Nanotechnology Perceptions
ISSN 1660-6795
www.nano-*ntp.com*

Policy-Over-Model Guardrails — An Agentic Mlops Control Plane For Safe Autonomy In Production Engineering And Infra

Prashant Kumar Prasad

Vice President.

The paper is a qualitative study of the control plane development of policy-over-model which is safe and reliable agentic MLOps on the engineering as well as infrastructure setting. The paper includes a literature review of the available agentic systems, MLOps, governance and operational safety to determine critical areas of gaps and integration requirements. It has been found that agentic AI needs to have stronger policy controls, more articulate oversight roles, continuous monitoring, and transparent audit trails. The suggested framework unites these factors regarding a Model Custodian Agent with the assistance of evaluation, drift, and audit agents. The article offers a viable premise of safer autonomous activities and system development in the future.

KEYWORDS: Infrasturcture, MLOps, Autonomy, Engineering, Safety, Policy, Production.

I. INTRODUCTION

The agentic AI systems are gaining significance in engineering, operations, and the infrastructure in the community. The existing systems are not well organized with safety systems and policy regulation as well as open monitoring. Modern trends in automation require organisations to have more visible means of controlling risks, tracing the behaviour of systems, and becoming compliant. The paper will seek to bridge this gap by examining how this could be achieved through policy rule, operational tools, and agentic architectures to provide a less risky and more controlled environment in which autonomous systems operate. The qualitative approach also employs the study to determine the elements of practices that are missing, compares practice across domains, and creates a single plane of control on policy-over-model of the safe agentic MLOps.

II. RELATED WORKS

Agentic AI and Architectural Patterns

Recent research indicates that agentic AI represents an important change in direction compared to traditional machine learning and generative AI, towards situations where one has autonomous, goal-representative, tool-using systems capable of reasoning, planning, and acting with very little human supervision. One line of thought traces back to note that reference suggests that agentic systems, particularly those driven by large language models (LLMs) need

novel architectural concepts since the current software and MLOps systems do not support continuous autonomy and dynamism in decisions.

A first impression is that the existing ecosystem does not have systems in place to structure mechanisms of safe tool orchestration, versioning and execution limits at large-scale production. To fill this gap the abstraction Control Plane as a Tool was introduced where agents are allowed to communicate with a unified interface and complex tool, environment and workflow decisions made under the covers [1].

This notion is the most essential to our paper, as it demonstrates that model-level capabilities cannot guarantee safe autonomy, and the system must possess an extended supervisory layer that deciphers policies and guides model behavior. The general surveys strengthen the necessity of such formal supervision.

Such agentic AI studies as 143 studies in systematic review reveal that the design patterns of designing agents planning, ReAct, tool use, reflection, and multi-agent collaboration generate significant differences in autonomy and safety requirements [4].

The environments (static versus dynamic, deterministic versus stochastic, single versus multi-agent) are also classified in the review and prove that complexity increases when actors in production infrastructure environments (e.g., cloud systems, edge networks, shared control planes) act.

A further overall review traces overall agentic AI development in five stages, stating that nowadays agents are characterized as multi-modal reasoners, proactive decision-makers, and autonomous goal seekers [8]. All these works indicate a definite research gap as the more agents are able to act and adjust, the more their control infrastructure also has to change to avoid unsafe or unregulated actions.

Other research relates agentic AI to high-pressure contexts like cybersecurity and the critical infrastructure. As an illustration, the background on Agentic Artificial Intelligence (AAI) in cyber defense is characterized by a focus on the need to ensure that modern systems are cognitive, dynamically responsive, and quantum resistant, yet a robust system of governance and oversight and an ethical constraint due to the ambiguity of dual-use systems [7]. All these themes contribute to the idea that AI autonomy cannot be based on model predictions, but rather it needs policy-based governance planes that would monitor agent activities on a regular basis.

All these underpinnings indicate that agentic AI has ceased to be merely a model capability but a system capability. The necessity of the integrated governance and control layer can be seen as agents become more independent, be it in the field of engineering, cloud operation, or cyber defense. This preconditions policy-over-model architectures which are enforced to constrain behavior, observe behavior and assure alignment between technical, regulatory and operational domains.

Security Architectures for Agentic Systems

The focus of scaling agentic AI to production settings is the safety and governance of AI. One of the key issues in the literature, and there is no enforceable and user control over available agentic system designs. There are many theoretical methods of explaining the manner in which identity systems, approaches or models of delegation run, yet few actual, secure working environments have been integrated together.

To bridge this gap, SAGA architecture proposes an extensible security and governance architecture where all agents connect with a central provider that controls access control, lifecycle management and cryptography authorization tokens [2]. This architecture can be used to guarantee that communication between agents is strictly controlled so that the user can control it and prevent the possibility of rogue or hazardous activities. SAGA shows that the agent lifecycle should include governance: as a tool to monitor it but as an enforcement layer.

The AAGATE framework offers a more powerful production-based view and introduces formal governance to the Kubernetes-based agent systems [3]. The NIST AI Risk Management Framework is operationalized by AAGATE and incorporates such components of the model as threat modeling (MAESTRO), vulnerability scoring (AIVSS/SSVC), red-teaming (CSA guidelines), zero-trust mesh networks, and explainable policy engines.

This paper demonstrates that autonomous agent systems need multi-layered security controls of behavioral analytics, identity rights frameworks (DIRF), injection protection (LPCI), and cognitive degradation monitors (QSAF). The need to have continuous and audible governance layer is accentuated by these mechanisms that are directly embedded in operations fabric.

The other studies take the discourse on governance into the foundations of corporate, political and environmental spheres. To illustrate this, AI governance through regulation documents reveals the impact of the global policies like the EU AI Act and CSRD on the accountability of organizations and internal control measures [6].

Disclosure, risk reporting and assurance are the key aspects highlighted by these policies which are more than sufficiently aligned with the system level AI governance demands. The AI-Policy-Governance Nexus model demonstrates the shift of organizations towards an AI-based continuous control and a proactive governance reporting system instead of a compliance-based one. In the case of agentic MLOps, this holds that, policy-over-model systems provide a channel of fulfilling operational safety requirements internally and regulatory demands externally.

Literature on security also identifies the issue of cross-jurisdiction governance, particularly in regards to systems which want to use multi-region clouds or ones with exports, or even delicate industrial settings such as semiconductors and storage frameworks [7]. Such insights are directly applicable to the recommended agentic MLOps control plane that is required to implement regional compliance, entitlement policy as well as data-residency constraints via policy-as-code, rather than merely by model-level inference.

This literature proves the fact that governance is not a passive compliance model but an active operational aspect. It further demonstrates that agentic systems must have strict security,

identity, log, audit and rollback facilities all of which are consistent with the concept of this paper (interplaning a Model Custodian Agent)y.

Infrastructure-Oriented Operational Challenges

Much of the process of making autonomous operation safe consists in bringing agent behavior and production-grade operations into contact. The literature on MLOps demonstrates that the process of selecting tools, managing model lifecycle, pipelines coordination and observing production behavior creates a high level of complexity in organizations.

Proposed systematic review of industry MLOps practice provides reference architecture to organise tools, workflow and infrastructure requirements of scalable maintainable machine learning systems [9]. In this paper, the aspects of lineage tracking, monitoring, deterministic rollback, CI/CD integration, and the environment-standardization, which are crucial attributes of agent-driven infrastructure changes, are highlighted.

Recent practice Since the evolution of operational methodology is now described in AgentOps [5], it took the form of MLOps, then LLMOps, then GenAIOps, and finally AgentOps. New problems unique to agentic systems that have been determined in the literature include prompt volatility, multi-agent communication debugging, inconsistent tool use behaviors and ethical risks that are likely to come with the autonomous agents making real-world decisions.

The research indicates that traditional MLOps tools cannot monitor such behaviors as the reflection loops, recursive planning, or agent coordination. It suggests a compounding operational framework that is made up of MLOps controls (datasets, versioning, pipelines) and agent-specific controls (session states, tool logs, memory containers, delegation chains).

These understandings provide a strong rationale to the essence of this research paper: autonomous agents cannot be governed by regular model monitoring systems; agent-specific policy-over-model mechanisms should be taken into account to provide safe operation, hairy autonomy and predictable lifecycle governance.

The literature about the production engineering also emphasizes the significance of the drift detection and system monitoring, and automated rollback behavior. An example is serverless drift detection design which shows how parallel infrastructure can easily scale to continuously track covariate drift at the same time across ML pipelines [10].

It applies to the agentic systems since they tend to act on changing environments, moving distributions of inputs and changing states of the infra. The control plane should also include a continuous drift detection mechanism so that there is no gradual degradation of model or agent behavior but rather is a feature that comes out as very important in the continuous evaluation agents suggested in our system.

The need of having integrated pipelines, strict audit logs, automated rollback as well as real-time monitoring are validated in MLOps and AgentOps-research. These systems compose the control-plane of a policy-over-model system and enable agents acting autonomously to work without posing systemic risk.

Policy-as-Code in Agentic MLOps

In the literature reviewed, one of the prevalent trends can be identified: complex AI systems should be regulated through policies and not actions of the models only. The agent-level autonomy presents the uncertainty, variability, and possible safety issues, particularly when it is used with highly complex engineering and infrastructure systems.

Available literature presents disjointed efforts to get control over autonomy: patterns within architecture to organize tools [1], patterns in security architecture to coordinate between agents [2], Kubernetes-native patterns of governance [3], and methodologies of how to conduct multi-agent systems [5]. Not a single one of those works however suggest a single, cross-stack control plane that serves to combine pipelines of data, model assessment, injury to tool access, infrastructure controls, capitalization and safety direction.

The following gap is what inspires the core input to the current paper namely a policy-over-model agentic MLOps control plane provided by a Model Custodian Agent. This approach is supported by various dimensions that are relevant and pointed out in the literature:

- **Supervision and guardrails -** As has been noted in the literature on agent architecture, model-level abilities are not sufficient to control the use of different tools or provide safety [1][4][8].
- **Embedded in the system —** Integrated within the system SAGA and AAGATE works have indicated that safety is inseparable with the operations of the runtime; it should be included in all the channels of interaction [2][3].
- **Compliance pressure —** Regulatory governance research demonstrates that organizations should have systems that will automatically impose policies across regions, logs, and decision processes [6].
- **Operational realities —** Research findings indicate that in SLOPs and AgentOps, the complexity of operations increase with increased autonomy that needs to be executed within regulated levels of drift, rollbacks, change control and lineage [5][9][10].

The concept of safe autonomy being supported by an integrated layer of governance has high support in the literature. The paper contributes to this area by making policy abstractions, compliance logic, and an ongoing safety assessment the first-class primitives, however, not MLOps, not future AgentOps, not existing security frameworks.

II. METHODOLOGY

Qualitative research approach is implemented in this study to design and describe a policy-over-model control plane over safe autonomy in the engineering and infrastructure settings. The target objective of the methodology is to cognize the ways and means of integrating current agentic AI, MLOps, security, and governance to create a single control and fundamental section that sustains secure and dependable autonomous functions.

Due to the fact that the field is fresh and rapidly evolving, the qualitative research method will be suitable to find patterns, generalize ideas, and create an intellectual framework that links isolated threads of studies in order to form a coherent framework.

The four stages of the methodology include: (1) selection of literature, (2) thematic analysis, (3) cross domain synthesis, and (4) formulation of framework. The actions give the study the ability to combine the knowledge of the extant literature, business practices as well as governance frameworks in a logical manner.

Literature Selection

The initial action involved determining research and industry publications on the topics at agentic AI, MLOps, AgentOps, governance, safety, and autonomous infrastructure operations. The reason to choose ten papers was based on the fact that the articles discussed fundamental aspects of agentic architectures, system safety, tooling to work with, governance structures, and drift or decay detection. These articles are academic research works as well as the new engineering paradigms.

They were selected due to their relevance, conceptual richness and suitability to the purpose of creating a single unified control plane among autonomous agents. There were no quantitative criteria; the selection was conducted according to conceptual value and direct answers to the question of research.

Thematic Analysis

Once the literature was picked, it was analyzed using the thematic approach to assess the similarities of ideas between the sources. These themes were four: (a) an architectural vulnerability of agentic AI systems, (b) increasing demands of embedded governance and security, (c) the operational issues associated with multi-agent and autonomous systems, and (d) continuous monitoring, drift checks, and auditability reasons.

These themes were used to determine the missing system components or the weak components in operations of the agentic operations. The author of this study was reading every paper line by line to identify some concepts which could be applied to autonomy, policy constraints, guardrails, infrastructure risk, and governance mechanisms. Identical procedure was used to industry structures that lay emphasis on drift identification, zero-confidence structures and AI risk management.

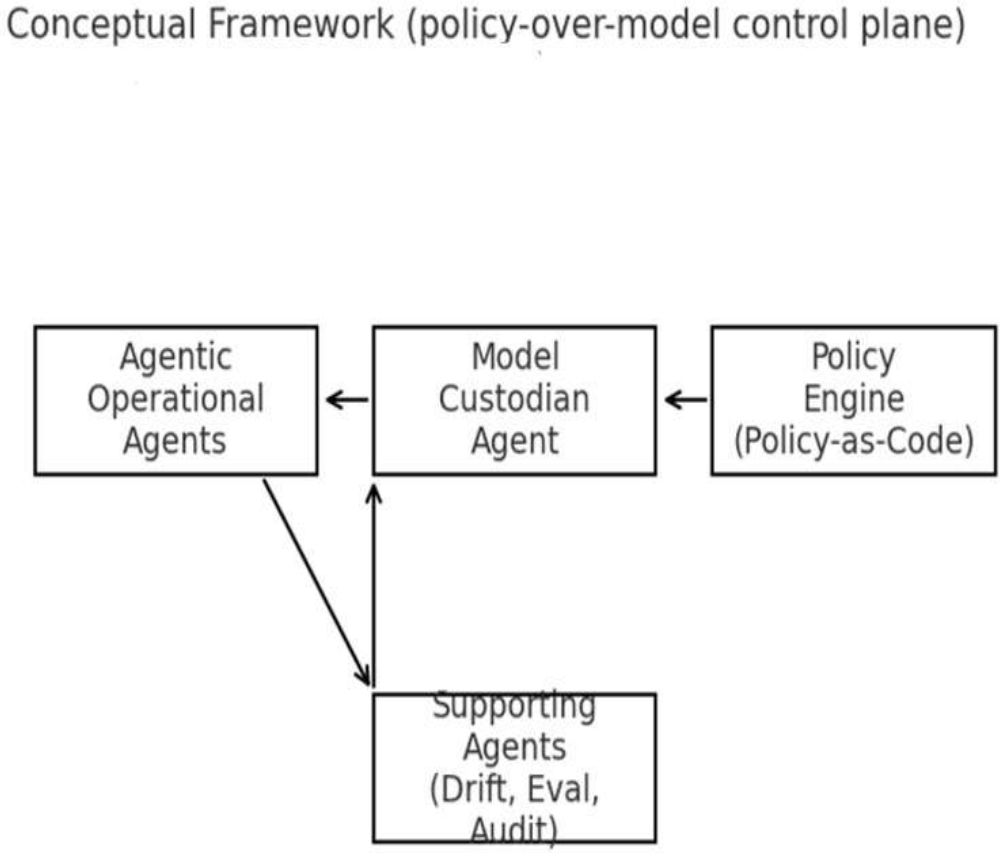
Cross-Domain Synthesis

The themes were identified and compared in various fields like cloud operations, data engineering, cybersecurity and governance. This stage was aimed at acquiring knowledge concerning the possible application of the ideas of one area to safety and oversight in another one.

To illustrate, it can be applied in two ways. Agent-tool authorization guidelines can be guided by the concepts of zero-trust networking, and continuous evaluation agents can be informed by two frameworks: MLOps lineage and drift. The cross-domain synthesis was very useful in highlighting gaps in the current systems like the absence of a central control plane to bind policy enforcement, data pipelines, tool access, and compliance logic.

Framework Formulation

A new conceptual framework was created that is based on the themes and cross-domain synthesis, a policy-over-model agentic control plane in MLOps. The model combines concepts of agentic architecture, risk governance, MLOps controls and CloudOps change-management practice. It proposes the use of central Model Custodian Agent and a provision of auxiliary agents that carry out the drift detection, counterfactual testing, rollout sequencing, and audit logging functions. Policies serve as the control layer of the first class as opposed to model outputs.



This qualitative approach method is useful in establishing a grounded, consistent, and realistic framework. It does not attempt statistical generalization but rather, attempts to develop an articulate well-grounded design on future empirical research regarding safe autonomy in production engineering and infrastructure systems.

IV. RESULTS

Policy-First Control Plane in Agentic MLOps

The qualitative analysis demonstrates that there is a consistent and straightforward conclusion, and it is impossible to safely use current agentic AI systems in production engineering and infrastructure environments without a robust policy-first governance layer. The trend is that in

the literature reviewed, the most common approach is to provide agentic systems with capabilities of models, strategies of reasoning, or planning actions without providing enough attention to the safety of the systems, their compliance and life-cycle controls.

This leaves a significant disparity during interactions between the agents and actual tools, infrastructure services, multi-region deployments, and production data. In the analysis, it was seen that current agentic architectures remain very much model reliant in coming to a conclusion of what to do.

The model inference will not guarantee the safe or compliant behavior. Indicatively, research on orchestration patterns indicate that, when agents are crossing multiple systems, they are faced with routing, choice of tools and boundary issues.

The literature on security reveals that even well-design agents may make unsafe actions rather accidentally when they are not on tight policy guidelines. These gaps prove that policy-over-model concept is not only useful but also required to the systems which should comply with regulations and data residency, change control, and audit requirements.

One outcome is that some unified control plane needs to substitute the concept of agents taking direct action against the tools. The results provide some evidence that a other structure exists in which a Model Custodian Agent interprets some policies, assesses risk, verifies agent requests, and provides an allowance, denial, sandbox or escalation.

This changes the autonomy of model reasoning by free associations to autonomy under control as by rules and under observation. This structure is consistent with findings in the literature in which governance structures (e.g., zero-trust designs, policy engines, identity systems) are necessary whenever agents interact with operations or infrastructure.

The results reveal that policy should be implemented on multiple levels: tool entitlement policies, rules of access to the data, regional limitations, business risks policy, and business operational safety standards. It is not possible to encode these to the LLM. They have to be external and version controlled and enforceable policies which the agents have to adhere. This supports it being concluded that policy-as-code develops into a requirement of any safe agentic MLOps system.

Table 1. Gaps in Existing Systems

Area of Concern	Observed Gap	Implication for Safe Autonomy
Tool use in agentic systems	There were no powerful guardrails on routing of tools or validation.	Agents can generate unsafe/irreversible infrastructure acts.
Security frameworks	Identity is available and authorization is available and does not enforce runtime policy.	Under autonomy, there is an increase in communication and action risks.

MLOps pipelines	The agent decisions are not related to monitoring and lineage.	The agents can do work on old or drifting models.
Compliance rules	Failing to be outlined as system executable constraints.	Regulated regimes and multi-region cannot have trust towards agent operations.

Multi-Agent Oversight

The second key observation is that the evaluation should be performed continuously so as to avoid the possibility of unsafe or degenerating behavior of agents with time. As demonstrated in the existing agentic systems, behavior may drift because of updates in models, context, change of tools, or environmental change.

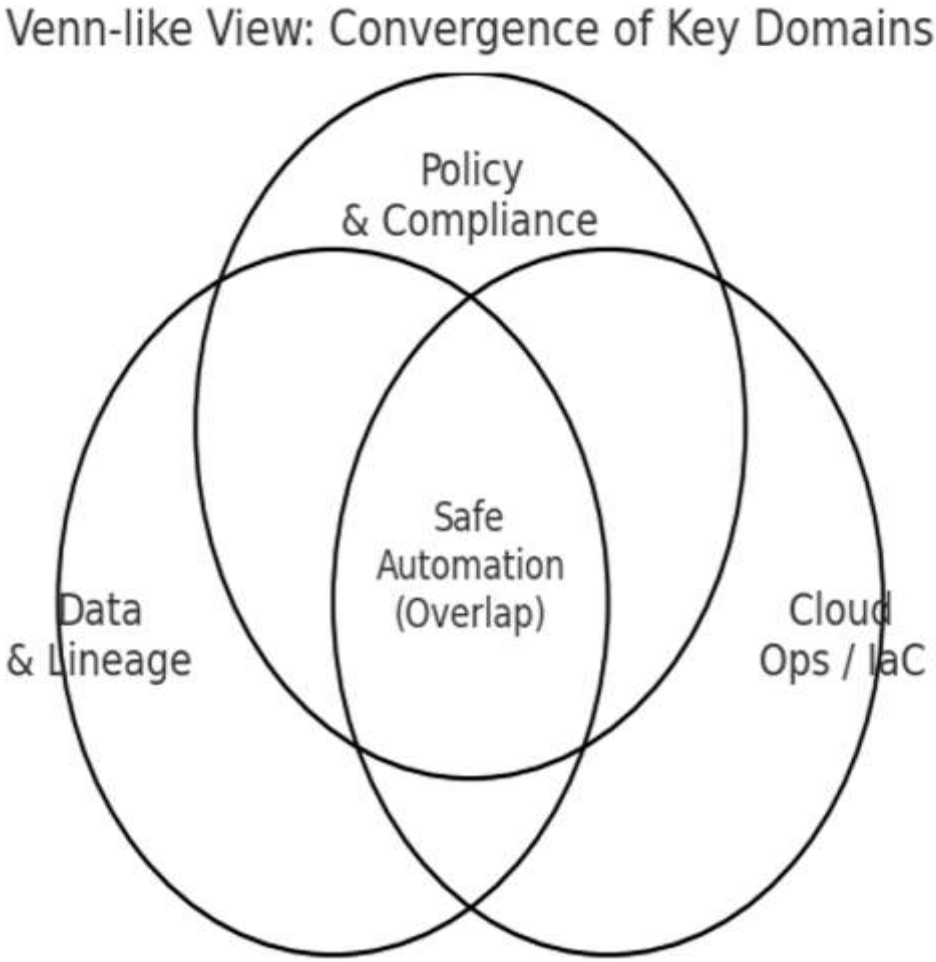
The available literature on drift and decay detection in ML systems demonstrates that drift is not as uncommon as organizations assume, and that once agents act independently, the consequences of the drifting can be significantly greater since the drift can not only influence predictions but also make decisions and take actions.

According to the qualitative synthesis, information continuity has to be in-built within the layer of governance, rather than a controlled tool adjoined to it. The other agents will need supporting agents like evaluation agents, drift-detection agents, and anomaly observers that will be required to execute in parallel with operational agents.

These agents monitor the behavior of the system in question, whether it still behaves in a safe way, whether the decisions are also in line with what is expected, and whether any indications of cognitive impairment or malicious activity are arising. This is in line with the results of security-based studies based on cognitive degradation monitoring, quantum-resilient systems, and agentic threat modeling.

One associated point is that some kind of assessment cannot be restricted to accuracy. These should have safety tests, counterfactual tests, test in scenarios and policy-compliance tests. This concurs with the literature of AgentOps and governance understanding which recognize the need of agent monitoring based on pattern tool-use, sequence decision, reflections loop and chain of interaction. As an example, when an agent generates the series of actions which are not compatible with the established workflows the system should signalize the action and decline to perform.

Sandboxing is also in great demand according to the analysis. All agent action needs to be simulated in an isolated sandbox environment before it is taken into account in production infrastructure modification. The sandbox allows the system to test the violation of any policies, risk, or unexpected outcome of the action. In all the sources reviewed, the theme test before impact is recurrent in many shapes e.g., dry- run, counterfactual assessment and isolated environment testing.

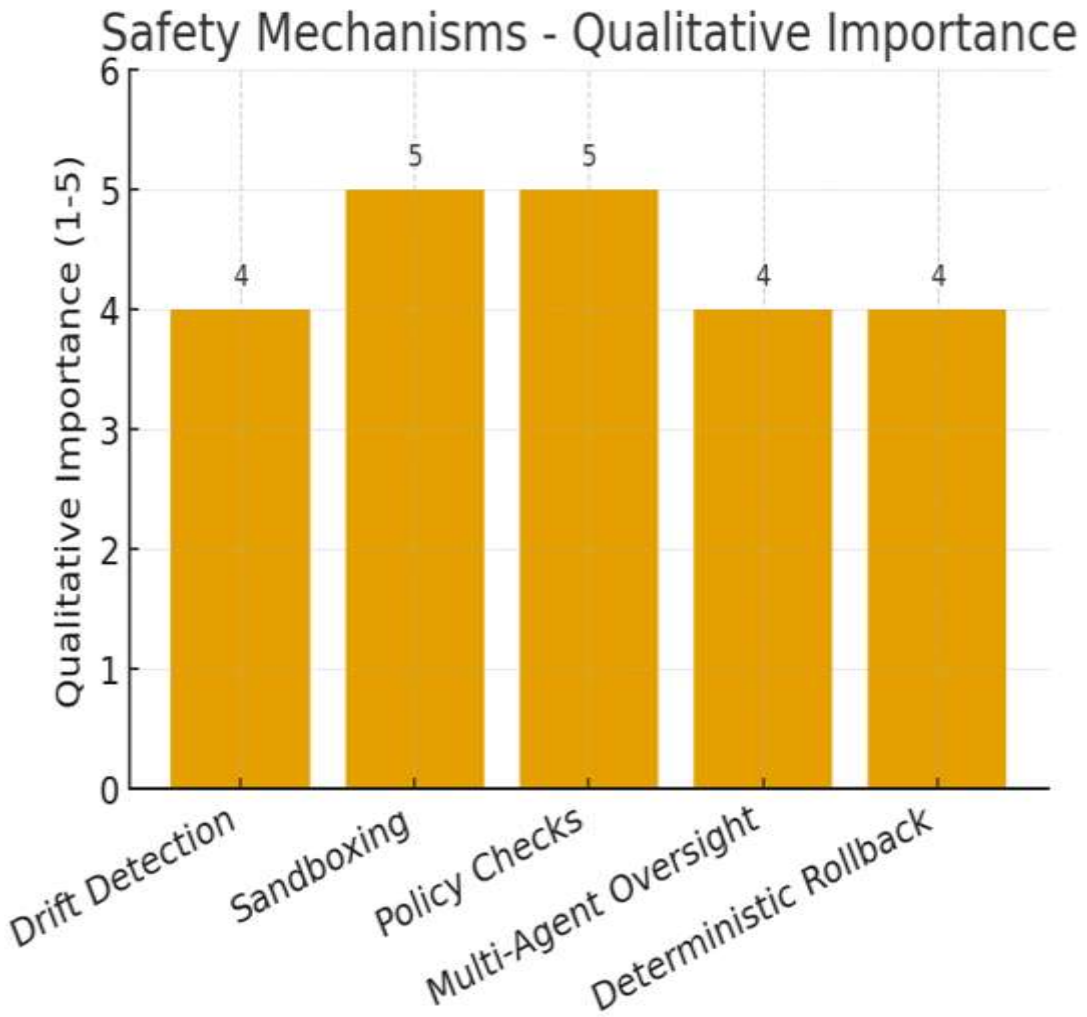


A detailed agent control became a common need. The results show that a single autonomous agent is dangerous to make use of. Rather, the supervision must be spread among specialized agents like in multi-role governance systems of cloud security and infrastructure management. The Model Custodian Agent is made the policy executor and the supporting agents analyze drift, risk monitoring as well as offer proposal validation. It is a system structure that minimizes the risks of a single point of failure and gives multiple layers of safety.

Table 2. Required Safety Mechanisms

Safety Mechanism	Purpose	Expected Benefit

Limitless drift and deterioration identification.	Determine the incidences of agent action alteration.	Ethical Combat unsafe or poor quality of action.
Sandbox simulation	Protest all the suggested activities in advance.	Lower the operational risk and permit a safe experimentation.
Policy compliance checks	Decisions made are required to be subject to rules, entitlements and regulations.	Prevent breaches in regulated systems or multi region systems.
Multi-agent oversight	Sharing of Supervision.	Enhance responsibility and minimize system failure.



Compliance Controls into a Unified Plane

The third important discovery is that safe agentic autonomy can be effective by ensuring that data engineering, model operations and cloud/infrastructure operations are operated on one control plane. These systems frequently are today found as disjointed layers:

MLOps deals with models, CloudOps with infrastructure or landscape and DataOps with lineage and features while compliance teams deal with policy or legal restraints. These boundaries however blur when an agent is touching infrastructure. The action can be basing on a dataset, model analysis, area-variable policy, tool access policy, and infrastructure capacity - altogether.

The qualitative review of the literature demonstrates that one of the greatest risk factors is fragmentation. Specifically, pipelines, lineage, rollbacks, and drift detection are mentioned in

MLOps research as being key to reliability of its models. The studies of the AgentOps indicate, however, that the monitoring of the tool usage, memories, and delegation chains should be provided to agents as well.

Cloud and security frameworks emphasize the importance of having zero-trust identity, policy engines, audit logs and change controls. The operation of autonomous agents in the case where these layers do not work in tandem is a battle since the context of decision-making is inconsistent/incomplete.

The results demonstrate that the cohesiveness of the control plane is the solution to this issue as the layers are brought to a single structure of governance. Model Custodian Agent works as the orchestrator to link DS/DE pipelines, MLOps evaluation, CloudOps tool gating, identity and authorization as well as compliance rules. This would enable all agents request to be implemented with a whole picture of the system state and policies.

The other outcome is that deterministic rollbacks come into play. Through an agent action a system should be able to automatically revert to a safe state in the event that an agent action generates some issues unanticipated to the system. This concept is firmly inspired in MLOps that place their emphasis on reproducibility and rollback and by cloud engineering whereby the state of infrastructure must be invertible. The analysis recommends that the rollback triggers be directly linked with the risk scores, drift indicators or the signal of policy-violation as tracked by the continuous evaluation agents.

The results also indicate that the audit logs have a vital role in the development of trust. All machine-readably, agent decisions, tool actions, authorization actions, policy assessment and outcomes have to be logged. In auditability, governance literature stresses that new AI regulations are necessitating a new capability, auditability, particularly where there are multiple regions. The centralized control plane guarantees that audit trails automatically amass without having to engage individual subsystems to add individual logging logic over them.

The findings indicate that the elements which make safe autonomy dependent are policy-as-code. Entitlements, data redactions and export controls, as well as residency restrictions policies, need to be executable versioned policies. The results emphasize the fact that the encoding policies within model prompts or natural-language instructions is hazardous as the models can unintentionally assume or misunderstand such instructions. A formal policy engine represents a policy that guarantees consistency in policy implementation without any model reasoning.

4. Contribution of the Policy-Over-Model Approach to Safe Autonomy

The last group of findings suggests that a policy-over-model agentic MLOps control plane has a number of valuable advantages:

- It lessens risk with the tools agents not acting directly without the supervision.
- It makes sure that compliance and governance are firmly incorporated into the operations rather than the add-ons.

- It enables organizations to safely scale agentic automation in many regions, lines and compliance areas.
- It encourages reliability in form of continuous assessment and forced sandboxing.
- It establishes good levels of audit and responsibility of any free decisions.
- It fortifies muscle strength because it permits deterministic swindles and multi-excited governance.

The literature evidence demonstrates that there is no system that can provide all such capabilities as a combination. Numerous frameworks tackle the various aspects of the problem, such as architectural abstraction, security governance, MLOps oversight, or drift detection, and none of them assembles them into a single control plane, specifically agentic MLOps.

The results of this study suggest that the discussed solution would be a real gap within the existing body of research since it would designate governance, policy abstraction, and ongoing safety assessment as the central control factors of agentic autonomy in an engineering and infrastructural setting.

V. CONCLUSION

This paper demonstrates that safe autonomy is not only based on powerful models, but also powerful policies, protective surveillance measures and governance agencies operating 24/7 with the model. The results expose some of the main requirements which include drift checking, audit, and controlled access to tools and a central custodian to maintain responsible behaviour. The control plane suggested has presented an effective framework on how to incorporate these aspects into actual operations. Though listed as being conceptual, this study provides very concise basis of testing and simulation action and scale deployments in the future. This framework can further be used to enable organisations to embrace agentic AI with increased confidence and reduced risk with development.

REFERENCES

[1] Kandasamy, S. (2025). Control Plane as a Tool: A Scalable Design Pattern for Agentic AI Systems. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2505.06817

[2] Syros, G., Suri, A., Ginesin, J., Nita-Rotaru, C., & Oprea, A. (2025). SAGA: A Security Architecture for Governing AI Agentic Systems. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2504.21034

[3] Huang, K., Lambros, K. R., Huang, J., Mehmood, Y., Atta, H., Beck, J., Narajala, V. S., Baig, M. Z., Haq, M. a. U., Shahzad, N., & Gupta, B. (2025). AAGATE: A NIST AI RMF-Aligned Governance Platform for Agentic AI. arXiv (Cornell University). https://doi.org/10.48550/arxiv.2510.25863

[4] Bandi, A., Kongari, B., Naguru, R., Pasnoor, S., & Vilipala, S. V. (2025). The rise of Agentic AI: A review of definitions, frameworks, architectures, applications, evaluation metrics, and challenges. Future Internet, 17(9), 404. https://doi.org/10.3390/fi17090404

[5] Joshi, S. (2025). LLMOPs, AgentOps, and MLOPs for Generative AI: A Comprehensive review. International Journal of Computer Applications Technology and Research. https://doi.org/10.7753/ijcatr1407.1001

[6] Cordeiro, C. M., Adomaitis, L., & Huang, L. (2025). The AI-policy-governance nexus: How regulation and AI shift corporate governance toward stakeholders. Technology in Society, 84, 103117. https://doi.org/10.1016/j.techsoc.2025.103117

[7] Adabara, I., Sadiq, B. O., Shuaibu, A. N., Danjuma, Y. I., & Venkateswarlu, M. (2025). A review of Agentic AI in Cybersecurity: Cognitive Autonomy, Ethical Governance, and Quantum-Resilient Defense. F1000Research, 14, 843. https://doi.org/10.12688/f1000research.169337.1

[8] Nisa, U., Shirazi, M., Saip, M. A., & Pozi, M. S. M. (2025). Agentic AI: The age of reasoning—A review. Journal of Automation and Intelligence. https://doi.org/10.1016/j.jai.2025.08.003

[9] Kumara, I., Arts, R., Di Nucci, D., Van Den Heuvel, W. J., & Tamburri, D. A. (2022). Requirements and Reference Architecture for MLOps:Insights from Industry. Requirements and Reference Architecture for MLOps:Insights From Industry. https://doi.org/10.36227/techrxiv.21397413.v1

[10] Sisniega, J. C., Rodríguez, V., Moltó, G., & García, Á. L. (2024). Efficient and scalable covariate drift detection in machine learning systems with serverless computing. Future Generation Computer Systems, 161, 174–188. https://doi.org/10.1016/j.future.2024.07.010