Quarterly Newsletter

CHAIR

iDANAE

INTELLIGENCE · DATA · ANALYSIS · STRATEGY

1Q26

Agentic AI Control Framework

POLITÉCNICA
UNIVERSIDAD POLITÉCNICA DE MADRID
MS Management Solutions
Making things happen

Introduction

A major paradigm shift is occurring in artificial intelligence (AI): the shift from systems primarily designed to retrieve or synthesize knowledge (search engines) to systems capable of planning and executing actions (action engines). The corporate sector is entering a new phase of AI adoption, marked by the development and integration of agentic systems. Unlike a traditional LLM, which is essentially passive and waits for a prompt, an agent can perceive, reason, use tools, and execute complex actions to achieve a goal with minimal human intervention. However, this capacity for action introduces an operational paradox: the greater the autonomy, the greater the risk of unpredictable outcomes.

There is currently a visible gap in the market. Many companies are developing proofs of concept (PoCs) for agents, but large-scale deployment in customer-facing production environments has not yet become widespread because the necessary safeguards are still lacking. An LLM hallucinating in a chat may create a relatively limited operational risk[1], legal, or reputational risk; by contrast, an agent that makes mistakes while executing an action (such as issuing a refund, accessing a database, or triggering a process in a manufacturing plant) poses a critical corporate risk. It may cause material damage, endanger individuals, or result in unlawful actions; the economic impact of an executed action could, in some cases, even exceed the company's financial resources.

Agentic systems present specific challenges that traditional cybersecurity cannot always address on its own. These include non-determinism (as with LLMs, the same input can generate different outputs and actions), infinite loops (with costs that can spiral if the agent fails to resolve a task), and manipulation of the agent's behavior through malicious instructions in the prompt (prompt injection) or in the surrounding context (for example, the incident involving a Chevrolet dealership chatbot that agreed to sell a car for $1 [2]).

These technical risks translate into highly significant operational and organizational challenges. Adopting agentic systems requires redefining processes, performance metrics, and accountability policies; establishing safe pathways to scale from PoCs to production; and ensuring regulatory compliance, cost control, data protection, and the traceability of automated decisions. All of this underscores the need to respond to these new challenges in a structured and thoughtful way by defining a robust control framework that enables companies to deploy reliable, auditable, and secure agentic systems. This publication briefly reviews some of the regulatory standards related to agentic systems, presents the main components of a control framework for agentic systems, and outlines several challenges that must be addressed to ensure the safe use of this new technology.

[1] From a theoretical point of view, the economic impact derived from an error produced by an LLM could become relevant. For example, the use of chatbots has generated precedents where justice has determined that the company is legally responsible for what its AI says on its website, which entails a clear legal risk, and whose economic impact can reach, theoretically, a relevant level for the company, depending on each case. There are rulings where an economic compensation is required for an affected client (see, for example, [17]). To that economic impact must be added the cost entailed for the company to activate the review of the AI system itself, as well as the preparation and execution of its own legal defense, or the possible reputational impact. Therefore, this statement must be understood "in relation to the possible economic impact that an error produced by an agent or agent system can cause", and not as an absolute statement.

Regulatory framework

The regulatory response to the rise of agentic systems has been rapid. Organizations must align their internal strategies with these emerging standards not only to avoid penalties, but also to build trust. Below are three regulatory frameworks that are particularly relevant to the agentic AI landscape in 2026.

EU AI Act

The entry into force of Regulation (EU) 2024/1689 laying down harmonized rules on AI (the EU AI Act, [3]) poses specific challenges for the deployment of agentic systems:

- General-purpose AI (GPAI): The AI Act imposes specific obligations on providers of GPAI models, such as technical documentation, copyright policy, and the publication of a summary of the content used for training. Agentic system deployments must ensure that the underlying models comply with these requirements when applicable.
- Conformity assessment for high risk: Not every agent is, by itself, a high-risk system; the classification depends on the use case and the deployment environment. However, when an agentic system falls into a high-risk category, it must undergo the corresponding requirements and conformity assessment, including the Human-in-the-Loop (HITL) requirement.

ISO/IEC 42001

ISO/IEC 42001 is an international benchmark for AI management systems (AIMS). Compared with purely local or sector-specific approaches, ISO/IEC 42001 provides a common framework for establishing, implementing, maintaining, and continually improving AI governance [4].

For agent-based systems, the standard is particularly useful due to its emphasis on the continuous lifecycle and the systematic management of risks and opportunities. In practice, this approach requires formal processes for change management, documentation, traceability, and continuous system improvement.

- Change management: evaluating how new data, new tools, prompt changes, or model fine-tuning might affect the agent's behavior.
- Risk and impact assessment: analyzing the operational, ethical, and compliance consequences associated with the system, both before deployment and during operation.
- Traceability and transparency: adequately documenting the system's design, operation, and controls—a critical aspect when agents make decisions or execute actions autonomously.

Singapore Model AI Governance Framework for Agentic AI (MGF)

In January 2026, Singapore's Infocomm Media Development Authority (IMDA) presented a reference framework on the governance of agentic AI models at the World Economic Forum[2] [5]. This is one of the first public reference frameworks specifically geared toward the governance challenges of agent-based systems, moving beyond frameworks focused exclusively on generative AI.

The framework proposes a governance structure based on four critical dimensions that organizations should consider:

1. Early risk assessment and bounding: Determining whether the task is suitable for autonomy and establishing limits on the agent's capabilities, access, and scope of action.
2. Meaningful human accountability: Technical autonomy does not eliminate legal and ethical liability. The framework proposes defining mandatory checkpoints where execution pauses to require explicit human approval for high-impact decisions (HITL).
3. Technical controls and processes: Includes the implementation of specific evaluation, monitoring, identity, permissions, testing, and red-teaming measures to subject agents to realistic adversarial scenarios.
4. End-user responsibility: Involves informing the user that they are interacting with an agent and equipping them with the necessary information to understand its limits and use it appropriately, thereby reducing automation bias.

[2] In 2020, the second version of the Model AI Governance Framework had been published, oriented towards traditional AI, and in 2024 the Model AI Governance Framework for Generative AI was published, an annex to address the specific risks of content creation (such as hallucinations and intellectual property).

Main components of the control framework for an agentic system

A control framework for an agentic system must consider multiple components: (i) sound governance processes; (ii) proper planning and organization during system development; (iii) the incorporation of technical and architectural elements that include defensive barriers against malfunction or misuse; (iv) tracking and monitoring systems that make it possible to identify potential errors in system operation, usage, expected outputs, and additional impacts related to technological resource consumption or operating costs; and (v) the possibility of incorporating human oversight into the system's workflow, among others. These components must be mutually consistent and address the different sources of risk that can trigger economic losses (arising from operational, reputational, legal, and other areas). This section explores several components of a control framework linked to technical elements (and therefore does not go into governance or organizational aspects in depth):

1. Designing the technological architecture of the system to allow for the incorporation of controls and firewalls for evaluating and detecting errors.
2. Establishing guardrails applicable during the system's execution processes.
3. Measuring performance metrics (for both the models and the overall system) and establishing mechanisms to ensure controlled behavior or system shutdown in the event of a failure.
4. Incorporating mechanisms to control potential additional impacts derived from errors (e.g., inefficient consumption of technological resources or increased costs).
5. Finally, incorporating human oversight during process execution, both in critical decision-making and in the acceptance of results (intermediate and final).

These are briefly outlined below.

Architecture as a control element

Architectural design is a determining factor in ensuring control over agentic systems. Agentic AI systems engineering is converging toward modular architectures, in which decoupling functions such as planning, memory management (short- and long-term), and tool use enables more robust orchestration [7]. This transition is not merely a technical evolution that enables model interchangeability; it also makes it easier to incorporate control mechanisms and structures into the system. By making the separation between orchestration, memory, tool use, and execution more explicit, these architectures improve traceability, evaluation, component-level analysis of results, and the application of firewalls to prevent error propagation or establish lines of defense against attacks or malfunctions. From a governance perspective, they also enable a clearer internal allocation of responsibilities, in line with the principles of governance, control, and accountability promoted by the EU AI Act, ISO/IEC 42001, and the Singapore Model AI Governance Framework for Agentic AI.

In practice, these architectures materialize in a functional separation into layers, each with specific responsibilities and well-defined control points. A five-layer schema for an agent system is described below[3]:

1. **Interface and perception layer:** this is the system's boundary. Its responsibility goes beyond simple communication: it acts as the first cognitive firewall. Its critical function is to receive user inputs (the goal the system must achieve) and deliver outputs, while actively filtering prompt injections before they reach the core. Typical control components in this layer include API gateways, guardrails, and data sanitization.
2. **Orchestration and planning layer:** This constitutes the system's strategic and control core. Its primary function is to abstract the task: instead of executing actions atomically, it breaks down high-level objectives into a structured plan or execution flow. This layer coordinates the selective activation of specialized tools or sub-agents and preserves operational coherence through state management mechanisms. Common components include planners, routers, and monitoring/error recovery mechanisms [8].

[3] For simplicity, a reduced vision is shown in this document. For a full discussion of this architecture, different options, and deployment use cases, see [6].

3. **Agent core layer:** This is the layer where the agents reside. Each agent is an encapsulated autonomous unit with a specific purpose (role), a set of instructions (system prompt), and an assigned Large Language Model (LLM). Its function is to perform focused execution of the tasks entrusted to it.
4. **Tools and services layer:** This equips agents with the capacity for real-world impact and interaction. Without these tools, an agent is just a chat; with them, it is an operational system. Control components like sandboxing (isolated execution environments) and the incorporation of human-in-the-loop review mechanisms for critical actions are essential here.
5. **Memory and knowledge layer:** This layer manages the agent's operational continuity and access to relevant context over time. Common components include short-term memory—linked to session history and state—and long-term memory—often implemented through external retrieval mechanisms like Retrieval-Augmented Generation (RAG) supported by embeddings and vector search. It is highly critical to establish data governance controls in this layer to prevent the improper retention or retrieval of confidential information, using data lifecycle policies and filtering mechanisms during the retrieval phase, in line with the privacy-by-design principle [9]. Robust governance in this layer ensures that session history and retrieved corporate knowledge do not become vectors for leaking confidential or unauthorized personal data in line with the data security and accuracy requirements in regulated environments, such as the European Union's framework for providers of high-risk AI systems.

This layered design prevents a single point of failure from collapsing the entire system. If one component fails, the orchestration or tools layers can intercept the error, contributing to a more resilient and fault-tolerant system.

In addition, security, authentication, and user profiling structures must be considered in architectural design. For example, a common approach is the use of Zero Trust paradigms, where implicit trust is never granted to users, devices, or services based on their network location, but instead, strict authentication, authorization, and access controls are required for all system resources [6].

Dynamic guardrails

In the context of autonomous agents, control mechanisms must go beyond the system's initial configuration: they must operate continuously during execution, accompanying every decision and every action. The incorporation of autonomous agents introduces a critical new risk vector: the capacity to act. Recent research warns that agentic systems are vulnerable to prompt injection and to the inter-agent propagation of

malicious instructions: a compromised agent can inject malicious instructions into another agent's memory, creating a cascade of systematic failures [10].

To mitigate this risk, it is not enough to simply instruct the model well. It is necessary to implement dynamic guardrails: layers of deterministic software that wrap around the probabilistic model. When the AI system lacks the intrinsic capability to validate the correctness, truthfulness, or acceptability of its own actions in real-time, this validation is delegated to a set of external controls that intercept traffic before and after each inference.

From an operational standpoint, and in line with risk-management and security frameworks such as the NIST AI Risk Management Framework (AI RMF) [11], including its update for the generative AI profile [12], and the OWASP Top 10 for LLMs [13], these controls can be organized into four complementary defense domains that allow risks to be managed in a granular manner (see Table 1).

- **Data guardrails:** Focused on ensuring data quality, integrity, privacy, provenance, minimization, and protection to prevent data poisoning of inputs feeding into the system. Through validation, sanitization, classification, and privacy controls, these mechanisms reduce the exposure of sensitive information and reinforce the reliability of the context in which the agent generates its responses.
- **Model guardrails:** Centered on supervising the security and robustness of the generative core. They include moderation mechanisms, input/output validation, detection of prompt injections and jailbreaks , coherence checks, and evaluators based on other models (like LLM-as-a-judge) that can monitor system parameters and/or results. Their goal is to reduce unsafe, out-of-bounds, or obviously erroneous responses before they reach the user or trigger further actions.
- **Application guardrails:** Operate on the business logic, restricting what the agent can do within the limits of resilience and observability, as well as its interaction with tools and APIs. They include parameter validation, policy enforcement, permission limits, structural output validation, and incorporating human-in-the-loop points for sensitive operations. Furthermore, the dynamic selection of external tools and services requires monitoring which services an agent can invoke autonomously, to maintain control over data processing, traceability, and the assignment of responsibilities.
- **Infrastructure guardrails:** Physical and logical security and protection mechanisms on the platform where the system runs (cloud or on-premise). They include access controls, identity management, environment segregation, process and tool isolation (sandboxing), and anomaly monitoring. Its purpose is to mitigate risks such as unauthorized access, privilege escalation, or data breaches.

Performance metrics and error handling mechanisms

Beyond architectural design, once deployed in production an agentic system requires mechanisms for deep monitoring and observation (known as "observability"). The incorporation of granular metrics makes it possible to detect inefficient executions or the presence of anomalies and errors, and to activate controlled response mechanisms when an error is identified. Integrating these elements also helps strengthen the maintainability and analysability attributes defined in the ISO/IEC 25059:2023 quality model for AI systems [14].

Here are four mechanisms that improve anomaly detection and error management:

1. Measuring model and system latency. This helps identify malfunctions, inefficient resource allocation, or overloads. The following metrics can be included:
    a. Rendimiento del modelo: mide la velocidad bruta del motor cognitivo. Se incluyen *time to first token* (TTFT, crítico para la percepción de latencia) y *time per output token* (TPOT, crítico para el rendimiento total).
    b. Rendimiento del sistema: captura la sobrecarga de la arquitectura del agente. Incluye la latencia de orquestación, los *spans* (segmentación temporal de cada operación que realiza el agente) y la latencia de las herramientas.

[4] Instruction injection attacks (adversarial prompt injection) designed to bypass security filters and ethical safeguards integrated by developers. In agentic systems, the danger of a jailbreak lies in an agent of the system ordering something improper, or the system executing something improper (such as, for example, launching malicious code or deleting a database), given that it is connected to tools outside the agent or the agentic system.

Table 1: Summary of defense domains, covered risks, and associated controls

Defense Domain	Mitigated Risks	Key Controls (ISO 42001 / NIST / Singapore)	Impact on Legal Liability
Data	▸ PII (Personally Identifiable Information) leaks ▸ Biases ▸ Data poisoning	▸ Data minimization ▸ Context sanitization (RAG) ▸ Provenance auditing	GDPR compliance and duty of care in information asset management
Model	▸ Jailbreaks ▸ Hallucinations ▸ Out-of-scope responses	▸ Input/output moderation ▸ External evaluators (LLM-as-a-judge) ▸ Allignment techniques	Evidence of technical oversight to mitigate algorithmic negligence
Application	▸ Ejecución de acciones no autorizadas ▸ Errores contractuales	▸ Human approval points ▸ API parameter validation ▸ System-level guardrails	Protection against the formation of erroneous electronic contracts or involuntary binding
Infrastructure	▸ Privilege escalation ▸ Unauthorized Access to critical systems	▸ Agent sandboxing ▸ Process isolation ▸ Network anomaly monitoring	Prevention of harm to third parties and compliance with industrial cybersecurity standards

2. Monitoring token usage, setting usage limits, dynamic model selection, and architectural optimization reduce unnecessary calls.
3. Fallback systems and error management: Because agentic systems are probabilistic and have external dependencies, it is crucial they are designed to fail gracefully (in line with the resilience principle of the NIST AI RMF [11]). Otherwise, these errors could cause the system to crash, trigger infinite loops, or generate incorrect responses. Common mechanisms include controlled retries, functional fallbacks (deterministic rules upon error), or circuit breakers (isolating a failing component from the execution flow).
4. Understanding decisions: Evaluating the agents' decision sequences is essential, since its non-deterministic behavior in decision-making or in generating results renders standard tests alone insufficient. This involves examining the quality of decisions, the soundness of the reasoning process, and coherence of the output, by tracing the steps, tool selection, and strategies used.

Ultimately, these mechanisms are not merely a technical control requirement, but a legal safeguard, as they help reconstruct the system's reasoning process in the event of an incident, thereby facilitating explainability and making it possible to determine whether an error resulted from malicious user input, a structural flaw in the agent's planning logic, or previously incorporated modifications introduced maliciously.

Impact management mechanisms

Unlike traditional software or direct queries to an LLM, where the impact of an action is relatively predictable, agent-based systems introduce far greater variability. Faced with a complex problem, an autonomous agent may decide to make multiple tool calls or iterate many times before reaching a final solution, which can result in inappropriate or inefficient use of resources, as well as increased associated costs. Without strict control mechanisms, anomalies such as infinite loops can exhaust budgets in a short period of time. To mitigate this operational risk, various technical solutions can be incorporated:

- Traceability and cost attribution: the incorporation of forecasting, traceability, and cost control platforms that complement the billing systems of model providers. These platforms allow each request to be tagged, token

consumption to be broken down, and the exact expense to be attributed to each agent.

- Dynamic model selection routing: not all tasks require the reasoning capacity of the most advanced and expensive models. Modern architectures implement dynamic routers that evaluate the complexity of the request before making the call to the model. If the task is simple, the system routes the request to a smaller, faster, and cheaper model, reserving the most powerful models for critical reasoning.
- Architecture optimization and token usage: restrictions must be coded at the orchestrator level, defining a maximum use of tokens per task or session and designing a robust architecture that reduces redundant steps and avoids unnecessary reasoning.

Human supervision

Despite advances in automatic control mechanisms, human intervention (human-in-the-loop, or HITL) remains an indispensable element in agent systems. In high-risk environments or in relation to high-impact actions, the system must incorporate effective mechanisms for human supervision and approval at significant points in the workflow, for example to verify the correctness of an intermediate or final result, or to approve part of a process and authorize continuation to the next step. This supervision does not eliminate the agent's autonomy, but it does introduce a level of control proportionate to the degree of risk, authority, and operational context. AI governance frameworks and regulations regard human oversight as a key element in the development and operation of responsible and safe AI systems.

The human-in-the-loop pattern can be understood as an interaction protocol that operates under two premises:

1. High-impact validation: actions classified as irreversible or critical, such as bank transfers or code deployment, require human validation.
2. Uncertainty management: when the model's confidence falls below a predefined threshold, the system escalates the case to an operator instead of generating a potentially erroneous response.

From a technical point of view, these aspects can be resolved through orchestration frameworks (such as LangGraph), which allow establishing logical interruption points[5]:

- Suspension: the agent stops its execution right before the action tool, saving its entire context in a database.
- Intervention: a human operator receives a notification, being able to approve, reject, or modify the state.
- Resumption: the agent resumes the task with the state updated by the human, executing the action as if it had reached that conclusion itself.

The implementation of HITL converts each human intervention into a data asset. Every time an operator corrects an agent, a high-quality input-correction data pair is generated. These data can feed subsequent processes of evaluation, fine-tuning, or improvement of the system, reducing over time the need for intensive supervision.

[5] In some complex cases, such as the review of a model whose confidence falls below a certain threshold, the application of intervention supervision mechanisms may not be as direct.

Business challenges

The convergence of advanced technical capabilities and new regulatory requirements gives rise to a series of challenges in the business sphere. The question is no longer simply whether agents work, but how to integrate them safely, efficiently, and in alignment with business objectives.

Indeed, as agents acquire greater autonomy, planning capacity, and access to external tools, new challenges emerge in terms of control, compliance, cost, and risk management. In this context, evolution involves moving experimental approaches towards solid operational frameworks that allow scaling autonomy in a safe and measurable way.

Below, some of these challenges and ways to address them are **explored.**

1. Redefinition of processes and new performance metrics. Traditional processes are not designed for non-deterministic flows or for agents that plan, invoke tools, or adjust their behavior in real time. The challenge consists of redesigning key processes to incorporate controlled autonomy, as well as establishing new specific KPIs (e.g., escalation ratio to HITL, cost per action, or planner stability) that allow evaluating value, efficiency, and risk.

2. Scaling from PoCs to production with real guarantees of safety and cost. Some projects can get stuck in pilots that do not scale due to fear of hallucinations with operational impact, unpredictable costs, or lack of auditing. The challenge consists of industrializing a safe path to production through layered architecture, dynamic guardrails, and observability, deploying agents with a controlled cost and full traceability of decisions and actions.

3. Regulatory compliance and operational risk in autonomous agents. The arrival of the EU AI Act, ISO/IEC 42001, and the new international frameworks for agents demands that companies demonstrate control over automated decisions and over the use of GPAI models. The challenge consists of designing an internal framework that ensures conformity, manages impact assessments, limits critical actions, and documents the decision logic without slowing down the speed of innovation.

4. Data protection and secure management of agent memory. Agents that store persistent memory or use RAG can reveal sensitive information to unauthorized users if strict controls do not exist. The challenge consists of guaranteeing a safe use of short- and long-term memories, avoiding leaks, incorrect attributions, or unwanted learning, and ensuring that only the information necessary for performance is retained without compromising privacy.

5. Integrating observability and traceability as critical business requirements. The autonomy of agents introduces operational uncertainty: non-deterministic decisions, long sequences of actions, and external dependencies that are difficult to predict. Without adequate monitoring, a minor error can escalate to become a systemic failure or a significant cost overrun. The challenge consists of equipping oneself with capabilities to monitor, record, and explain every action of the agent in real time, guaranteeing complete audits, early detection of anomalies, and a safe operation aligned with business objectives.

6. Dynamic lifecycle management. The learning and adaptation capabilities of agent-based systems challenge traditional static validation models. These systems are likely to progressively incorporate mechanisms for continuous learning, adaptation, and iterative improvement based on experience and evaluation, reducing in some cases the need for human intervention in fine-tuning and corrective maintenance tasks [15]. Auditing will therefore need to become a continuous process. Mechanisms will be required to ensure that model adaptations do not shift system behavior away from corporate policies or introduce biases that went undetected during the design phase.

7.Hybrid systems and operational certainty. For highly regulated sectors, the probabilistic nature of LLMs continues to create control and validation challenges. In this context, hybrid architectures are emerging that combine the reasoning and orchestration flexibility of LLMs with rule-based components or specialized models for specific tasks [16]. In certain processes, this combination makes it possible to impose deterministic constraints on specific actions or outputs, thereby strengthening robustness, interpretability, and operational control at critical points in the workflow.

Altogether, these challenges reflect that the adoption of agents is not solely a technological challenge, but a change in the way of operating, governing, and scaling automated decision systems.

Conclusion

The move toward agentic systems marks a structural change in the way intelligent automation is designed, executed, and governed. The autonomy of these systems opens up transformative potential but also introduces significant operational, ethical, and regulatory risks.

To harness this value safely, a control framework must be adopted that combines modular architectures, dynamic guardrails, advanced monitoring capabilities, management of the impacts of system errors, and human oversight.

In this context, data governance, lifecycle management, and regulatory compliance emerge as essential pillars to guarantee that agents operate with transparency, resilience, and alignment with business objectives.

A well-designed control framework does not limit innovation; rather, it enables it by providing the trust needed for agents to act as true performance accelerators across many types of tasks.

Bibliography

[1] S. Nath, R. W. White, F. E. Faisal, M. E. Sharp, R. W. Gruen y L. R. Sivalingam, From Search Engines to Action Engines, Computer, vol. 58, nº 6, p. 59–68, June 2025.

[2] S. McGregor, Incident ID 622: Chevrolet Dealer Chatbot Agrees to Sell Tahoe for $1, AI Incident Database, 2023.

[3] Regulation (EU) 2024/1689 of the European Parliament and of the Council.

[4] International Organization for Standardization, Information technology - Artificial intelligence - Management system (ISO/IEC 42001:2023), 2023.

[5] Infocomm Media Development Authority (IMDA), Model AI Governance Framework for Agentic AI (Version 1)., Government of Singapore, 2026.

[6] National Institute of Standards and Technology, Zero Trust Architecture (NIST SP 800-207), U.S. Department of Commerce, 2020.

[7] J. Luo y al., Large Language Model Agent: A Survey on Methodology, Architectures, and Applications, arXiv:2503.21460, 2025.

[8] D. Souza y P. Machado, Toward Architecture-Aware Evaluation Metrics for LLM Agents, arXiv:2601.19583, 2026.

[9] Z. Zhang y al., A Survey on the Memory Mechanism of Large Language Model-based Agents, ACM Transactions on Information Systems, vol. 43, nº 6, pp. 1-47, April 2025.

[10] D. Lee y M. Tiwari, Prompt Infection: LLM-to-LLM Prompt Injection within Multi-Agent Systems, arXiv:2410.07283, October 2024.

[11] National Institute of Standards and Technology, AI Risk Management Framework (NIST AI RMF 1.0), U.S. Department of Commerce, 2023.

[12] National Institute of Standards and Technology, AI Risk Management Framework: Generative Artificial Intelligence Profile (NIST AI 600-1), U.S. Department of Commerce, 2024.

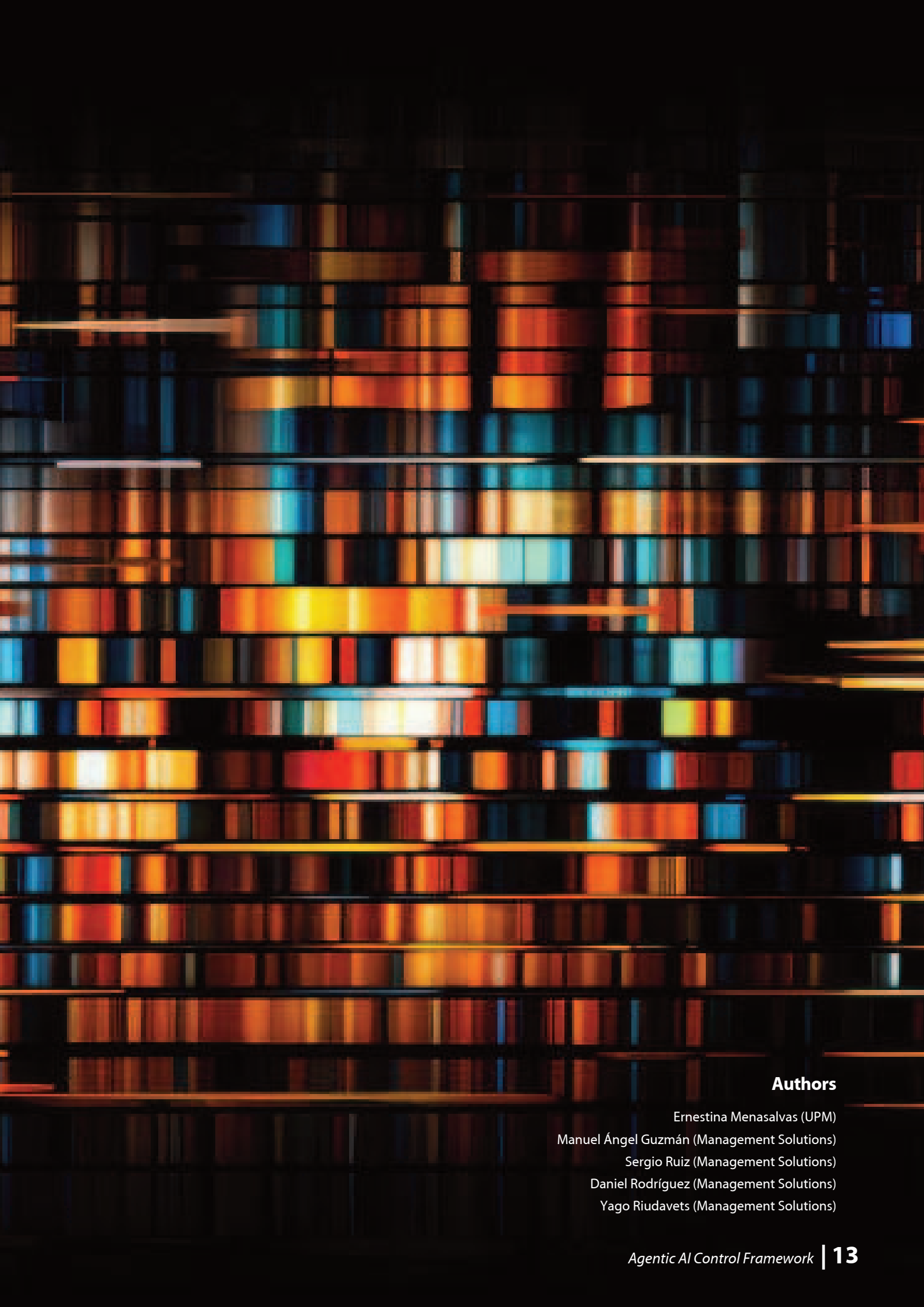
[13] OWASP, Top 10 for LLM Applications 2025, Version 2025.

[14] International Organization for Standardization, Software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Quality model for AI systems (ISO/IEC 25059:2023).

[15] H. Gao y al., A Survey of Self-Evolving Agents: What, When, How, and Where to Evolve on the Path to Artificial Super Intelligence, arXiv:2507.21046, 2026.

[16] M. A. Farahania, M. I. Khana y T. Wuest, Hybrid Agentic AI and Multi-Agent Systems in Smart Manufacturing, 2026. [En línea]. Available: https://arxiv.org/pdf/2511.18258.

[17] Moffatt v. Air Canada. Case number BCCRT 149. Court: Civil Resolution Tribunal, British Columbia., 2024.



Authors

Ernestina Menasalvas (UPM)
Manuel Ángel Guzmán (Management Solutions)
Sergio Ruiz (Management Solutions)
Daniel Rodríguez (Management Solutions)
Yago Riudavets (Management Solutions)

UNIVERSIDAD POLITÉCNICA DE MADRID

The Universidad Politécnica de Madrid is a multi-sector and multi-disciplinary Public Law Entity, which carries out teaching, research and scientific and technological development activities.

www.upm.es

Management Solutions
Making things happen

Management Solutions is an international consulting firm, focused on business, finance, risk, organization, technology and process consulting, operating in more than 50 countries and with a team of 4,000 professionals working for more than 2,200 clients worldwide.

www.managementsolutions.com

For more information visit

blogs.upm.es/catedra-idanae/