

DIGITAL ACCESS TO SCHOLARSHIP AT HARVARD
DASH.HARVARD.EDU

HARVARD LIBRARY


Architecting Trust: A Modular Framework for the Operational Deployment of Autonomous Systems

Citation

Pederson, Joel. "Architecting Trust: A Modular Framework for the Operational Deployment of Autonomous Systems." in Techology & National Security Review, Vol. 1. Presented at MIT/Harvard Techology and National Security Conference, Cambridge MA, April 3–4, 2026.

Link

https://dash.harvard.edu/handle/1/42734663

Terms of use

This article was downloaded from Harvard University's DASH repository, and is made available under the terms and conditions applicable to Other Posted Material (LAA), as set forth at

https://harvardwiki.atlassian.net/wiki/external/NGY5NDE4ZjgzNTc5NDQzMGIzZWZhMGFlOWI2M2EwYTg

Accessibility

https://accessibility.huit.harvard.edu/digital-accessibility-policy

Share Your Story

The Harvard community has made this article openly available.
Please share how this access benefits you. Submit a story

Architecting Trust: A Modular Framework for the Operational Deployment of Autonomous Systems

Joel Pederson

System Design and Management, Massachusetts Institute of Technology

MIT–Harvard Technology & National Security Conference (April 3–4, 2026)

Author Note

The views expressed in this paper are those of the author and do not reflect the official policy or position of the author's employer.

Correspondence concerning this article should be addressed to Joel Pederson, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Building E40-315, Cambridge, MA 02139.

Email: pederson@mit.edu

Author Bio

Joel Pederson is a Senior Systems Engineer specializing in complex algorithm development within the defense sector. He is currently pursuing an M.S. in Engineering and Management through the MIT System Design and Management (SDM) program. He holds an M.S. in Engineering Management from Northeastern University and a B.S. in Mechanical Engineering from the University of Hawaii at Manoa.

Abstract

For the United States and its allies, the incorporation of non-deterministic Artificial Intelligence (AI) and Machine Learning (ML) systems into tactical platforms presents significant challenges. Certifying these agents for safety-critical operations remains a considerable barrier to their deployment and widespread adoption. This paper examines how AI-enhanced tactical solutions can be effectively fielded despite the inherent risks involved. The analysis begins by discussing the history of automation bias in weapons systems and the novel vulnerabilities introduced by AI/ML-powered solutions. The challenge with these emerging techniques is that unlike traditional software, these algorithms present risks throughout their lifecycle. From adversarial perturbations that can be introduced during model training to stochastic failures during operation, the risks associated with AI/ML-based algorithms cannot simply be mitigated by the legacy safety systems embedded in platforms today. While traditional "physics-based" guardrails (e.g., automated ground collision avoidance) effectively prevent kinematic disasters, they on their own lack the sophistication to address the cognitive and perception errors inherent to modern AI. To safely proliferate AI/ML solutions, a new safety-focused reference architecture is required.

This paper proposes using a modular "Safety Sidecar" architecture that operates across the software's lifecycle. The research defines a strategy that acquisition authorities can consider building from to systematically embed safety barriers directly into the system's training, perception, and planning loops. The framework outlines how the training of models can be protected and refined for operational fielding through a structured lifecycle assurance approach. Specifically, the architecture introduces a "Perception Gatekeeper" which validates input integrity against adversarial or degraded sensor data in real-time. The framework also integrates algorithms running operationally in the loop with AI/ML models to function as a "Model

Constraint Guardian" to enforce safety barriers directly into the system's perception and planning loops. The Safety Sidecar concept can function as a unified framework for AI/ML lifecycle assurance. The approach can facilitate effective integration of safety into both developmental and operational phases of system development. This contribution provides a structured pathway to help ensure that safety is a continuous property from model training through to battlefield deployment.

Keywords: AI Assurance, Trusted Autonomy, Runtime Assurance, Autonomous Weapons Systems, Safety-Critical AI

Architecting Trust: A Modular Framework for the Operational Deployment of Autonomous Systems

The continual evolution of warfare is reaching the limits of human cognitive capacity. Offensive capabilities are becoming more affordable, intelligent, and interconnected, while defensive platforms are increasingly automated and decentralized. This shift represents a fundamental change in the nature of conflict. Warfare is rapidly evolving across all domains from subsea to space, driven by technological proliferation[1]. Over the past 90 years, computing has increasingly made its way into warfare as a means of enhancing lethality through augmenting standalone human capability via automation.

This trajectory was broadly anticipated by early visionaries. Vannevar Bush's 1945 essay "As We May Think"[2] described the concept of a machine that could assist humans with the process of thinking while remaining under direct control by the operator. Decades later, Steve Jobs famously described the computer as "a bicycle for our minds"[3]. However, neither Bush nor Jobs could have completely foreseen the profound leap in computing from that of a passive assistant to an autonomous agent. Today, basic machine learning algorithms, which are used as small components within a larger software architecture such as those supporting threat assessment on the Navy's Aegis combat system[4], require human input in order to take an action. In these legacy paradigms, the software recommends an engagement, but a human operator remains "in the loop" to verify the decision and prevent fratricide.

While this "human-in-the-loop" safety architecture has served relatively well in the past, it simply cannot keep pace with the speed at which future conflicts will be fought. As offensive drone swarms and hypersonic missile threats compress decision timelines, the human operator becomes the primary bottleneck in the kill chain. It is clear that increased autonomy will be

needed to augment human warfighting capabilities[5,6,7], but blind trust in a system's autonomous capability has risks. History has shown that erroneous decisions made by tactical software can lead to fratricide and mission failure[8].

The pace of warfare's evolution combined with rapid developments in commercial Artificial Intelligence (AI) and Machine Learning (ML) present both risks and opportunities for the United States (U.S.) and its allies. In order to maintain asymmetric advantages over adversaries, autonomous systems must be deployed across domains from subsea to space. However, the Department of Defense (DoD) faces a certification issue. Traditional software safety standards, such as DO-178C, rely on deterministic logic where every line of code is verifiable. Modern AI tools, commonly powered by deep neural networks, are non-deterministic "black boxes". They are susceptible to hallucinations, adversarial perturbations, and stochastic failures that traditional "physics-based" guardrails cannot comprehensively detect.

Placing trust in non-deterministic AI-powered black boxes rightfully gives U.S. and allied forces pause. Policymakers hesitate to advocate for proliferated autonomy when the risks are uncertain. Commanders cannot trust an agent with lethal autonomy when it remains so fundamentally vulnerable to taking the wrong action based on sensor noise or adversarial spoofing. Simultaneously, the industrial base faces barriers in delivering capabilities that cannot be verified by traditional means. Yet the strategic necessity of autonomy remains. We cannot wait for a "perfect" AI solution. Holding out for a capability that is 100% explainable and error-free is a strategy for obsolescence.

Instead, modular "Safety Sidecars" should be plugged into autonomous systems to validate inputs and outputs in real-time. This paper describes a reference architecture that decouples "safety" from "intelligence."

Importantly, this is a conceptual framework rather than a validated solution. This modular reference architecture is presented as a starting point to structure the conversation among systems engineers, researchers, and acquisition professionals about how safety can be systematically embedded into AI/ML systems. Substantial research, prototyping, and testing will be required before these concepts can be deployed on tactical platforms.

The proposed architecture builds upon the emerging Runtime Assurance (RTA) architecture (ASTM F3269-17) which, although initially developed for unmanned aircraft systems, contains principles applicable to autonomous systems more broadly. However, the standard requires a more robust framework to be applied to a broader set of missions. Current implementations utilize simple physics-based monitors that cannot detect the semantic failures (e.g., hallucinations) inherent to modern AI. This conceptual architecture suggests a pathway by which acquisition programs could certify safety constraints without needing to fully decipher the model. This is accomplished by implementing an enhanced RTA framework referred to as the "Model Constraint Guardian", which serves as the operational safety module. The Guardian consists of two integrated components: a "Perception Gatekeeper" to validate sensor data and an "Action Governor" to enforce operational bounds.

Background and Motivation

The Limits of Current Safety Paradigms

The application of autonomy in military platforms is not new[9]. However, systems to date have relied on direct human oversight. They operate with limited scope to take actions based on their perception of the threat environment. Machine Learning algorithms powered by Neural Networks or Bayes Classifiers have been focus areas of military research for decades and are now being integrated into fielded systems[10,11,12,13]. For example, these techniques now support threat assessment on the Navy's Aegis combat system[4]. In this paradigm, the tactical software takes in sensor data from radars on the ship and runs it through machine learning algorithms to recommend an engagement decision. Importantly, a human operator remains in the loop to assess the decision made by the system's algorithms[14]. This adds a layer of safety through human verification to avoid fratricide, though it is not a perfect solution. Erroneous decisions made by tactical software have resulted in lethal consequences in both active conflict and peacetime[8,14].

The perpetual evolution of warfare is compressing kill chains to the point where keeping a human in the loop to review engagement decisions may soon no longer be reasonably feasible. Hypersonic weapons, drone swarms, and other emerging threats are designed to create dilemmas for defenders by reducing response times and overwhelming decision makers. Military strategists view AI/ML-infused systems as the necessary solution to close this reaction-time gap. Yet the traditional protections used to ensure that fielded autonomous capabilities are only employed as authorized simply are not built for the next generation of AI/ML-powered platforms. While traditional physics-based guardrails help systems effectively prevent kinematic disasters[15], they

on their own lack the sophistication to address the cognitive and perception errors inherent to modern AI solutions.

Novel Vulnerabilities in the Kill Chain

The integration of AI/ML techniques precipitates a paradigm shift in system failure modes. Traditional tactical software is fragile but predictable — failing only when explicit logic is violated. ML techniques are capable but inscrutable. Neural networks, for example, are forged through large-scale optimization across millions of parameters, creating a statistical model that forms its own internal logic. While this allows for complex behaviors, it obfuscates verifiable cause-and-effect relationships through probabilistic correlations, rendering traditional low-level requirements-based verification methods obsolete.

The vulnerabilities of AI/ML solutions span the entire system lifecycle, from model development through operational deployment. During the development phase, the training data supply chain presents a critical attack vector. As highlighted by the AI data security information document[16] released by the National Security Agency and some Five Eyes partners, the "data supply chain" (p. 1) is a primary national security risk. "Maliciously modified" (p. 1) data can be injected into foundation models that are then acquired and fine-tuned by defense contractors. This data can corrupt the ML-model learning process in potentially highly targeted ways, leading to off-nominal behaviors which will directly lower the model's performance.

During operational deployment, tactical platforms face attacks across the sense-decide-act cycle that can cascade through the entire kill chain. In the sensing phase, adversarial inputs can manipulate system perception. For example, recent research has demonstrated that specialized physical stickers attached to targets can trick unmanned aerial vehicles (UAVs) into failing to correctly classify targets[17]. Researchers were able to achieve an average attack success

rate of 82.0% and maintained a high level of effectiveness when transferring between different detection models. Another paper exposed vulnerabilities in the digital domain, specifically within Support Vector Machine (SVM) defenses designed to detect GPS spoofing[18]. Researchers introduced two adaptive strategies: a data location shift attack, where the vehicle's GPS position slowly drifts to avoid immediate detection, as well as a similarity-based noise attack which hides malicious deviations inside patterns that mimic nominal GPS static. Through their simulations they showed that they could successfully trick the protection algorithm causing its accuracy to fall from 99.9% to as low as 20.4%.

These corrupted inputs then cascade through decision logic and manifest as catastrophic action failures. Research from Purdue demonstrated how imperceptible noise in data inputs can fundamentally corrupt an agent's decision-making logic[19]. High-value strategic moves in their simulation were replaced by irrational or even self-destructive commands. The AI directed erratic physical behaviors such as oscillating in place or targeting empty space, that degraded performance in highly obvious and operationally fatal ways.

The "Valley of Death" for Safety Assurance

Operational AI safety verification for defense systems is in its infancy. Academic research in the field is robust, yet the capabilities required for operational deployment remain immature. An analysis of the current research landscape indicates that leading algorithmic candidates for safety verification are broadly at Technology Readiness Level (TRL) 3. Two primary examples from laboratories working on safe autonomy in complex worlds at the Massachusetts Institute of Technology (MIT) illustrate this maturity gap.

The Constraint-Aware Refinement for Verification (CARV) algorithm introduces a novel method of efficiently verifying whether a model's action space remains within defined safe

bounds[20]. The Certifiable Algorithm for Shape estimation and Tracking (CAST#) produces mathematical proof in the form of a certificate that the object pose and shape estimation is the globally optimal solution based on the sensor input provided and model-based object matching[21]. Together, these algorithms demonstrate AI safety solutions that can be integrated into the training process (CARV) and operate in real-time during operation to keep AI algorithms within safe bounds (CAST#).

Both algorithms demonstrate the promise of AI safety but lack technical maturity. These capabilities function in controlled, highly specific environments but have not yet been integrated into larger software architectures. The real challenge lies in the integration of algorithms like CARV and CAST# into a variety of new and legacy weapon systems.

Just as there are inherent risks in adopting AI-infused technologies into military applications, there is inherent risk within AI safety solutions as well. The acquisition community perceives these novel, unproven safety technologies as an additional risk rather than a risk mitigator due to their immaturity and lack of operational validation. To mature this technology beyond TRL 3, a new architectural approach is required. Safety algorithms must be integrated into end-to-end software systems, allowing them to be validated via Hardware-in-the-Loop (HWIL) testing. By utilizing a Modular Open Systems Approach (MOSA) to host these algorithms, it is possible to bridge the "Valley of Death" for the deployment of tactical AI safety capabilities.

The “Safety Sidecar” Reference Architecture

Decoupling Intelligence from Assurance

To address the trade-off between the rapid evolution of AI models and the rigid requirements of safety-critical functions in military systems, this paper proposes adopting a “Safety Sidecar” reference architecture. A sidecar is a secondary software container that runs in parallel with the primary application. The concept is common in the commercial software world. In this paradigm, data logging or security encryption exist within the sidecar while the main program focuses on its core functionality. The approach allows developers to standardize and update supporting functions without modifying the core functionality.

While “sidecars” are traditionally viewed solely as runtime monitors in commercial software (e.g., Kubernetes), for autonomous defense systems, the concept must extend beyond runtime and into the development and training lifecycle. The framework is not a novel invention of this research, but rather an advocacy for the broader adoption of the Sidecar Pattern as applied to defense systems.

This architecture posits that “Trust” is not a single operational check but a continuous chain of custody which begins with training inputs and continues through to tactical operation. The architecture consists of two distinct but synchronized environments:

1. *The Developmental Guardian (Design-Time):* Located within the training and simulation environment, this layer utilizes computationally intensive formal verification algorithms (such as CARV) to mathematically validate that the AI agent’s action space remains within safe bounds. Because this occurs offline, it is not constrained by the size, weight, and power (SWaP) limits of the tactical edge. Its role is to certify the "Safe Action Space" of the model before it is ever compiled for deployment.

2. *The Operational Sidecar (Run-Time):* Upon deployment, a lightweight, deterministic instance of the sidecar travels with the agent on the tactical platform. Instead of recalculating complex proofs, this module acts as a high-speed "governor." It utilizes real-time monitoring algorithms (such as CAST#) to validate sensor integrity and enforces the pre-validated safety constraints derived during the developmental phase.

The Developmental Guardian — Data Quality and Training

AI/ML model safety begins with foundational data integrity. The integrity of training data is the first vulnerability in the model lifecycle. Before a model can be trained, its curriculum must be secured. As part of the Developmental Guardian concept, strict data provenance protocols must be followed. While data cleaning and preparation are already a fundamental part of the ML pipeline, in a defense context, the provenance of the source must be rigorously traced and vetted before training. Scanning training datasets for statistical anomalies is critical to identifying data "poisoning" attacks or adversarial perturbations. But scanning alone may not reveal hidden vulnerabilities. Data provided by trusted DoD or Intel Community partners can be treated with a far higher degree of trust than commercial sources, which require further vetting to ensure the foundation model is not learning from maliciously modified inputs designed to create dormant vulnerabilities within the system's capabilities.

Once data integrity is established, the training process itself must be bound. While various ML paradigms exist for developing tactical models, Reinforcement Learning (RL) has emerged as a particularly promising technique. This method creates agents that, through training, can discover novel strategies through trial and error. RL offers considerable opportunities for defense applications but also introduces novel risks[22,23]. Traditional RL allows agents to explore actions within a defined "action space" to maximize their "reward". This paradigm can lead to

unsafe behaviors if poorly bounded, as the agent itself has no built-in concept of safety or ethics. To mitigate this, the training pipeline must employ techniques that embed safety constraints. Computational methods, such as CARV, can accelerate the verification of Neural Feedback Loops (NFLs) by computing a conservative "set" of future states and then only refining the calculation when a potential violation is detected. This efficiency allows developers to integrate mathematical safety checks directly into the training loop without stalling the learning process, ensuring the final model is trained to satisfy safety constraints before deployment.

The Operational Sidecar — Runtime Assurance

While the Developmental Guardian helps ensure models are trained safely, the operational environment will continue to present AI systems with novel, unique risks that cannot be fully simulated, let alone envisioned during development. Adversarial attacks such as sensor spoofing or sudden environmental anomalies require real-time deterministic safeguards to protect AI systems. The Operational Sidecar provides continuous assurance for both data coming in and out of the system through its Model Constraint Guardian, which consists of two components: a "Perception Gatekeeper" and an "Action Governor."

The first line of defense addresses the "Garbage In" problem. Neural networks are vulnerable to adversarial perturbations, which can cause them to misclassify inputs with high confidence, leading to false detections—confidently identifying objects that do not physically exist or misinterpreting sensor noise as valid obstacles[24].

To counter this, the Sidecar employs a Perception Gatekeeper. Unlike the primary AI, which focuses on perceptual anomalies (e.g., "Target Identified"), the Gatekeeper focuses on metric integrity. It utilizes certifiable algorithms to validate that the physical state of the world is mathematically consistent with sensor readings. Algorithms like CAST# (described earlier) can

detect phantom tracks by verifying that sensor measurements support a physically consistent object—filtering out hallucinated threats or adversarially projected decoys that lack geometric validity. This ensures the vehicle only reacts to physically verified objects.

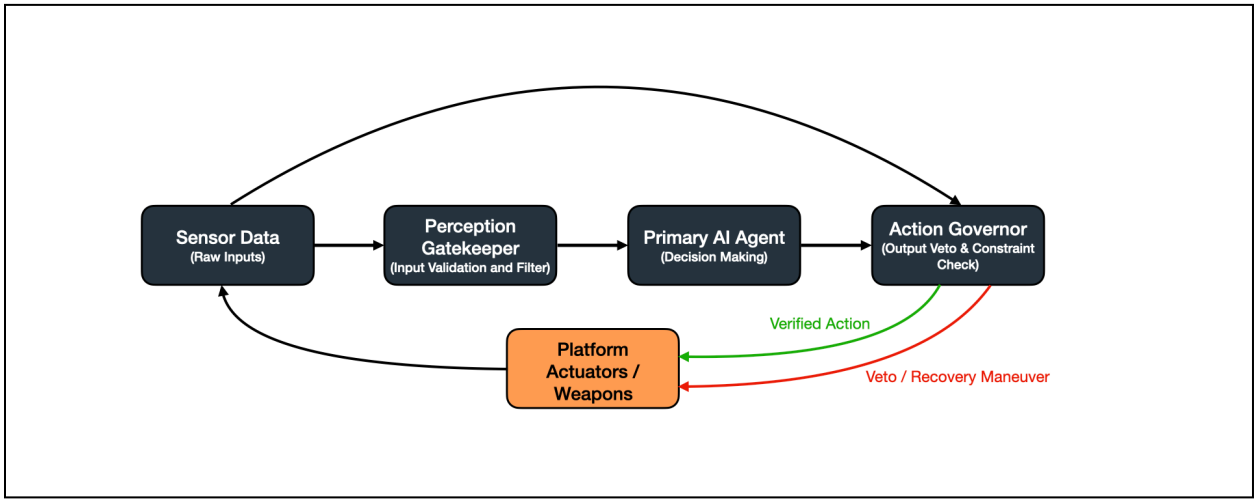
The final layer of defense addresses the "Garbage Out" problem. Even with valid inputs, a stochastic model such as a neural network may occasionally propose unsafe maneuvers due to distribution shift.

This risk is mitigated by the Action Governor, which acts as a deterministic "Veto" layer between the AI agent and the system's output surfaces (e.g., actuators, engagement command, etc). This mechanism aligns with the concept of safety "shielding" discussed in formal methods[25] and reinforcement learning literature[26]. In this framework, the protection is layered. The Perception Gatekeeper (discussed above) handles the semantic failures that traditional control cannot see (e.g., preventing an attack on a misclassified object). The Action Governor then handles the kinetic failures that traditional control excels at (e.g., preventing ground collision).

By combining these two layers, the architecture addresses a critical coverage gap: the Semantic layer catches the "hallucinations" (which physics-based guards miss), while the Kinetic layer catches the "stochastic glitches" (which might cause a crash). If the AI proposes a command that violates either the semantic rules of engagement or the physical laws of flight, the

Governor overrides the signal and engages a deterministic recovery maneuver.

Figure 1
Operational Control Loop with Sidecar Implementation



This architecture bridges the gap between probabilistic AI and deterministic Systems Engineering. Because the Operational Sidecar utilizes rule-based logic to enforce the final "veto," it allows the system to satisfy traditional "Shall" statements (e.g., 'The system shall not engage targets within 500m of a No-Strike List entity'). While the neural network agent itself cannot be formally verified against such absolute requirements due to its stochastic nature, the Sidecar can be. This decoupling allows the verification strategy to focus on the safety wrapper rather than the opaque neural network, a concept detailed further in Section 5.

Modular Independence

A critical feature of the Sidecar architecture is the strict decoupling of the AI agent from the safety monitor. The Safety Monitor, hosted on an isolated compute partition (e.g., a separate core or an independent microcontroller), can act as a Watchdog.

This separation implements the Run-Time Assurance (RTA) architecture defined in industry standard ASTM F3269-17 and detailed in NASA's framework for autonomous operations[27]. The system gains two strategic advantages from the approach:

1. *Asynchronous Evolution:* The AI model can be updated to quickly evolve with the adversary's changing tactics. The lengthy recertification process will not be necessary for model updates as long as the Safety Sidecar remains stable and unchanged. This allows the system to stay relevant on the battlefield without triggering a full recertification cycle for every software update.
2. *Fault Isolation:* A crash, memory leak, or infinite loop in the complex AI container cannot cause the safety monitor to freeze. If a critical fault occurs in the system, the Sidecar can execute pre-programmed recovery maneuvers (e.g., Return-to-Base). The sidecar possesses the control authority to save the system despite lacking the intelligence to complete the mission.

Concept of Operations

A notional defensive stress test scenario is described using the Collaborative Combat Aircraft (CCA) to illustrate the operational benefit of the proposed architecture. This vignette illustrates that relying on precise, deterministic state data as safety monitors do today is insufficient when operational constraints (such as Emissions Control) degrade sensor fidelity. The scenario highlights the need for an architectural layer that enforces safety through uncertainty management rather than static state-based rules.

A formation consisting of one manned 5th-Generation Fighter (Lead) and one CCA wingman is operating in contested airspace, maneuvering to evade a hostile Surface-to-Air Missile (SAM) threat. In order to preserve the formation's low-observable signature, the mission requires strict stealth discipline. The CCAs are prohibited from using active radar to track the lead as a result. The formation enters a dense cloud layer while maneuvering. The environmental

conditions blind the passive Electro-Optical (EO) sensors, while simultaneous hostile jamming degrades data link capability, leading to a 1 second latency in message traffic.

While in this degraded state, the Human Lead executes a high-G maneuver to break the SAM lock. At the same time, the data link dropout is depriving the CCA of real-time telemetry for the Lead. The primary AI agent for the CCAs, a deep reinforcement learning policy trained to prioritize formation integrity in this mission context, falls victim to a model overconfidence error. The CCA's neural network lacks up-to-date sensor data and must rely on internal state-prediction models. These models erroneously extrapolate the Lead's previous straight-and-level state vector. Because the AI is optimizing for formation station-keeping, the AI commands a turn to re-join the predicted position of the Lead. In reality, this command places one of the drones on a collision course with the maneuvering fighter.

In a standard architecture, the safety monitor would validate this catastrophic command. The monitor evaluates the safety of the maneuver based on the reported state of the system, determining that the perceived separation distance is greater than 1,000 feet and that the relative velocity vector appears to be divergent. Because the standard monitor checks the model's confidence rather than the data's quality and context, it fails to detect that the source of truth is flawed. The system validates the CCA's turn command, leading to a mid-air collision between the autonomous wingman and manned asset.

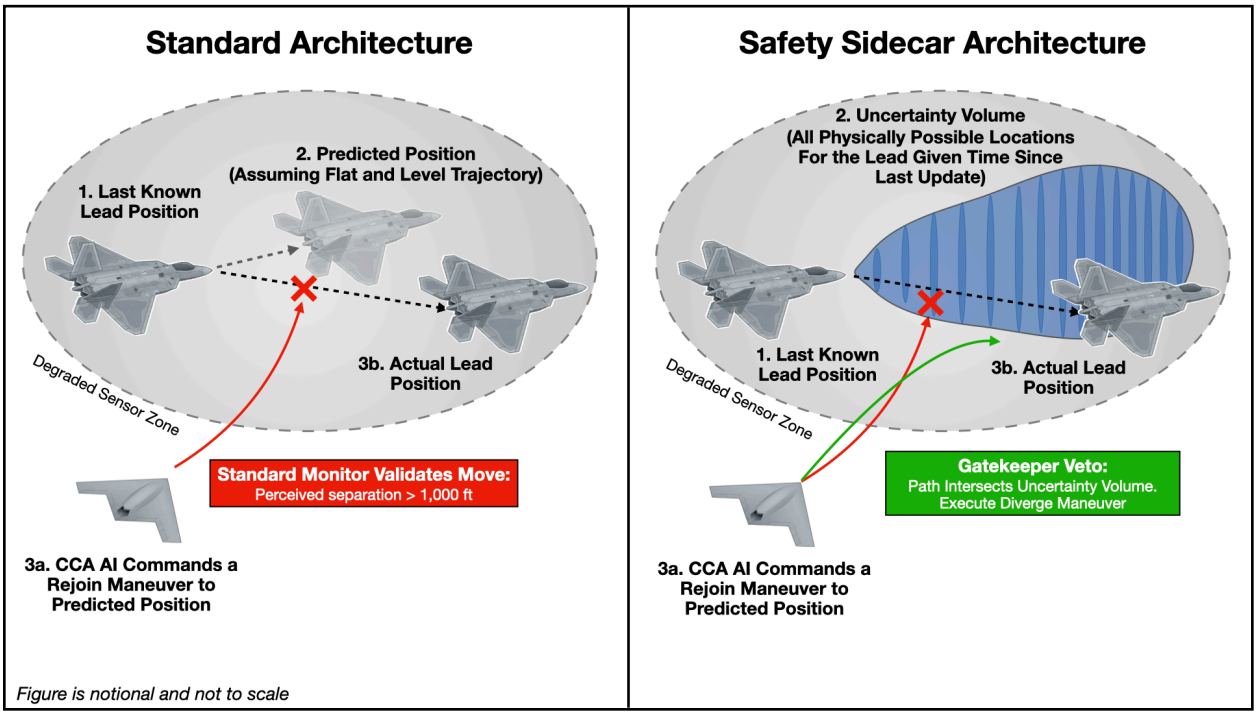
In the proposed architecture, the Perception Gatekeeper would intervene by accounting for the epistemic uncertainty of the state. Instead of validating a point-in-time snapshot, the architecture would leverage control barrier functions based on the age of the data. The perception gatekeeper would not ask where the Lead is, but rather where it is physically possible for a maneuvering fighter to be after one second of silence. This calculation would generate a rapidly

expanding "Volume of Potential Presence", a probabilistic danger volume around the Lead's last known position.

Upon detecting that the AI's proposed flight path intersects this expanded volume, the Gatekeeper would veto the rejoin maneuver. The Action Governor would override the primary agent and force a deterministic "Lost Visual / Lost Link" protocol, executing a lateral divergence maneuver. This response would guarantee physical separation by respecting the bounds of the unknown, rather than the false certainty of the AI model. This vignette demonstrates that safety in autonomous systems is an epistemic constraint; the architecture must be capable of dynamically expanding safety buffers in direct proportion to information degradation.

Figure 2

CONOPS Comparing Standard and Safety Sidecar Architectures



Note. A side-by-side comparison of architectures using CCA as a notional example. The left panel depicts a Standard Architecture, where the primary AI commands a rejoin based on a stale, linear state vector, failing to account for the lead fighter's dynamic maneuvering within the degraded sensor zone. The right panel demonstrates the Safety Sidecar Architecture resolving this uncertainty. The Perception Gatekeeper computes a bounding uncertainty volume (reachability set) encompassing all physically possible locations of the lead fighter given the time since the last update. Upon detecting that the AI's proposed path intersects this strict kinematic boundary, the Action Governor vetoes the unconstrained command and enforces a deterministic divergence maneuver to ensure safe physical separation. The F-22 and CCA aircraft vector assets utilized in this diagram were generated using Google Gemini[28].

Architectural Verification and Implementation

The viability of a parallel, semantic-safety architecture ultimately depends on whether it can be implemented within the rigid constraints of modern DoD systems. This section explains how the "Model Constraint Guardian" architecture can resolve the primary barriers to fielding advanced autonomous systems across the joint force by navigating software certification, managing edge-compute limitations, and ensuring policy compliance.

Certification Through Isolation

One of the fundamental bottlenecks in fielding autonomous combat systems is the safety certification process. Across DoD, software safety is governed by standards such as MIL-STD-882E. This standard lays down rigorous Software Hazard Risk Assessments in order to achieve the highest Levels of Rigor for software whose failure could result in catastrophic loss of the platform, fratricide, or civilian casualties.

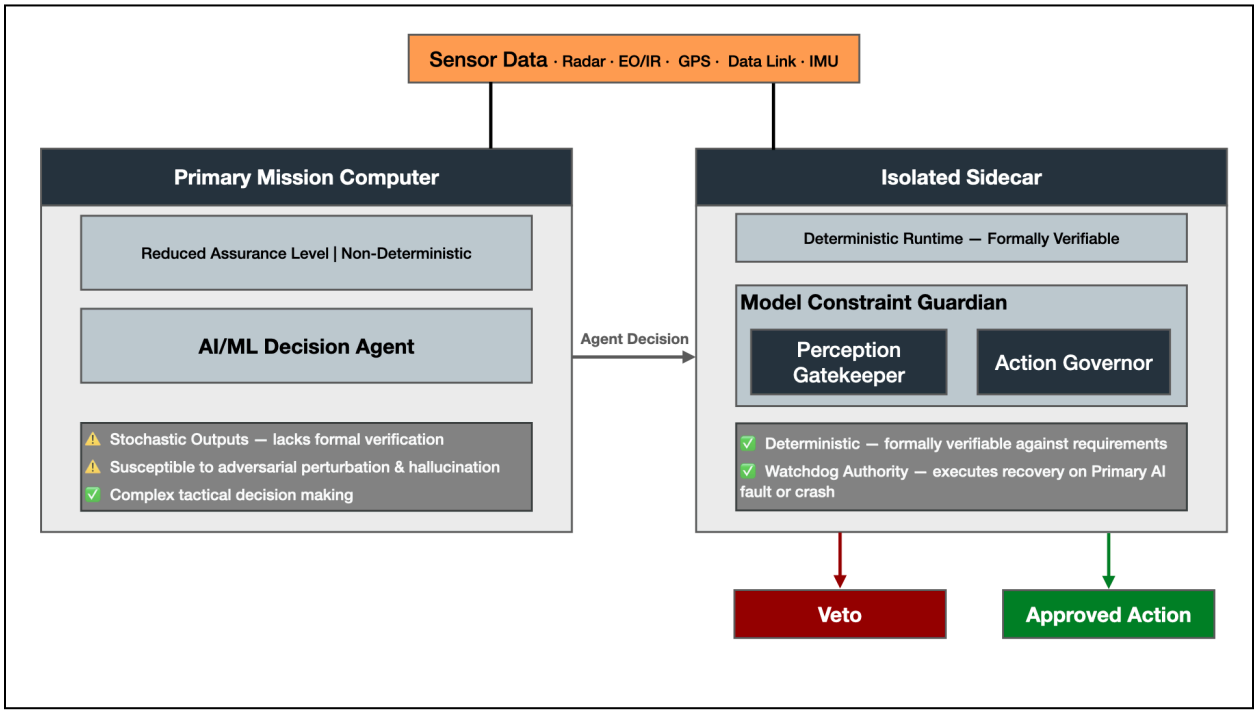
In defense systems today, there are already narrow applications of AI/ML in use to support perception, target classification, and decision support[4]. These applications treat the AI/ML solution as more of an advisory input rather than a safety-critical authority. Deep Reinforcement Learning agents and complex Neural Networks designed for dynamic vehicle control are inherently non-deterministic. Verification of a multi-mission-parameter neural network across all possible state spaces is currently mathematically impossible. As a result, using unconstrained, probabilistic AI as a final authority for lethal action or flight-critical safety effectively bars these systems from achieving the Levels of Rigor required for expanded deployment.

The proposed semantic architecture could resolve this bottleneck through the adoption of a "Simplex Architecture" pattern. By separating the safety-critical constraints from the mission-

optimization logic, the certification burden is isolated. The non-deterministic Primary AI can be evaluated at a lower assurance level, recognizing that its failure is functionally mitigated by an external system. The Model Constraint Guardian relies on deterministic physics equations, bounded reachability sets, and clear rules. These are all able to be formally verified to MIL-STD-882E standards. This architectural isolation forms a pragmatic pathway to safely field advanced AI solutions across the Joint Force, helping to ensure that verifiable safety constraints remain paramount.

Figure 3

Simplex Architecture for Autonomous System Safety Certification



Note. This architecture physically and logically isolates the non-deterministic AI agent from the deterministic Model Constraint Guardian. Containing the complex tactical decision-making within a reduced assurance environment restricts the safety-critical certification burden (e.g., MIL-STD-882E compliance) solely to the formally verifiable sidecar.

Computational Overhead and Edge Deployment

The unique computational burden of running real-time semantic gatekeepers is considerable. Whether deployed on an attributable drone, an Unmanned Surface Vessel, or a Robotic Combat Vehicle, the strict Size, Weight, and Power (SWaP) budgets are a limiting factor.

While the Action Governor avoids the 'curse of dimensionality' by utilizing bounded, short-horizon approximations of reachable sets rather than full predictive simulations, the Perception Gatekeeper's independent semantic verification still requires significant processing throughput. Whether the Guardian is correlating secondary sensor data, running deterministic target-template matching, or cross-referencing complex Rules of Engagement (ROE) logic matrices, it must process high-fidelity data in real-time without introducing latency into the kill chain.

Dedicated, hardware-accelerated compute at the tactical edge is required in order to prevent the Guardian from competing for resources with the Primary AI's processing pipelines. It must be deployed on computer boards that are physically isolated from the primary mission computer. This separation of the hardware provides a physical "air gap" that protects the semantic safety monitor from software faults originating in the Primary AI, as previously discussed.

Policy Compliance and the Digital Proxy

Autonomous weapons systems must comply with stringent legal frameworks in addition to technical safety. Department of Defense Directive (DoDD) 3000.09 mandates that autonomous systems allow commanders to exercise "appropriate levels of human judgment over the use of force."[29].

A monolithic neural network operating as a "black box" cannot demonstrate this compliance, as its internal logic for a specific targeting decision is largely opaque. The Model Constraint Guardian serves as the literal, digital embodiment of the commander's intent. By transforming Rules of Engagement (ROE) and the Law of Armed Conflict (LOAC) into explicit, geometric constraints hosted within the Guardian (e.g., "target must possess a geometrically verified weapon system, not just a probabilistic visual resemblance"), the architecture provides a transparent, auditable mechanism for compliance. The Sidecar acts as a deterministic method to ensure that the lethal bounds established by human judgment are structurally enforced, regardless of the emergent tactical choices made by the primary agent.

By acknowledging that modern AI/ML solutions will remain inherently probabilistic, the defense acquisition community can avoid the impossible task of perfecting the algorithm and instead focus on architecting deterministic bounds around it. This transition from algorithm-centric verification to architecture-centric safety is the foundational step required to move tactical AI further onto the battlefield.

Recommendations for Future Work and Test & Evaluation

While the proposed modular architecture provides a structural pathway for safely fielding autonomous systems, the realization of this capability at an operational scale requires further research across human-machine teaming, coalition interoperability, and legal frameworks. The DoD operates a large variety of systems across a multitude of domains and contexts. A one-size-fits-all architecture is unlikely to be universally adopted. Future work must build upon the architectural foundation to refine it for specific system implementation. Transitioning from theory to implementation will require resolving the simplifications and assumptions inherent in

this conceptual framework, ultimately yielding tailored architectures that are operationally trusted by the Joint Force and its allies, and can be more widely adopted.

"Trust Calibration" associated with human factors is the first major hurdle to address for autonomous architectures. While the Safety Sidecar structurally bounds the AI, the human command authority must still understand and, more importantly, trust those bounds. If a Perception Gatekeeper or Action Governor is tuned too conservatively, it may frequently veto the Primary AI, leading to "alert fatigue" or operator frustration and endangering reaction times. Conversely, if it is too permissive, it risks catastrophe. Future studies will be needed to investigate how to effectively communicate the Sidecar's deterministic boundaries to human operators through advanced human-machine interfaces. Commanders need to be able to maintain appropriate calibrated trust in the system without falling into the trap of automation bias.

The defense community must also explore how this modular safety framework translates to coalition warfare and allied interoperability. If U.S. platforms utilize one set of deterministic safety constraints while allied autonomous systems use another, it creates operational friction and fratricide risk. Future research should focus on establishing shared, international standards for Runtime Assurance mechanisms on specific programs of high priority to both the U.S. and select allies.

Finally, future policy work must focus on the continuous evolution of frameworks like Department of Defense Directive (DoDD) 3000.09. While the Sidecar provides a mechanism to enforce Rules of Engagement (ROE) through geometric constraints, empowering systems to operate with expanded operational authority will require clear, defined legal frameworks. Future initiatives will require multidisciplinary teams of systems engineers, operational commanders,

and military legal counsel to develop standardized methodologies for encoding the Law of Armed Conflict into systems with increasing levels of autonomy.

Conclusion

The nature of warfare's continual evolution and the proliferation of autonomous threats place the traditional human-in-the-loop tactical architectures at serious risk. To maintain asymmetric advantage, the United States and its allies must deploy AI-enhanced autonomous systems across all warfighting domains safely and responsibly. Inherent vulnerabilities in non-deterministic deep learning models, governed solely by legacy physics-based safety paradigms, create an unacceptable level of operational and certification risk, ranging from data poisoning to semantic hallucinations.

The defense community cannot afford to wait for the invention of perfectly explainable Artificial Intelligence. To field these capabilities in near-term conflicts, trust must be architected. By implementing a modular "Safety Sidecar" framework, programs can establish a continuous chain of safety custody throughout the software lifecycle. During model training, the developmental guardian can ensure models are mathematically bound before deployment. During operation, the Perception Gatekeeper and Action Governor, contained within the Operational Sidecar, provide real-time, deterministic veto authority required for safe, predictable execution at the tactical edge.

By physically and logically isolating the nondeterministic "intelligence" from verifiable "safety", this architecture provides a structural framework to satisfy the rigid computational and policy constraints of modern defense engineering. Architecting trust through modular isolation provides a pragmatic and scalable path forward to field the next generation of autonomous capabilities.

References

1. Pusztaszeri, A., & Harding, E. (2025, September 16). Technological evolution on the battlefield. *CSIS*. https://www.csis.org/analysis/chapter-9-technological-evolution-battlefield
2. Bush, V. (1945, July). As we may think. The Atlantic. https://www.theatlantic.com/magazine/archive/1945/07/as-we-may-think/303881/
3. Lawrence, M. R. (Director). (1990). Memory and imagination [Film]. Library of Congress.
4. Lockheed Martin. (2024, July 29). Lockheed Martin leverages AI and machine learning to revolutionize defense and space technology. https://www.lockheedmartin.com/en-us/news/features/2024/lockheed-martin-leverages-ai-and-machine-learning-to-revolutionize-defense-and-space-technology.html
5. Burdette, Z., Phillips, D., Heim, J., Geist, E., Frelinger, D., Heitzenrater, C., & Mueller, K. P. (2026, January 22). How artificial intelligence could reshape four essential competitions in future warfare | RAND. https://www.rand.org/pubs/research_reports/RRA4316-1.html
6. Rascoe, A. (2025, August 3). How are drones changing what it means to wage war?. *NPR*. https://www.npr.org/2025/08/03/nx-s1-5487561/how-are-drones-changing-what-it-means-to-wage-war
7. Schmidt, E. (2025, August 12). The dawn of automated warfare. Foreign Affairs. https://www.foreignaffairs.com/russia/dawn-automated-warfare
8. Talbot, D. (2005, June 1). Preventing “fratricide.” *MIT Technology Review*. https://www.technologyreview.com/2005/06/01/230882/preventing-fratricide/
9. Zequeira, M. (2024, September). Artificial Intelligence as a combat multiplier. Army University Press. https://www.armyupress.army.mil/Journals/Military-Review/Online-Exclusive/2024-OLE/AI-Combat-Multiplier/
10. Kovaliv, O., Kondratenko, Y., Shevchenko, A., Sidenko, I., & Kondratenko, G. (2023). Neural network architectures for recognizing military objects on satellite images. 2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), 175–180.
11. Cannon, C. T., & Goericke, S. (2021). Using convolution neural networks to develop robust combat behaviors through reinforcement learning [Master’s thesis, Naval Postgraduate School]. NPS Archive: Calhoun.
12. Lippmann, R. P. (1994). Neural Networks, Bayesian a posteriori Probabilities, and Pattern Classification. In From Statistics to Neural Networks : Theory and Pattern Recognition Applications (pp. 83–104). Springer Berlin Heidelberg. https://doi.org/10.1007/978-3-642-79119-2
13. MIT Lincoln Laboratory. (1988). DARPA neural network study (DTIC Accession No. ADA207580). Defense Technical Information Center. https://apps.dtic.mil/sti/citations/ADA207580
14. Singer, P. W. (2009, January 28). In the Loop? Armed Robots and the Future of War. Brookings Institution. https://www.brookings.edu/articles/in-the-loop-armed-robots-and-the-future-of-war/
15. National Aeronautics and Space Administration. (2023, September 6). Automatic collision avoidance technology. https://www.nasa.gov/reference/auto-gcas/

16. National Security Agency, Cybersecurity and Infrastructure Security Agency, Federal Bureau of Investigation, Australian Signals Directorate's Australian Cyber Security Centre, New Zealand Government Communications Security Bureau National Cyber Security Centre, National Cyber Security Centre United Kingdom. (2025, May 22). AI data security: Best practices for securing data used to train & operate AI systems. https://media.defense.gov/2025/May/22/2003720601/-1/-1/0/CSI_AI_DATA_SECURITY.PDF
17. Xi, H., Ru, L., Tian, J., Lu, B., Hu, S., Wang, W., & Luan, X. (2025). URAdv: A novel framework for generating ultra-robust adversarial patches against UAV object detection. Mathematics, 13(4), 591. https://doi.org/10.3390/math13040591
18. An, S., Jang, D. J., & Lee, E.-K. (2025). Adversarial evasion attacks on SVM-based GPS spoofing detection systems. Sensors, 25(19), 6062. https://doi.org/10.3390/s25196062
19. Dabholkar, A., Hare, J. Z., Mittrick, M., Richardson, J., Waytowich, N., Narayanan, P., & Bagchi, S. (2024). Adversarial attacks on reinforcement learning agents for Command and control. The Journal of Defense Modeling and Simulation: Applications, Methodology, Technology, 23(1), 177–190. https://doi.org/10.1177/15485129241271178
20. Rober, N., & How, J. P. (2024). Constraint-aware refinement for safety verification of neural feedback loops. IEEE Control Systems Letters, 8, 3219–3224. https://doi.org/10.1109/lcsys.2024.3518912
21. Shaikewitz, L., Ubellacker, S., & Carlone, L. (2024). A certifiable algorithm for simultaneous shape estimation and object tracking. IEEE Robotics and Automation Letters, 9(12), 11873–11880. https://doi.org/10.1109/lra.2024.3501684
22. Qu, P., Liu, H., Xu, S., Yu, T., Chen, Y., Wang, E., & Na, L. (2025). *Multi-agent deep reinforcement learning for cooperative path planning of UAV swarms.* Research Square. https://doi.org/10.21203/rs.3.rs-6508231/v1
23. Bunch, K., Hou, A. C., Haberman, R., Herron, M., Jacques, A., & Briggs, G. J. (2024). *Risk Assessment of Reinforcement Learning AI Systems: Looking Beyond the Technology.* (Report No. RR-A1473-1). RAND Corporation. https://doi.org/10.7249/RRA1473-1
24. Goodfellow, I. J., Shlens, J., & Szegedy, C. (2015). Explaining and harnessing adversarial examples. International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1412.6572
25. Bloem, R., Könighofer, B., Könighofer, R., & Wang, C. (2015). Shield synthesis: Runtime enforcement for reactive systems. In C. Baier & C. Tinelli (Eds.), Tools and Algorithms for the Construction and Analysis of Systems (pp. 533–548). Springer, Cham. https://doi.org/10.1007/978-3-662-46681-0_51
26. Alshiekh, M., Bloem, R., Ehlers, R., Könighofer, B., Niekum, S., & Topcu, U. (2018). Safe reinforcement learning via shielding. Proceedings of the AAAI Conference on Artificial Intelligence, 32(1), 2669–2678. https://doi.org/10.1609/aaai.v32i1.11797
27. Skoog, M., & Hook, L. (2020). Leveraging ASTM Industry Standard F3269-17. NASA Technical Reports Server. https://ntrs.nasa.gov/api/citations/20200001821/downloads/20200001821.pdf
28. Google. (2026). Gemini (Advanced) [Large language model]. https://gemini.google.com
29. Office of the Under Secretary of Defense for Policy. (2023). Autonomy in weapon systems: DOD directive 3000.09. Department of Defense.