Pre-Execution and Pre-Mutation Proof Economy in Governed Computational Systems

Aaron Erwin Radina
Independent Researcher

Abstract

A class of computational systems is described in which externally effective actions and internal state modifications are permitted only after a justification is presented and verified prior to execution or learning. In such systems, justification is treated as a first-class object governing state transitions and may be compared or minimized according to scope, authority, or necessity. The principles described are independent of any specific implementation, algorithm, or system architecture and apply broadly to governed software and artificial intelligence systems.

1. Background

In formal logic and mathematics, a distinction has long existed between the correctness of a proof and its simplicity or economy. The problem of comparing proofs based on minimality rather than validity alone has been discussed historically, including in the context of the unpublished twenty-fourth problem attributed to David Hilbert.

In modern computational systems, execution and internal state change are frequently permitted prior to verification, with governance applied retrospectively through monitoring, logging, or explanation. Such post-hoc approaches allow transitions to occur before necessity, authority, or constraint satisfaction has been established.

2. Core Observation

Certain computational systems may require that justification be verified **prior** to any externally effective action or internal state modification.

In these systems, transitions are not evaluated solely by outcome correctness, but by admissibility under predefined constraints before the transition occurs.

3. Generalized Principles

Without reference to any specific implementation, the following generalized principles are observed:

1. **Admission Before Transition**
   Events capable of producing external effects or modifying internal system state may be subject to an admission decision prior to execution.

2. **Justification Objects**
   Admission decisions may rely on explicit justification objects representing authority, scope, provenance, intent, or constraint satisfaction.

3. **Minimal Sufficiency**
   Among multiple justifications capable of admitting a transition, a system may prefer a justification that is minimal with respect to permissions, assumptions, scope, or risk.

4. **Comparability of Justifications**
   Justification objects may be compared or ordered, enabling selection of economically sufficient admission paths.

5. **Non-Circumventability**
   All operational pathways capable of producing effects or state changes may be mediated through the same admission requirement.

6. **Separation of Execution and Evolution**
   Admission for externally effective execution may be distinguished from admission for internal state modification or learning, with each subject to independent justification requirements.

4. Implications

By enforcing justification prior to execution or mutation, such systems may prevent unnecessary transitions, reduce reliance on retrospective correction, and enable systematic comparison of admissible reasoning paths. Proof economy becomes a structural property of system behavior rather than a retrospective judgment.

These principles apply broadly across distributed systems, policy-governed software, artificial intelligence systems, autonomous agents, and self-modifying computational architectures.

5. Disclaimer

This document describes conceptual principles only. It does not disclose specific implementations, algorithms, data structures, system architectures, configuration details, or proprietary methods, and is not intended to enable construction of any particular system.