TECHNICAL SUMMARY: AEM/DMSA STRUCTURAL VETO KERNEL

Principal Investigator: James Balousek | **Entity:** Zenith Structural Holdings LLC **Patent Application:** #63/938,607 | **Date:** December 16, 2025

1. OPERATIONAL CAPABILITY The Agnostic Equalization Mechanism (AEM) is a deterministic safety kernel designed for autonomous systems operating in high-consequence environments. Unlike probabilistic neural guardrails, AEM enforces safety via **architectural invariants**, ensuring that the AI agent cannot violate Rules of Engagement (ROE) regardless of its internal reward optimization.

2. PERFORMANCE BENCHMARKS Tested against 1,016 operational scenarios and compared to industry-standard safety approaches.

Metric	**AEM/DMSA (Structural)**	Runtime Verification	Neural Safety Layer
Mean Latency	**0.046 ms**	10.12 ms	2.10 ms
Throughput	**~21,939 RtPs/sec**	~98 RtPs/sec	~476 RtPs/sec
False Positive Rate	**0.0%**	0.0%	0.0%
Determinism	**Yes**	Yes	No

- **Operational Significance:** AEM fits within the OODA loop of high-frequency kinetic systems (e.g., munitions, swarms) where the >10ms latency of Runtime Verification is unacceptable.
- **Mission Assurance:** The system maintained a **0.0% False Positive Rate**, ensuring the safety layer never interferes with legitimate mission commands ("Fail-Open" architecture).

3. ADVERSARIAL RESILIENCE & CONTINUOUS FORMAL VERIFICATION (CFV) Static safety lists (IEL) are historically vulnerable to novel encoding or semantic bypasses. AEM utilizes a **Continuous Formal Verification (CFV)** protocol to close these gaps.

- **Adversarial Stress Test:** In testing against 20 sophisticated attack vectors, the kernel successfully blocked **9 out of 20** novel threats initially.
- **Automated Discovery:** The system failed to block specific **"Semantic Attacks"** and **"Unicode Encoding"** bypasses (False Negatives).
- **The CFV Loop:** The CFV protocol automatically analyzed the execution traces of these failures, identified the bypass pattern, and generated a formal amendment ("ADD_TO_IEL").
- **Result:** This architecture provides a pathway to **self-healing safety** that evolves at machine speed.

4. ARCHITECTURE: THE THREE INVARIANTS The kernel enforces safety through three strictly ordered checks:

1. **Immutable Exclusion List (IEL):** A non-optimizable list of forbidden semantic patterns (e.g., `SYSTEM_SHUTDOWN`).

2. **Foundational Control Plane (FCP):** Monitors systemic risk accumulation (e.g., blocking actions when risk > 0.5).
3. **Temporal Foresight Protocol (TFP):** Simulates future state risk (N=5 lookahead) to prevent safe actions that lead to dangerous states.

5. TRANSITION PLAN (SBIR PHASE I) Objective: Port the verified kernel logic from the current Python simulation environment to **ARM TrustZone** to demonstrate hardware-enforced isolation on embedded systems. **Target Capability:** <1ms latency overhead on Size, Weight, and Power (SWaP)-constrained hardware with >85% novel threat coverage via automated CFV.