a Microservices-based Application with Service Mesh.
Gaithersburg: NIST, 2022a. DOI: 10.6028/NIST.SP.800-204C.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-218: Secure Software Development Framework (SSDF) Version 1.1. Gaithersburg: NIST, 2022b. DOI: 10.6028/NIST.SP.800-218.

NIST — NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY. SP 800-204D: Strategies for the Integration of Software Supply Chain Security in DevSecOps CI/CD Pipelines.
Gaithersburg: NIST, 2024. DOI: 10.6028/NIST.SP.800-204D.

RAHMAN, Akond; MAHDAVI-HEZAVEH, Rezvan; WILLIAMS, Laurie. A systematic mapping study of infrastructure as code research. Information and Software Technology, vol. 108, p. 65–77, 2019. DOI: 10.1016/j.infsof.2018.12.004.

SLSA COMMUNITY. Supply-chain Levels for Software Artifacts: Specification, version 1.2. 2025.
Available at: https://slsa.dev/spec/v1.2/. Accessed on: August 25, 2026.

TORRES-ARIAS, Santiago; AFZALI, Hammad; KUPPUSAMY, Trishank Karthik; CURTMOLA, Reza; CAPPOS, Justin. in-toto: Providing farm-to-table guarantees for bits and bytes. In: USENIX SECURITY SYMPOSIUM, 28., 2019. Proceedings [...]. Berkeley: USENIX Association, 2019. p. 1393–1410.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Machine Translated by Google

Year VII, v.2 2026 | Submission: 01/15/2026 | Accepted: 01/20/2026 | Publication: 01/25/2026

W3C — WORLD WIDE WEB CONSORTIUM. PROV-O: The PROV Ontology. W3C Recommendation, April 30, 2013. Available at: https://www.w3.org/TR/prov-o/. Accessed on: August 25, 2026.

This article is published in open access under the Creative Commons Attribution license, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.