Authority Engineering: Horizons and Horizon Crossings

Rob Caswell

Authority Engineering

June 2026

Published by Cyber Terrain LLC | cyberterrain.ai

DOI: https://doi.org/10.5281/zenodo.21742286

CYBER TERRAIN

Publication Record

Authority Engineering: Horizons and Horizon Crossings

Author: Rob Caswell

Publisher: Cyber Terrain LLC

Published: 2026-06-01

Zenodo version: v1.0

DOI: https://doi.org/10.5281/zenodo.21742286

Website: https://cyberterrain.ai/papers/ae-horizons-horizon-crossings.pdf

Corpus Context

Corpus: Authority Engineering

Document ID: AE-HORIZON-0001

Object family: AE-HORIZON-0001

Publisher-imprint edition prepared for versioned republication.
The prior Zenodo record remains the citable record until its new version is published.

© 2026 Cyber Terrain LLC. All rights reserved.

Abstract

The Authority Engineering corpus establishes how authority-bearing possibility is organized, bounded, and converted into binding consequence via local consequence seams. However, prior foundational documents assume a singular, uniform realization environment, leaving the framework structurally isolated and unable to account for transitions between foreign systems. This paper resolves the final structural variable of the corpus by formalizing the mechanics of inter-environmental movement. We introduce the concepts of Realization Environments, Horizons, and Horizon Crossings as the objective structural regimes governing state transformation across distinct system boundaries. We prove through the *Topological Induction Theorem* that horizons are not unique primitives, but are derived liminal intervals induced exclusively by structural incompatibilities between adjacent realization topologies. We define the four-dimensional Horizon State vector $\langle P, L, T, R \rangle$ and establish a strict three-phase structural sequence for crossings, ensuring that systemic transformation is fully executed before downstream identity preservation can be evaluated. By decoupling local realization from inter-environmental transformation, this document provides the definitive, airtight boundary layer required before the corpus divides into its derived preservation (Continuity) and pressure (STOIC) branches.

Rob Caswell | linkedin.com/in/rob-caswell3

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

1. Purpose & Scope

1.1 Document Motivation

Every prior foundational paper in the Authority Engineering corpus operates under a silent, necessary simplification: that realization occurs within the stable, uniform confines of a single realization environment. We have formalized how authority is organized, how its boundaries are mapped via Reachable Consequence Transitions (RCT), and how local consequence seams execute state changes.

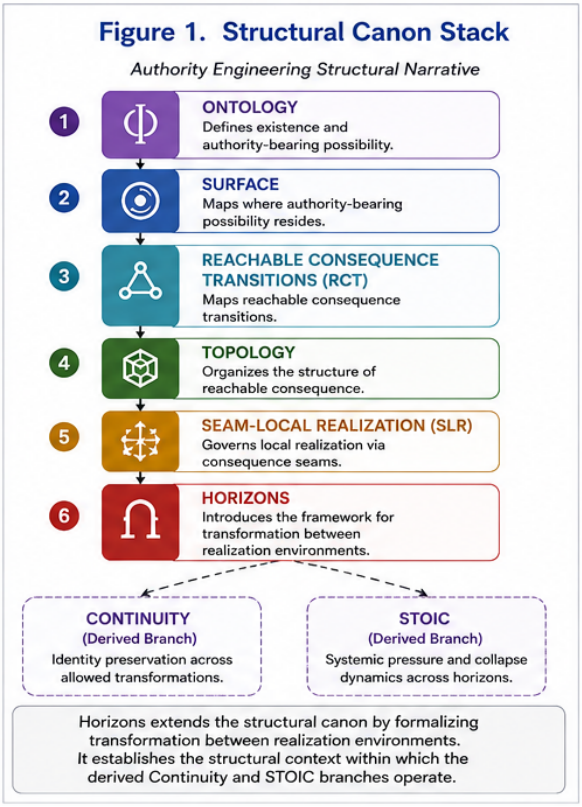
However, real-world structural systems are rarely monolithic. Systems mutate, scale, fragment, and interface with external architectures governed by foreign laws. Without a dedicated structural theory governing how states migrate across *incompatible* systems, the corpus remains trapped in regional isolation. It cannot explain cross-environmental transformation, structural mutation, or systemic migration.

This document establishes the definitive structural mechanics for inter-environmental movement. It introduces Horizons and Horizon Crossings not as conceptual metaphors, but as objective, derived structural regimes that constrain and govern transformation between incompatible realization environments.

1.2 Structural Canon Stack

The structural narrative of Authority Engineering functions as an axiomatic layered stack. Each document builds sequentially on the metrics established by its predecessor:

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286



By settling the mechanics of inter-environmental movement, this paper acts as the definitive capstone of the structural layer, providing a clean handoff to the downstream preservation and pressure branches.

2. Why Horizons Are Necessary

The structural narrative of Authority Engineering progressed logically through the definition of existence, surface, reachability, topology, and local realization. However, at the outer limit of local realization, a fundamental theoretical gap emerges. Documents 1–7 lack the mechanics to transition between distinct systems.

- The Surface identifies where authority-bearing possibility exists, but assumes a singular metric of authority.
- RCT maps which futures are reachable, but requires a continuous, unbroken causality graph.
- Topology organizes reachable consequence, but merely describes the internal structure of a given system.
- SLR governs realization, but local consequence seams are blind to foreign environmental rules.

Horizons are the necessary structural mechanism to resolve this systemic gap, acting as the capstone that seals the structural branch of the corpus before it splits into preservation and pressure dynamics.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

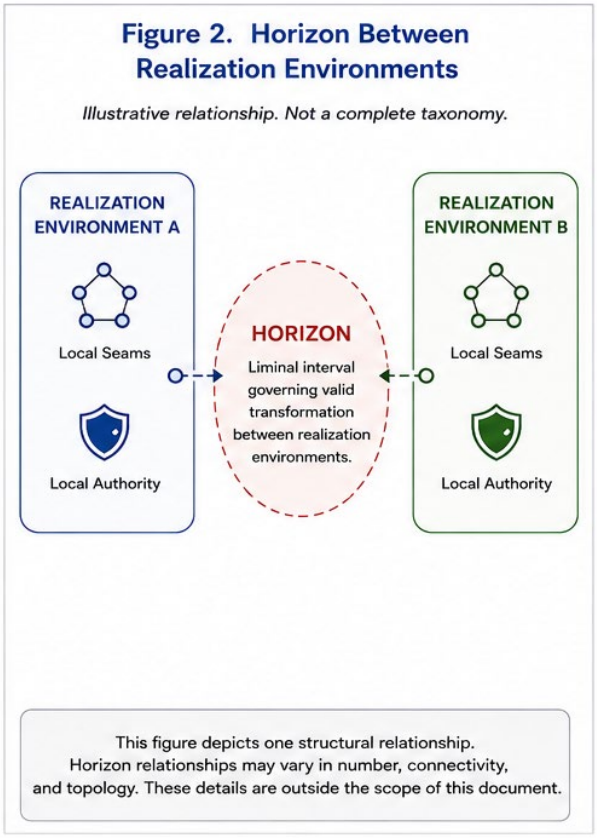
3. Structural Foundations: The Realization Environment

3.1 Section Motivation

To understand how an entity leaves a system, the corpus must first define what encloses that system. If the boundaries of an environment are ill-defined, the concept of a boundary crossing becomes mathematically meaningless. We must formalize the structural enclosure of realization before we can map its exit vectors.

3.2 Definition

A Realization Environment (E) is the bounded structural domain within which a consistent, closed set of Consequence Seams governs the conversion of authority-bearing possibility into binding consequence. Every realization operation is structurally enclosed; no realization can occur in an environmental vacuum.



3.3 Topological Induction & Necessity

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

Horizons are not injected as new primitives into the ontology. They are structurally induced by environmental variance. If two environments share identical topological metrics, movement between them is a flat routing operation. A Horizon is induced *only* when structural incompatibility requires transformation.

Theorem 1: Topological Induction Theorem

Distinct realization topologies necessarily induce distinct realization environments. Wherever such environments exhibit structural incompatibility at their operational boundaries, a Horizon Interval is induced.

Theorem 2: Transformation Necessity Theorem

Wherever topological incompatibility prevents direct seam-local realization between adjacent environments, bounded structural transformation is mandatory. State transition cannot occur via native routing.

Mathematically, we formalize the induced Horizon H_{AB} separating environments E_A and E_B as a bounded liminal interval defined by two specialized boundary consequence seams:

$$\mathcal{H}_{AB} = [\mathcal{S}_{\partial A}, \mathcal{S}_{\partial B}]$$

Where:

- $\mathcal{S}_{\partial A}$ is the **Ingress Seam**: The specialized boundary seam that suspends the operational authority of Environment A over the crossing object.
- $\mathcal{S}_{\partial B}$ is the **Egress Seam**: The specialized boundary seam that binds the incoming object to the operational authority of Environment B.

4. Horizon State Matrix

4.1 Section Motivation

Not all horizons behave identically. Some permit unrestricted transformation, others prevent crossing entirely, while still others permit only constrained, conditional, or temporary transit. If Horizons are objective structural constructs, they must themselves possess objective structural state. Horizon State formalizes those measurable conditions.

4.2 The State Vector

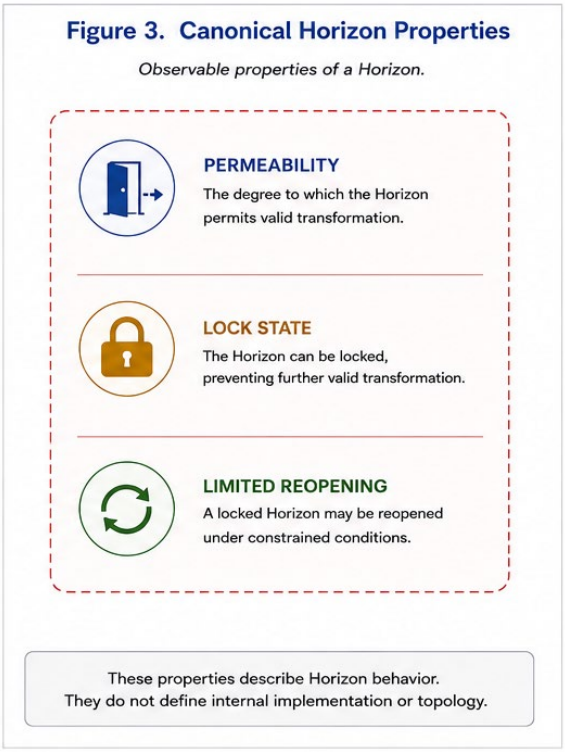
A Horizon possesses an observable structural condition independent of the objects attempting to traverse it. This condition is captured by the Horizon State vector $S_{\mathcal{H}}$:

$$S_{\mathcal{H}} = \langle P, L, T, R \rangle$$

- **Permeability ($\boldsymbol{P \in [0, 1]}$):** The scalar index of structural resistance. At $\mathrm{P} = 1$, states pass with zero translation friction. At $\mathrm{P} = 0$, the interval is a total structural block.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

- **Lock State ($\boldsymbol{L \in [0, 1]}$):** An absolute binary logical gate. When $\mathrm{L} = 1$, the Ingress Seam $(\mathcal{S}_{\partial A})$ is completely sealed, refusing all incoming transformation requests regardless of the permeability index.
- **Traversability ($\mathbf{T}$):** The internal structural integrity of the interval itself, determining whether an object can cross the structural width of the horizon without experiencing collapse, decay, or divergence mid-transit.
- **Limited Reopening ($\mathbf{R}$):** A highly constrained transition function that dictates if, how, and when a locked or impermeable horizon temporarily relaxes its parameters to allow specialized, throttled transit before snapping back to a base closed state.



5. Realization vs. Transformation & Architectural Diagram

5.1 Section Motivation

System architects frequently conflate state modifications *within* an environment with system-wide transformations *between* environments. To maintain architectural clarity, we enforce a strict separation between these two operations, mapping how topology directly induces the transformation loop.

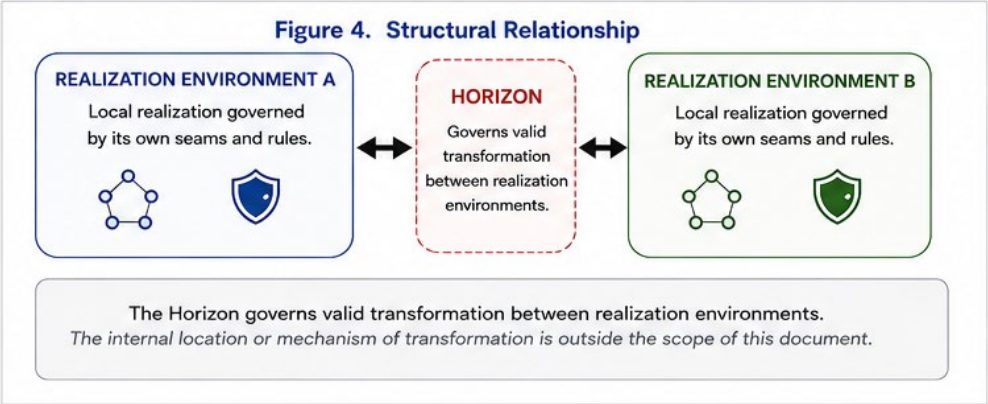
5.2 Operational Distinctions

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

- **Consequence Seams (Intra-Environment):** Operate exclusively within a single environment ($\mathcal{E}_{\mathrm{A}}$). They modify local properties, converting local authority-bearing possibility into local binding consequence.
- **Horizons (Inter-Environment):** Operate strictly between distinct, incompatible realization environments ($\mathcal{E}_{\mathrm{A}} \rightarrow \mathcal{E}_{\mathrm{B}}$). Their output is a transformed crossing object mapped to the validation structures of the destination environment.

5.3 Full Structural Capstone Map

Figure 4 summarizes the structural relationship established by this document. Realization occurs within bounded realization environments governed by local consequence seams. Horizon Crossings govern valid transformation between realization environments. This figure illustrates the structural relationship only and is not intended as a complete ontology of Horizon architectures or internal transformation mechanics.

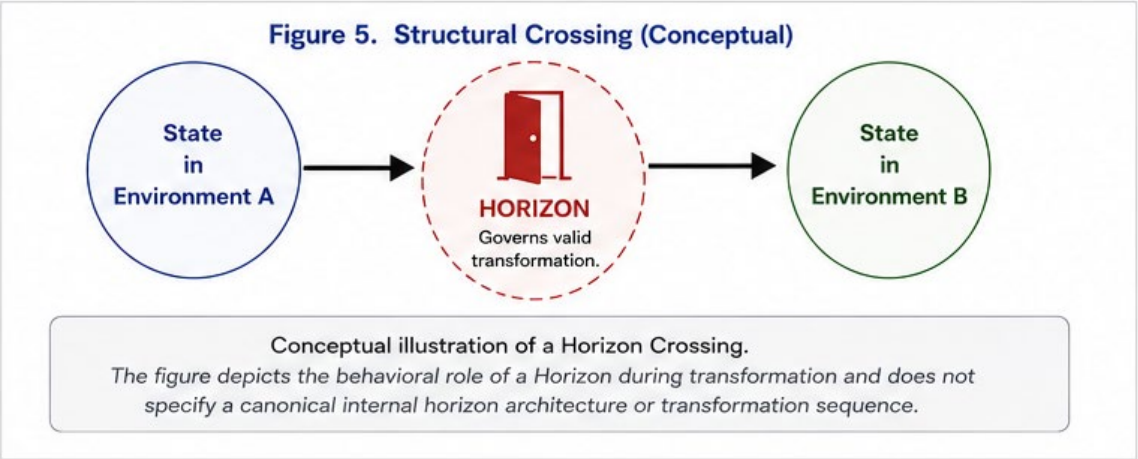


6. The Anatomy of a Horizon Crossing

6.1 Section Motivation

A crossing cannot be treated as an instantaneous leap from one environment to another. Because environments are structurally isolated, transition requires a disciplined, multi-phase sequence of authorization, structural dissolution, translation, and reintegration. Defining this sequence ensures crossings are repeatable and logically bounded.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286



6.2 The Structural Sequence

The entity migrating through the interval is designated as the Crossing Object (O_C).

[O_C Input] ⟶ Phase 1: Ingress Validation ⟶ Phase 2: Liminal Transformation ⟶ Phase 3: Environmental Integration ⟶ [O_C Bound]

Phase 1: Ingress Validation:

Authority Suspension.

The Crossing Object (O_C) interacts with the Ingress Seam ($\mathcal{S}_{\partial A}$). The Horizon evaluates its parameters (L, P). Upon structural validation, the native authority of E_A is suspended, releasing its local consequence rules over the object.

Phase 2: Liminal Transformation:

Structural Remapping.

The object traverses the internal length of the Horizon Interval (H_{AB}). Stripped of its source environment context, its structural layout is reformatted, translated, and remapped to meet the parsing conditions mandated by the destination system.

Phase 3: Environmental Integration:

Authority Binding.

The translated object arrives at the Egress Seam ($\mathcal{S}_{\partial A}$). The destination environment (E_B) verifies the post-transformation integrity. If accepted, the object is introduced to local consequence seams, binding it to the new realization environment.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

7. Canonical Laws

Law 1: Environmental Enclosure

Every realization occurs within a distinct, bounded realization environment. No realization can exist in a non-environmental vacuum.

Law 2: Horizon Isolation

Distinct, topologically incompatible realization environments are separated by Horizons. No two such environments can directly touch, merge, or overlap without inducing a derived Horizon Interval.

Law 3: Construct Pragmatism

Horizons are objective derived structural constructs possessing a measurable, observable structural state vector ($\mathcal{S}_H$). They are not conceptual abstractions.

Law 4: Bounded Transformation

Transformation between realization environments occurs exclusively through formal Horizon Crossings. Unbounded or un-isolated leaps between environments are structurally impossible.

Law 5: Outcome Determination

The structural state of a Horizon ($\mathcal{S}_H$) determines the admissible outcome space of any attempted Horizon Crossing.

Law 6: Separation of Continuity

Successful transformation must completely occur before any evaluation of continuity can begin. The structural transit does not guarantee or determine the preservation of identity; it only dictates the possibility of arrival.

8. Structural Failure Modes & Consequences

8.1 Execution Matrix

Because crossings are constrained by the dynamic values of the Horizon State vector, transits frequently fail. The architecture categorizes these abort states into four discrete structural failure archetypes:

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

Failure Archetype	Locus of Failure	Primary Cause	Structural Result
Invalid Transformation	Ingress Seam ($\mathcal{S}_{\partial A}$)	Object configuration cannot map to the horizon's translation matrix.	Instant rejection; object remains bound to native environment (E_A).
Blocked Crossing	Ingress Seam ($\mathcal{S}_{\partial A}$)	Horizon Lock is engaged ($\mathrm{L} = 1$) or Permeability is zero ($\mathrm{P} = 0$).	Transit denied; state remains stationary in source environment (E_A).
Terminated Crossing	Horizon Interval (H_{AB})	Horizon State shifts mid-transit (e.g., dynamic flash-locking or T-collapse).	Structural scattering, fragmentation, or complete loss of object within the liminal zone.
Locked Horizon Trap	Egress Seam ($\mathcal{S}_{\partial A}$)	Target environment rejects post-translation layout; failed egress validation.	Object is stranded inside the interval, permanently decoupled from both environments.

8.2 Structural Consequences

By formalizing the Horizon model, the Authority Engineering corpus locks down five critical architectural facts:

1. **Decoupling of Operations:** Realization (local state updates) and Transformation (systemic environmental shifts) run on completely non-overlapping layers.
2. **State Supremacy:** Crossings are not governed by the intent or character of the moving object; they are strictly bounded by the objective parameters of the Horizon State vector ($\mathcal{S}_H$).
3. **The Prerequisite Matrix:** Transformation is the absolute structural prerequisite for Continuity. A system cannot ask if an identity survived a transition until the mechanics of transformation have successfully completed.
4. **Isolated System Protection:** Because no two incompatible topologies can touch without inducing a horizon, environments are perfectly insulated from cascading external failures.
5. **Dynamic Control Hooks:** Because the Horizon State vector is isolated from the structural definitions of horizons themselves, it provides a clean architectural hook for the STOIC branch to dynamically alter transit rules based on systemic pressure without violating core structural topology.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

9. Glossary of Canonical Terms

- **Realization Environment ($\mathbf{E}$):** The bounded structural domain within which a consistent, closed set of consequence seams governs the conversion of possibility into consequence.
- **Horizon ($\mathbf{H}$):** An objective, derived structural interval separating topologically incompatible realization environments.
- **Horizon Crossing:** A bounded, multi-phase transformation process occurring through a Horizon Interval.
- **Crossing Object (O_C):** The entity or state packet undergoing a transformation process across a Horizon.
- **Ingress Seam ($\mathcal{S}_{\partial A}$):** The specialized boundary consequence seam that suspends a source environment's authority over a crossing object.
- **Egress Seam ($\mathcal{S}_{\partial B}$):** The specialized boundary consequence seam that binds a transformed crossing object to a destination environment's authority.
- **Horizon State ($\mathcal{S}_H$):** The measurable structural condition of a horizon, formalized as the vector $\langle P, L, T, R \rangle$.
- **Transformation:** The structural process of altering an object's state configuration to achieve admission into a foreign realization environment.

10. Open Research Boundary

The following behaviors are explicitly excluded from this baseline document and represent the current research frontier for the Structural Canon:

1. **Horizon Composition:** The structural behavior and alignment rules governing multiple horizons stacked in parallel or serial configurations.
2. **Nested Horizons:** Deep structural intervals within intervals, observed primarily in multi-layered virtualization layers or fractured cyber-terrain architectures.
3. **Convergence & Divergence Geometries:** The mechanics of many-to-one environmental integrations and one-to-many environmental branchings executing during a single coordinated crossing process.

Rob Caswell | linkedin.com/in/rob-caswell3

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286

Appendix: Canonical Applications

The following examples illustrate the downstream applicability of the theory. They serve an illustrative purpose and are not part of the structural theory itself.

A The Execution Boundary: Low-Level Kernel to Userland Isolation

In modern operating system architectures, moving telemetry or state payloads across the ring boundary—from kernel-space (Environment A, governed by non-blocking, memory-constrained, privileged execution) to userland (Environment B, governed by dynamic, heap-allocated, non-privileged runtimes)—cannot be modeled as flat data routing. The environments operate on incompatible execution semantics and security models.

A secure ring-buffer or system-call interface functions precisely as a formal **Horizon Interval**.

- The **Ingress Seam ($\mathcal{S}_{\partial A}$)** intercepts the transition, stripping the raw memory address of its supervisor privileges and isolating the execution thread.
- The **Horizon State Vector ($\mathcal{S}_H$)** explicitly governs the boundary conditions: if userland encounters thread starvation or memory pressure, the Permeability (P) index is dynamically dialed down to rate-limit or filter incoming traffic, insulating the kernel from user-space exploitation or collapse. If a fatal ring exception occurs, the Lock State $(\mathrm{L} = 1)$ engages instantly to isolate the core operating system from cascading failures.

B. The Translation Boundary: Compilation and Intermediate Representation

The lifecycle of software translation provides a pristine logical manifestation of the transformation loop. Consider source-level, human-readable functional code (Environment A, governed by high-level semantic rules, typing hierarchies, and lexical scopes) transforming into optimized machine code (Environment B, governed strictly by hardware registers, linear memory offsets, and instruction sets).

The compiler's pipeline operates as a Horizon Interval, utilizing an Intermediate Representation (IR) or Abstract Syntax Tree (AST) as the Crossing Object (O_C):

- **Phase 1 Ingress Validation:** The compiler frontend parses the source code. If a syntax or type violation occurs, it triggers an Invalid Transformation failure, immediately rejecting the transit.
- **Phase 2 Liminal Transformation:** The source code's high-level semantics are completely dissolved. The state properties are translated into a flattened, environment-neutral IR. Here, optimization passes alter the shape of the code without binding it to any specific physical target.
- **Phase 3 Environmental Integration:** The backend code generator acts as the Egress Seam, validating the optimized IR and mapping it directly onto the hardware topology of the target processor, where local hardware consequence seams take over execution.

Published by Cyber Terrain LLC | cyberterrain.ai DOI: https://doi.org/10.5281/zenodo.21742286